Quantum Information

Lecture: Tuesday 1415-1600

Dr James Wootton

Excercise Classes: Tuesday 1615-1800

First hint session today!

Final evaluation based on combination of results for exam and the graded exercises

Slides are released before every lecture:
sites.google.com/site/woottonjames

Good resources:
- Nielsen and Chuang (Book)
- Preskill's notes - www.theory.caltech.edu/people/preskill/ph229
- My Blog - medium.com/@decodoku
- Other stuff: github.com/desireevl/awesome-quantum-computing

Overview

The modern world depends on 'information technology'

This breaks everything (text, sound, pictures) down into numbers and does something with it, such as

Process (Computer)
Send (Communication)
Encrypt (Cryptography)
Preserve (Error correction)

'Quantum information technology' is the same, but using quantum systems as the building blocks, rather than bits

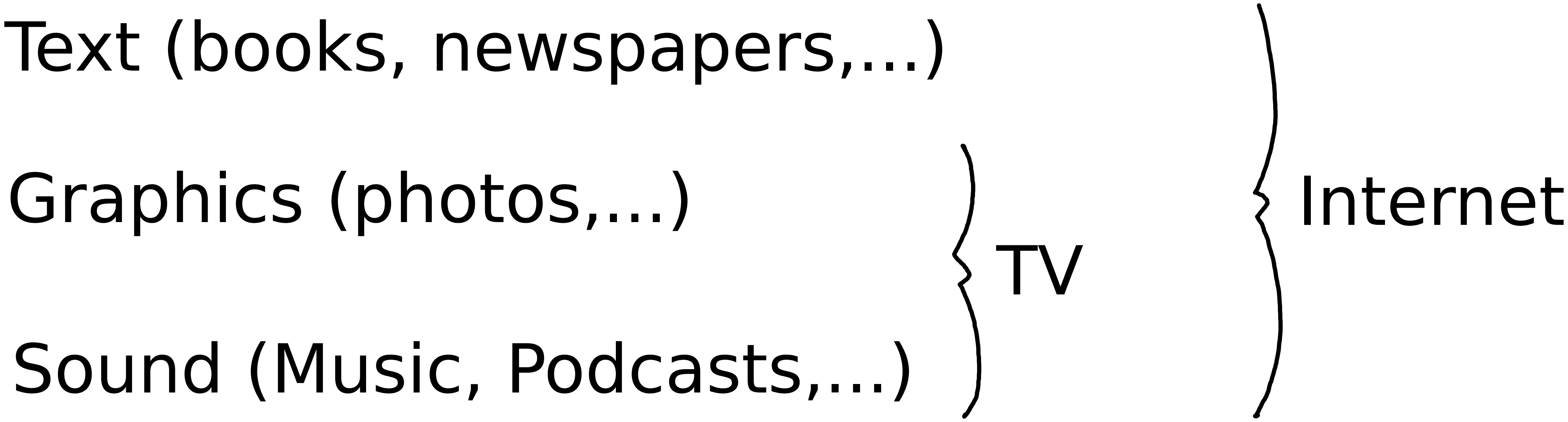
This allows us to do all the above in new and exciting ways

Classical Information Theory

N+C: 11.1-11.2, P: 5.1

What is information?

What forms does information come in?



Text (books, newspapers,...)

Graphics (photos,...)

Sound (Music, Podcasts,...)

TV

Internet

...

How can we quantify information?

Converting between information types

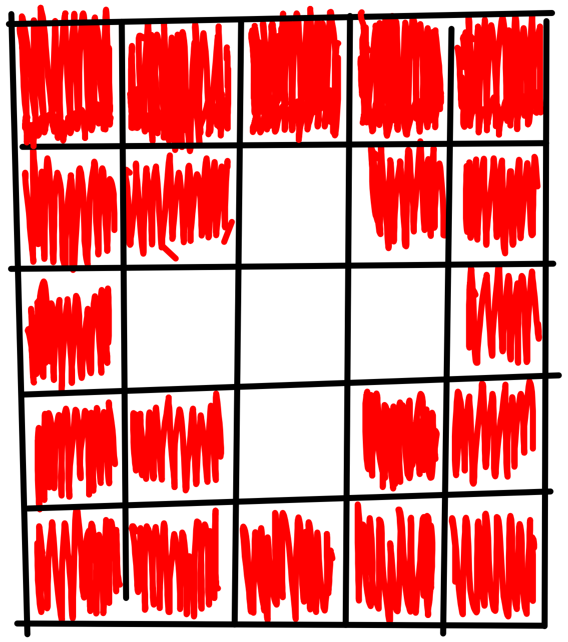
We can convert information stored in one form to information stored in others

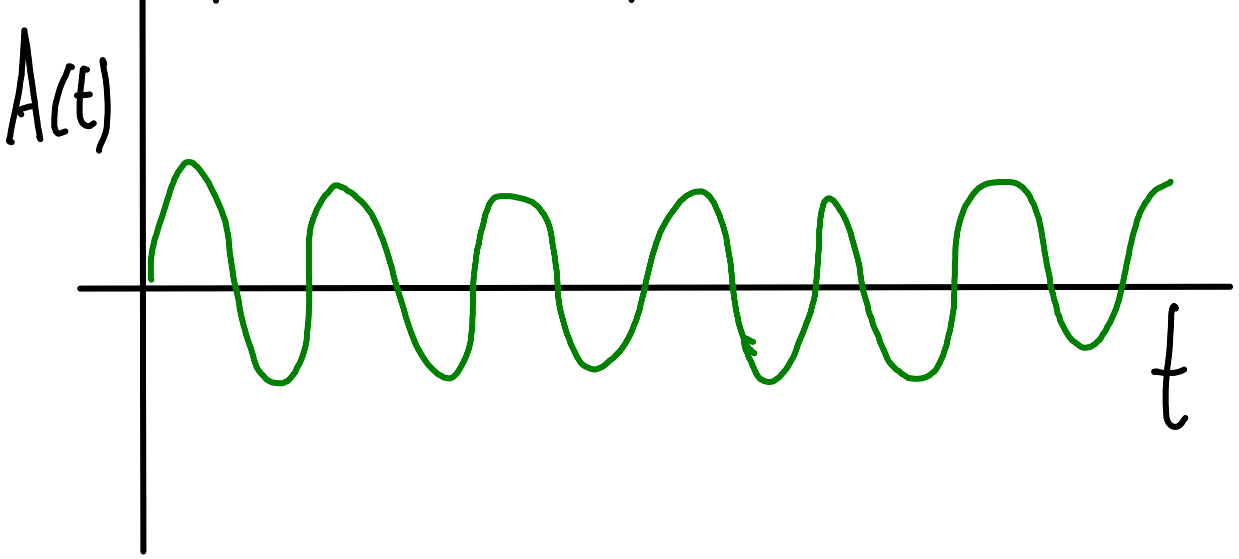
- Text -> Graphics: Take a picture of the text
- Text -> Sound: Read out and record the text
- Sound -> Graphics: Take a picture of the wave form

Any discrete information can be converted to text (and any continuous information can be approximated by discrete)

Pixel at (0,0) is red,
Pixel at (0,1) is red, ...
Pixel at (2,3) is white, ...

$A = 0.1$ at $t = \Delta t$
$A = 0.11$ at $t = 2\Delta t$
$A = 0.12$ at $t = 3\Delta t$
$A = 0.11$ at $t = 4\Delta t$





Text

Since text is universal, we'll focus on it alone

Requires an alphabet (set of symbols)

We (European humans) typically use the Latin alphabet

A, B, C, ... a, b, c, ...

ä, ƀ, ç, ... :-) ...

0, 1, 2, ...

~ 100 symbols

Computers use the binary alphabet

0, 1   2 symbols

A process for encoding meaning within the symbols is also required

Language + spelling convention (for Latin)

Machine code / ASCII (for binary)

We can convert between alphabets

For example: Latin -> Binary

ASCII

a → 1100001 b → 1100010 c → 1100011

A → 1000001 B → 1000010 C → 1000011

Morse Code

A = 10111 B = 111010101

C = 11101011101 D = 11101010

AB = 1011100111010101

So the answer to

"What forms does information come in?"

Is that we can, without loss of generality (at least for discrete information) consider only one form: strings of symbols

"Hi!"

"10010001101001100001"

"0x480x690x21"

Quantifying Information

How much information is contained within a given block of text?

Proposed Measure #1:

amount of information = # characters

The units of this would depend on the alphabet used

"Hi!" 3 latin characters

"100100011010010100001" 21 binary characters or 'bits'

Conversion between these units

$$\frac{\log(128)}{\log(2)} = 7$$ bits per latin character (using ASCII encoding)

Does this provide a good measure?

Consider:

1) QM textbook: N pages, 3000 characters per page
2) N pages, each containing nothing but 3000 zeros

Both contain the same number of characters, but do they contain the same amout of information?

Note that case (2) is completely described by

$$\underbrace{\text{"N}}_{\log N}\underbrace{\text{ pages, each containing nothing but 3000 zeros"}}_{34\ \text{characters}}$$

Since this sentence can reproduce case (2) completely, it must contain all the same information

So case (2) in fact holds no more than

$\log N + 34$ latin characters

of info

Note that the amount of information per page is then

$$\frac{34 + \log N}{N} \to 0 \ , \ N \to \infty$$

A well written textbook cannot be summarized so easily (though it can a bit using rules of spelling and grammar)

Amount of infomation per page will remain finite

Compression

This example shows us that the number of characters in a block of text depends the encoding scheme as well as the amount of information

We need to remove the dependence on the encoding from our measure of information

Done by considering only the best encoding scheme, which uses the smallest possible number of characters in a given alphabet: compression

Proposed Measure #2:

amount of information = # characters after optimal compression

Interestingly, we don't always need to know how to compress in order to know how many characters it will use!

Example: Random Variables

Let us do as we physicists usually do: consider a simple example

But what is simple? And what is complicated? Is a book a simple example of information, or a hard one?

Letters occur in text with different frequencies

English:

letter	p
e	0.127
t	0.096
a	0.082
...	...

German:

letter	p
e	0.174
n	0.098
i	0.076
...	...

So similar to a string of random characters, from a biased (language dependent) distribution

But it is more complex, there are correlations: 'q' almost always followed by 'u', '_a_' more likely than '_e_'.

These correlations make things complicated. They can also help with compression but need to be properly analyzed first

It would be simpler without them, so our simple example is that of i.i.d. random variables

X random variable,	x a value X may take
Coin toss	heads, tails
Dice throw	⚀, ⚁, ⚂, ... ⚅
Result of game	Win, Lose, draw

Random variable X

d possible values $x \in \{x_1, x_2, \ldots, x_d\}$

occur with probabilities $P(X = x_n) = P(x_n)$

How much information (how many characters when optimally compressed) do we need to store N outcomes in order? (Kolmogorov complexity)

$$K_N[P(x_n)] \quad \text{(measured in bits)}$$

And how many required per outcome? (Shannon Entropy)

$$H[P(x_n)] = \lim_{N \to \infty} \frac{K_N[P(x_n)]}{N} \quad \text{(measured in bits)}$$

For simplicity: d=2

$\mathcal{X} = \{0, 1\}$

Let's consider a few cases

$p(0) = 1, \; p(1) = 0$

N outcomes $\quad 0\,0\,0\,0\,0\,0\,0\,0\ldots$

Can be encoded as "N zeros" $\therefore K_N[p(x_n)] = \log_2 N + 35$ bits

$$\therefore H[p(x_n)] = \lim_{N\to\infty} \frac{\log_2 N + 35}{N} = 0 \text{ bits}$$

Asymptotically, each outcome is stored using zero bits!

$P(0) = P(1) = \frac{1}{2}$ N outcomes $0010000110011101011\ldots$

No nice patterns and so no nice encoding

Best option: print as is $\therefore K_N[p(x_n)] = N$ bits

$H[p(x_n)] = 1$ bit

$p(0) \approx 1, \quad p(1) \ll 1$ N outcomes $00000100000001000100001 1$

Encode using # consecutive zeros: $5, 7, 3, 4, 0$

On average, these are $\frac{1}{P(1)}$

# bits required to encode this $\sim \log_2 \frac{1}{P(1)} = -\log_2 P(1)$

Average number of such groups $\sim P(1)\, N$

Total # bits required $K_N[p(x_n)] \approx -N\, P(1) \log_2 P(1)$

$\therefore H[p(x_n)] \approx -P(1) \log_2 P(1)$ bits

In general, Shannon's noiseless, or source, coding theorem states

$d=2:$ $$H[p(x)] = -p(0)\log_2 p(0) - p(1)\log_2 p(1)$$ bits

$d \geqslant 2:$ $$H[p(x)] = -\sum_x p(x)\log_n p(x)$$ nits

Shannon entropy

The Shannon entropy

$$H[P(x)] = -\sum_{x} P(x) \log_n P(x)$$

Is the compression rate for i.i.d. random variables, and so a measure of information

It is also a measure of randomness

Not very random => easily compressable => not much information

Very random => difficult to compress => lots of information

In fact, it is the unique measure of randomness!

Shannon entropy and randomness

We have a probability distribution $\{p(x)\}$ for variable X

We want to quantify its randomness

Such a measure should satisfy the following conditions

1) Should be maximized by the uniform distribution

$$p(x) = \frac{1}{d} \quad \forall\, x$$

2) Should be a continuous function of the probabilities

3) If X is composed of two independent variables

$$\text{randomness } X = \text{randomness } X_1 + \text{randomness } X_2$$

These conditions are only satisfied for

$$H[p(x)] = -\sum_x p(x) \log_n p(x)$$

Choice of units is only free parameter

Shannon entropy and surprise

Can also be thought of as a measure of surprise!

If we get an x for which p(x)=1: Not at all surprising

$$S(x)=0 \quad \text{if} \quad p(x)=1$$

If we get an x for which p(x)=0: Infinitely surprising

$$S(x)=\infty \quad \text{if} \quad p(x)=0$$

The surprise for two independent events is the sum of the two surprises

$$S(x_1,x_2)=S(x_1)+S(x_2)$$

A suitable measure is $S(x) = -\log p(x)$

Shannon entropy is then the average surprise

$$\langle S(x)\rangle = \sum_x p(x)\, S(x) = H[X]$$

Shannon entropy and thermodynamic entropy

In thermodynamics, entropy is defined as

$$S = -k_B \sum_i p_i \ln p_i$$

Where the probabilities are those of the microstates at equilibrium

How is this related to Shannon entropy?

S is a measure of the information about the system that you do not know due to disorder

It is the information you would gain if you determined what microstate it was in

Note on notation

The Kolmogorov complexity and Shannon entropy are functions of probability distributions

$$K_N[p(x)] \qquad H[p(x)]$$

But sometimes it's more convenient to label them by the random variable

$$K_N[X] = K_N[p(x)] \qquad H[Y] = H[p(y)]$$

For d=2, probability distribution depends only on one variable

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

So we sometimes write

$$H[p(x)] = H(p) = -p\log p - (1-p)\log(1-p)$$

Summary so far

Information can take many forms, but it can always be treated as text

Text is a set of symbols + encoding scheme

Information content is minimum number of symbols required when best encoding scheme is used

The information required to store a set of random values is the Shannon entropy $H[p(x)] = H[X]$

Mutual Information

We've considered the entropy of a random variable X

What if this is made up of two component variables, Y and Z?

$$y \in \{heads, tails\} \qquad z \in \{1,2,3,4,5,6\}$$

$$X \in \{(heads,1), (heads,2), \ldots (tails,1), (tails,2)\}$$

The entropy for X is

$$H[p(x)] = -\sum_{x} p(x) \log p(x) = -\sum_{y,z} p(y,z) \log p(y,z)$$

For the case that Y and Z are uncorrelated

$$p(y,z) = p(y)p(z)$$

$$H[p(x)] = H[p(y)] + H[p(z)]$$

Exercise

They contribute independently to the entropy, as we required earlier

If they are not independent (they are correlated)

$$P(y,z) = P(y)\, P(z|y) = P(y|z)\, P(z)$$

$$P(y) = \sum_{z} P(y|z)\, P(z)$$

$$P(z) = \sum_{y} P(z|y)\, P(y)$$

The entropy of X will be less than from Y and Z combined

$$H[P(x)] < H[P(y)] + H[P(z)]$$

Exercise

This is because - Y and Z share information

- The correlations allow better suppression
- Learning the value of Y means you have some idea what Z will be, and so will be less surprised when you learn it

Example

$y \in \{\text{heads}, \text{tails}\} \quad z \in \{\text{heads}, \text{tails}\}$

$P(\text{heads}, \text{heads}) = P(\text{tails}, \text{tails}) = \frac{1}{2}$

$\therefore H[P(x)] = H[P(y)] = H[P(z)] = 1$ bit

both y and z store a bit, but it's the same bit

How can we quantity the amount of information shared by Y and Z?

$$H[p(y,z)] = \text{amount of information only in } y$$
$$+ \text{amount of information only in } z$$
$$+ \text{amount of information shared by } y \text{ and } z$$

$$H[p(y)] = \text{amount of information only in } y$$
$$+ \text{amount of information shared by } y \text{ and } z$$

$$\therefore I[p(y,z)] = H[p(y)] + H[p(z)] - H[p(y,z)]$$
$$= \text{amount of information shared by } y \text{ and } z$$

This is called the 'Mutual Information'

We can also use the notation

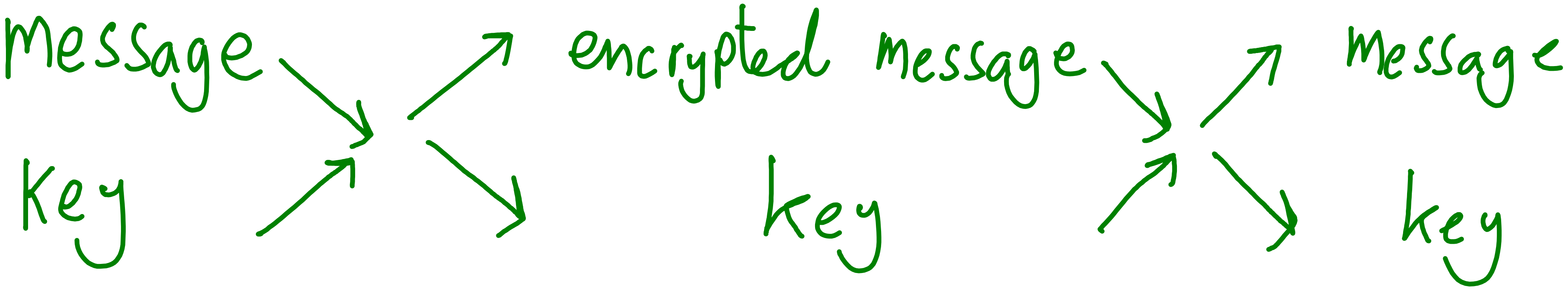
$$I(Y;Z) = H(Y) + H(Z) - H(Y,Z)$$

Cryptography

Sometimes we want to send information, which can be read only by the intended recipient

- Orders in war
- Bank account information

This is done using cryptography



We need to first share a secret with the receiver: The Key

This is then used the make the message unreadable

Receiver can use the key to make it readable again

Example: Ceaser Cipher

Consider a message encoded in 27 symbols:
a to z and space

"we will attack kings landing"

Choose a random number between 1 and 27 to act as the key, such as $k=2$, and cycle all symbols around by this

$a \to c,\ b \to d,\ c \to e,\ d \to f, \ldots, z \to a,\ \ \to b$

"ygbyknnbcvvcembkpiubncpfkpi"

Message is now unreadable

Anyone who knows the key can simply cycle back and make it readable again

But it is also easily cracked

- Frequency analysis
- Brute force trial of all possible keys

This could be improved by using a different key for every character $k_1, k_2, \ldots, k_N$

This obscures correlations between characters, making it uncrackable by frequency analysis

Also, each possible N character message has a corresponding key

$$\left.\begin{matrix}\text{"hi"}\\ k_1=3, k_2=5\end{matrix}\right\} \rightarrow \text{"jz"}$$

So which is "jz"?

$$\left.\begin{matrix}\text{"no"}\\ k_1=24, k_2=26\end{matrix}\right\} \rightarrow \text{"jz"}$$

This makes it uncrackable by brute force. Uncrackable by anything but knowing the key (as long as it is only used once)

This scheme is called the 'one time pad'

Cryptography and MI

When encrypting information, we have three variables

Message	M
Encrypted message	M'
Key	K

We want the message to be unreadable without the key

$$I(M;M')=0$$

But reveal all its information with the key

$$I(M;M'K)=H(M)$$

From the first condition

$$I(M,M') = H(M) + H(M') - H(M,M') := 0$$

$$\therefore \quad H(M,M') = H(M) + H(M')$$

Note that, given M and M', we should be able to deduce what key was used. So the composite variable MM' is equivalent to the composite variable MK. Since M and K are uncorrelated

$$H(M,M') = H(M) + H(K)$$

Combined with the above, this gives

$$H(M') = H(K)$$

Since M' is a more randomized version of M, clearly

$$H(M') \geq H(M) \quad \therefore \quad H(K) \geq H(M)$$

So, in order to obtain ideal cryptography, the key must contain at least as much entropy as the message

To encode each character of an N character message with entropy $H(M)$, a key must be drawn from a distribition with entropy

$$H(K) \geq H(M)$$

Also, key can only be used once: One Time Pad

Since this is uncrackable, only problem is key distribution: Quantum helps with this!