Information - ↓ uncertainty

Info content $\log_2 \frac{1}{P(seq)}$ in bits

Expected info content

$$H(X) = E(I(X)) = \sum_{i=1}^{n} p(x_i) \log_2\left(\frac{1}{p(x_i)}\right)$$

If all pi's = uniform, $\frac{1}{N}$

$\log_2(N)$

Outcome reduced to M possible choices

Entropy after reciept

$$\frac{1}{M} \log_2\left(\frac{1}{M}\right) = \log_2 M$$

Entropy in message is change

$$H_{before} - H_{after} = \log_2(N) - \log_2(M)$$

$$= \log_2\left(\frac{N}{M}\right)$$ original # choices

↑ # of choices left!

Example: Have 52 cards, tell you its a ♢

So $\log_2\left(\frac{52}{13}\right) = 2$ bits info

I gave you

↑ assuming all prob =

Or add $\log_2\left(\frac{1}{.2+.2+.2}\right)$ ↑ remaining probs

Additive

Fixed is eaiset

Variable saves space

- max down to entropy

Huffman coding - optimal

Compute avg lenght of code

$\sum P_{symbol}(L_{symbol})$

$p \approx \frac{1}{2^k}$

Log the power to which the base must be raised to to produce that #

$\log_2 16 \rightarrow 2^x = 16$

$\log(xy) = \log(x) + \log(y)$

$\log_b(x^p) = p \log_b x$

$e^{\ln(x)} = x \quad \ln(e^x) = x$

$\log_b\left(\frac{x}{y}\right) = \log_b(x) - \log_b(y)$

$\log_b(\sqrt[p]{x}) = \frac{\log_b(x)}{p}$

$\log_b(x) = \frac{\log_k(x)}{\log_k(b)}$

6.02 #1

Clock Recovery The constant adjust forward/backwards

- Sample middle one

Other Stuff

How many samples/bit

Where does byte start?

8b/10

1. Lots of bit transitions
2. DC balance 0s, 1s
3. Special sync symbol

$x[n]$ = input

$y[n]$ = output

$u[n]$ = unit step

$s[n]$ = unit step response

$\delta[n]$ = unit sample

$h[n]$ = unit sample response/channel description

fading coefficent

break everything down to unit samples

Time invarient $x[n-N] \rightarrow y[n-N]$

Linear $a x_1[n] + b x_2[n] \rightarrow a y_1[n] + b y_2[n]$

weighted sums in → out

allows deconvolution

Convolution

- commutative $x[n] \cdot h[n] = h[n] \cdot x[n]$
- associative

$x[n](h_1[n] \cdot h_2[n]) = (x[n] \cdot h_1[n]) h_2[n]$

- distributive

$x[n] \cdot (h_1[n] + h_2[n]) = x[n]h_1[n] + x[n]h_2[n]$

Parallel ⊕

Series ⊗

Just a product of what came before

Just clearest point - not where transition is

Causal - depends only on current + previous values

Scalar - real #, not vector

$u[n] = s[n] - s[n-1]$

3/3

ISI

$$B = \left[\frac{\text{lenght } h[n] \text{ active}}{N}\right] + 2$$

test pattern $2^N \cdot B$

pick samples/bit

diff than fast/slow channel

deconvolution

$$w[n] = \frac{1}{h[0]}(y[n] - w[n-1]h[1] + \cdots)$$

Stability

$$\sum_{m=1}^{k} \left|\frac{h[m]}{h[0]}\right| < 1 \quad \sum_{m=1}^{k} h[m] < h[0]$$

drop first $h[0]$ if is or close to 0

Noise

Mean $\mu_x = \frac{1}{N}\sum_{n=1}^{N} x[n]$

$$P_x = \frac{1}{N}\sum_{n=1}^{N} x[n]^2 \quad \tilde{P}_x = \frac{1}{N}\sum_{n=1}^{N}(x[n] - \mu_x)^2$$

$$E_x = \sum_{n=1}^{N} x[n]^2 \quad \tilde{E}_x = \sum_{n=1}^{N}(x[n] - \mu_x)^2$$

$$SNR = \frac{\tilde{P}_{signal}}{\tilde{P}_{Noise}} \quad SNR(db) = 10\log\left(\frac{\tilde{P}_{signal}}{\tilde{P}_{noise}}\right)$$

stationary vs ergodic random processes

$$P(x_1 \le X \le x_2) = \int_{x_1}^{x_2} f_x(x)\,dx$$

$$\mu_x = \int_{-\infty}^{\infty} x f_x(x)\,dx$$

$$\sigma^2 = \int_{-\infty}^{\infty} (x - \mu_x)^2 f_x(x)\,dx$$

$$\text{PDF normal} = \frac{1}{\sqrt{2\pi\sigma^2}} e^{n\left(\frac{-(x-\mu)^2}{2\sigma^2}\right)}$$

$$\text{CDF} - \text{erf}(x) = \frac{2}{\sqrt{\pi}}\int_0^x e^{-t^2}\,dt$$

$$\phi_{\mu,\sigma}(x) = \phi\left(\frac{x-\mu}{\sigma}\right)$$

BER bit error ratio

$$\mu_{Ynf} = \frac{1}{N}\sum_{n=1}^{N} Y_{nf}[n] = \frac{1}{N}\frac{N}{2} = \frac{1}{2}$$

$$\tilde{P}_{Ynf} = \frac{1}{N}\sum_{n=1}^{N}\left(Y_{nf}[n] - \frac{1}{2}\right)^2 =$$

$$= \frac{1}{N}\sum_{n=1}^{N}\left(\frac{1}{2}\right)^2 = \frac{1}{N}\frac{N}{4} = \frac{1}{4}$$

$$P(\text{error}) = P(0)\cdot P(\text{error}\mid \text{trans}0) + P(1)P(\text{error}\mid 1)$$
$$= .5\cdot\phi(-.5/\sigma) + .5\cdot\phi(-.5/\sigma)$$
$$= \phi(-.5/\sigma)$$

$$SNR(db) = 10\log\left(\frac{\tilde{P}_{signal}}{\tilde{P}_{noise}}\right) = 10\log\left(\frac{.25}{\sigma^2}\right)$$

Eye diagram

all possible voltage sequences in a certain # of bits

If have a channel that recieves more 1s make it more likely to recieve a 1.
Find ϕ w/ test signals

$$B = \left[\frac{M}{N}\right] + 1$$

Deconvolution w is estimated x

$$y[n] = h[0]w[n] + h[1]w[n-1] + \dots$$

$$w[n] = \frac{y[n] - (h[1]w[n-1] + h[2]w[n-2] + \dots)}{h[0]}$$

$$w[0] = \frac{y[0]}{h[0]}$$

$$w[1] = \frac{y[1] - h[1]w[n-1]}{h[0]}$$

$$w[2] = \frac{y[2] - (h[1]w[n-1] + h[2]w[n-2])}{h[0]}$$

Ce.02 Exam 1 study

Cheatsheet!

Make that instead

Worse case - just do the problem at!

Huffman lenght: not new # items - 1

Oh that was wrong in class anyway

→ Hamming distance

Bj example $\log_2\left(\frac{N}{M}\right)$ ← # choices left (52-3=49)

LZW I think I know

- or should I do a practice??

Just do psydo code they give

Im skipping JPEG

- we never revieved

What is least significant vs most sig bit first?

Not writing convolution math formula

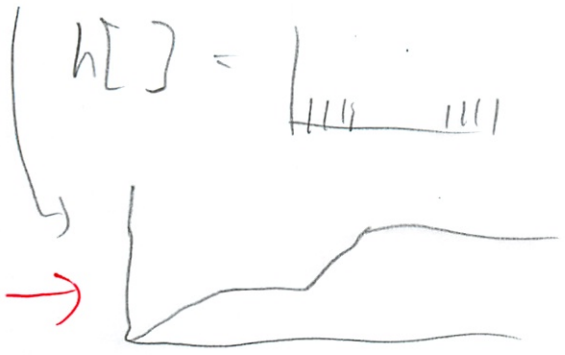
What is ~ again?

Causal?

Is prob on test?

(2) a)

Unit step response



derivative-ish

b) Which approch same result as $n \to \infty$

→ - need to add up all

→ - can't do by pic

(I had totally the wrong intuition)

c) Back at h

- what was the intuition here

No that it is not it

Just below

No can do math wise

Oh forgot that they gave channels

Notice is it increasing a lot

$y[15] = h[0]x[15] + h[1]x[14] + \dots + h[15]x[0]$

$x[15] - x[1]$ is 0 - rest are 1

Just add up h

③

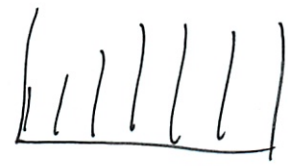
eye which

✓ got

13 What is y[24]

0 0 1 1 0 1 0

4 samples per bit

h[ ] = 

↑ never goes away ↑

- no is unit step response

4 8 16

12 16 20 24

right after 1

So 1 ⊗ .8

$x[24]h[0] + x[23]h[1] + x[22]h[2] + x[21]h[3]$

$0 \cdot .2 + .1 \cdot .2 + 1 \cdot .3 + 1 \cdot .3$

↑ This was 0

messed me up

↑ did not see was larger

Oh start at 0, I started at 1

When counting

B Input to deconvolver

$$w_A[n] = \frac{1}{\alpha}\left[y[n] - \left(.2\,w_A[n-1] + .3\,w_A[n-2] + .3\,w_A[n-3]\right)\right.$$

$w_a[n] = 0$ for $n < 0$

for what value of α will $w_a[n] = x[n]$

(4)

"How do you do this"

Use convolution sum and h[n] from above

$y[n] = x[n] \cdot .2 + x[n-1] \cdot .2 + x[n-2] \cdot .3 + x[n-3] \cdot .3$

Reorg for deconvolver

$y[n] = w_a[n] \cdot \alpha + w_a[n-1] \cdot .2 + w_a[n-2] \cdot .3 + w_a[n-3] \cdot .3$

If want $w_a[n] = x[n]$

then α must = .2

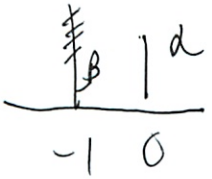
---

2. Result in terms of unit sample

$\delta[n+1] + 4\delta[n] + 8\delta[n-1]$

Watch the negitive time values

4b What is h

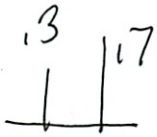


$\alpha > \beta$ ✓ 0 for all other values

(5)

c) another convolution one

1 0 0 1 1 0 1 1 1 1 1 1 →



.7

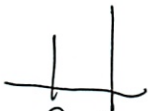
.3

0 .7

1 .3 .7 1 1 1 1 → ✓

d) Derive a deconvolver

y[n] = .7 1 1 .3 .7 1 .3 0 0 0 0



Oh I can see what to do

1 1 1 0 1 1 0 0 0 0 0 → ✓

$w[n] = \frac{1}{h[0]}\left(y[n] - h[1]w[n-1]\right)$ ← what does this mean?

$= \frac{y[n]}{.7} - \frac{.3}{.7} w[n-1]$

(6)

W is an estimate of X

$H \cdot W = Y$

$$y[n] = \sum_{m=0}^{m=n} w[n-m]\,h[m]$$

(just a reverse of $y[n] = \sum_{m=0}^{m=n} x[m]\,h[n-m]$

$y[n] = h[0]\,w[n] + h[1]\,w[n-1] + h[2]\,w[n-2] + \cdots$

(So just wrote w for x where we flip in ts/ide

Then reverse for w[n]

w[0] = previously calced w[]

$$w[n] = \frac{y[n] - (h[1]\,w[n-1] + h[2]\,w[n-2] + \cdots)}{h[0]}$$

So back to our problem

$y[0] = h[0]\,x[0]$

$.7 = .7 \quad 1$

So backwards

$x[0]$ (called $w[0]$) $= \frac{y[0]}{h[0]}$

$= \frac{.7}{.7} = 1$

(7)

$$y[1] = .7 + h[1]\,x[0]$$

$$1 = .7 + .3 \cdot 1$$

So

$$w[0] = \frac{y[1] - .7 \cdot 1}{h[1]} \qquad h[1]\,w[n-1]$$

e: always $h[0]$

$$= \frac{1 - .7}{.3}$$

$$= 1 \quad (1)$$

$$w[2] = \frac{y[2] - .3 - .7}{h[0]}$$

$$w[2] = \frac{y[2] - (h[1]\,w[n-1] + h[2]\,w[n-2])}{h[0]}$$

$h[2]=0$

$$= \frac{1 - (.7 \cdot 1 + .3 \cdot 1)}{.7}$$

watch the formulas!

$$= 1$$

~~Oh I was taking wrong input so Opps~~

or was I? - no not quite right

(8)

6. Clever question

- have h[ ]

0

.05 + .09 + .13 ← got the addative here

but forgot h[0] = 0

so 0 + .5 + .9 = 12

.05(1) + .09(1) + .13(1) + .20(1) +

↑

When it changes

What should I do?

? still climing from inital and subtract

+ .13 + .13 + .13 ~~...~~ + .05

− .05 − .09 − .13 − .20 − ~~.13 .13~~

← actually right

~~from~~ or 10

but looking for 3

First have looked at how many bits interfere

I implicitly did this

Sample 8 is 3rd bit

↳ Bits start 0, 5, 10

↳ so third sample of bit 3

$y[n] = \sum_k h[k]\, x[n-k]$

But y[12] ← first such value where third sample of third bit time

?? why ? to get it started?

(9)

n	0	1	2	3	4	5	6	7	8	9
h[n]	0	.04	.12	.24	.40	.60	.72	.84	.96	1

Oh right!

Its additive!!!!!!

↙ that just means slower rise not decrease

actually was I right; I did add

Why don't I get that?

Now consider all possible values of current + future bit

Use superposition (add on top of each other)

111 $v[n]$ $y[12] = h[12] = 1$

110 $v[n] - v[n-10]$ $y[12] = h[12] - h[3] = 1 - .12 = .88$

↑ so it is add it up + subtract

101 $v[n] - v[n-5] + v[n-10] = 1 - .84 + .12$

↑ so from first ↑ 2nd ↑ 3rd

: use their way in future

(10)

7. Same as previous problem, but just one line

Remember lowest curve is 101

So convolve

I see

8. Just do the convolution

Hamming Code

bit error correction

"outside of scope"

---

9. * for unit step response look at difference for h

↑ for responses

back at changes

Only have to look at past responses

Know which step you are at

Unit sample = Unit step − Unit step

$h[n] = s[n] - s[n-1]$

"do they want for whole pulse"

(4)

11. To find $h[\,]$ from deconvolve linear eq

Write in terms of $y[\,]$

and then scaling factor is $h[\,]$

Hamming distance

Strings = lenght

# of positions where stings are different

i.e. errors

of

Information - ↓ uncertainty

Info content $\log_2 \frac{1}{P(seq)}$ in bits

Expected info content

$$H(X) = E(I(X)) = \sum_{i=1}^{n} p(x_i) \log_2\left(\frac{1}{p(x_i)}\right)$$

If all pi's = uniform, $\frac{1}{N}$

$\log_2(N)$

Outcome reduced to M possible choices

Entropy after reciept

$$\frac{1}{M} \log_2\left(\frac{1}{M}\right) = \log_2 M$$

Entropy in message is change

$$H_{before} - H_{after} = \log_2(N) - \log_2(M)$$

$$= \log_2\left(\frac{N}{M}\right)$$

original # choices

# of choices left!

Example: Have 52 cards, tell you its a ♢

So $\log_2\left(\frac{52}{13}\right) = 2$ bits info

I gave you

assuming all prob =

Or add $\log_2\left(\frac{1}{.2+.2+.2}\right)$

remaining probs

Additive

Fixed is eaiset

Variable saves space

- max down to entropy

Huffman coding - optimal

Compute avg lenght of code

$\sum P_{symbol}(L_{sympel})$

$p \approx \frac{1}{2^k}$

Log the power to which the base must be raised to to produce that #

$\log_2 16 \rightarrow 2^x = 16$

$\log(xy) = \log(x) + \log(y)$

$\log_b(x^p) = p \log_b x$

$e^{\ln(x)} = x \quad \ln(e^x) = x$

$\log_b\left(\frac{x}{y}\right) = \log_b(x) - \log_b(y)$

$\log_b(\sqrt[p]{x}) = \frac{\log_b(x)}{p}$

$\log_b(x) = \frac{\log_k(x)}{\log_k(b)}$

6.02 #1

Clock Recovery The constant adjust forward/backwards

- Sample middle one

Other Stuff

How many samples/bit

Where does byte start?

8b/10

1. Lots of bit transitions
2. DC balance 0s, 1s
3. Special sync symbol

$x[n]$ = input

$y[n]$ = output

$u[n]$ = unit step

$s[n]$ = unit step response

$\delta[n]$ = unit sample

$h[n]$ = unit sample response/channel description

fading coefficent

break everything down to unit samples

Time invarient $x[n-N] \rightarrow y[n-N]$

Linear $a x_1[n] + b x_2[n] \rightarrow a y_1[n] + b y_2[n]$

weighted sums in → out

allows deconvolution

Convolution

- commutative $x[n] \cdot h[n] = h[n] \cdot x[n]$
- associative

$x[n](h_1[n] \cdot h_2[n]) = (x[n] \cdot h_1[n]) h_2[n]$

- distributive

$x[n] \cdot (h_1[n] + h_2[n]) = x[n] h_1[n] + x[n] h_2[n]$

Parallel ⊕

Series ⊗

Just a product of what came before

Just clearest point - not where transition is

Causal - depends only on current + previous values

Scalar - real #, not vector

$u[n] = s[n] - s[n-1]$

ISI

$$B = \left[\frac{\text{lenght } h[n] \text{ active}}{N}\right] + 2$$

test pattern $2^N \cdot B$

pick samples/bit

diff than fast/slow channel

deconvolution

$$w[n] = \frac{1}{h[0]}(y[n] - w[n-1]h[1] + \cdots)$$

Stability

$$\sum_{m=1}^{k} \left|\frac{h[m]}{h[0]}\right| < 1 \quad \sum_{m=1}^{k} h[m] < h[0]$$

drop first $h[0]$ if is or close to 0

Noise

Mean $\mu_x = \frac{1}{N} \sum_{n=1}^{N} x[n]$

$$P_x = \frac{1}{N} \sum_{n=1}^{N} x[n]^2 \quad \tilde{P}_x = \frac{1}{N} \sum_{n=1}^{N} (x[n] - \mu_x)^2$$

$$E_x = \sum_{n=1}^{N} x[n]^2 \quad \tilde{E}_x = \sum_{n=1}^{N} (x[n] - \mu_x)^2$$

$$SNR = \frac{\tilde{P}_{signal}}{\tilde{P}_{Noise}} \quad SNR(db) = 10 \log\left(\frac{P_{signal}}{P_{noise}}\right)$$

stationary vs ergodic random processes

$$P(x_1 \le X \le x_2) = \int_{x_1}^{x_2} f_x(x)\,dx$$

$$\mu_x = \int_{-\infty}^{\infty} x f_x(x)\,dx$$

$$\sigma^2 = \int_{-\infty}^{\infty} (x - \mu_x)^2 f_x(x)\,dx$$

$$\text{PDF normal} = \frac{1}{\sqrt{2\pi\sigma^2}} e^{\wedge}\left(\frac{-(x-\mu)^2}{2\sigma^2}\right)$$

$$\text{CDF} = \text{erf}(x) = \frac{2}{\sqrt{\pi}} \int_0^x e^{-t^2}\,dt$$

$$\phi_{\mu,\sigma}(x) = \phi\left(\frac{x-\mu}{\sigma}\right)$$

BER bit error ratio

$$\mu_{Y_{NE}} = \frac{1}{N}\sum_{n=1}^{N} Y_{NE}[n] = \frac{1}{N}\frac{N}{2} = \frac{1}{2}$$

$$\tilde{P}_{Y_{NE}} = \frac{1}{N}\sum_{n=1}^{N}\left(Y_{NE}[n] - \frac{1}{2}\right)^2 =$$

$$= \frac{1}{N}\sum_{n=1}^{N}\left(\frac{1}{2}\right)^2 = \frac{1}{N}\frac{N}{4} = \frac{1}{4}$$

$$P(\text{error}) = P(0)\cdot P(\text{error}\mid \text{trans }0) + P(1)P(\text{error}\mid 1)$$
$$= .5\cdot\Phi(-.5/\sigma) + .5\cdot\Phi(-.5/\sigma)$$
$$= \Phi(-.5/\sigma)$$

$$SNR(db) = 10\log\left(\frac{\tilde{P}_{signal}}{\tilde{P}_{noise}}\right) = 10\log\left(\frac{.25}{\sigma^2}\right)$$

Eye diagram

all possible voltage sequences in a certain # of bits

If have a channel that recieves more 1s make it more likely to recieve a 1

Find Φ w/ test signals

$$B = \left[\frac{M}{N}\right] + 1$$

Deconvolution w is estimated x

$$y[n] = h[0]w[n] + h[1]w[n-1] + \dots$$

$$w[n] = \frac{y[n] - (h[1]w[n-1] + h[2]w[n-2] + \dots)}{h[0]}$$

$$w[0] = \frac{y[0]}{h[0]}$$

$$w[1] = \frac{y[1] - h[1]w[n-1]}{h[0]}$$

$$w[2] = \frac{y[2] - (h[1]w[n-1] + h[2]w[n-2])}{h[0]}$$

Encode LZW

initalize TABLE[0 to 255] = code for individual bytes

STRING = get input signal

while there are still input symbols:

  SYMBOL = get input symbol

  if STRING + SYMBOL is in TABLE

    STRING = STRING + SYMBOL

  else:

    output the code for STRING

    add STRING + SYMBOL to TABLE

    STRING = SYMBOL

output the code for STRING

Decode LZW

initalize TABLE[0 to 255] = code for individual bytes

CODE = read next code from encoder

STRING = TABLE[CODE]

output STRING

while there are still codes to recieve

  CODE = read next code from encoder

  if TABLE[CODE] is not defined:

    ENTRY = STRING + STRING[0]

  else:

    ENTRY = TABLE[CODE]

  output ENTRY

  add STRING + ENTRY[0] to TABLE

  STRING = ENTRY