Mutual information on metric spaces.

Conor Houghton

CS, U Bristol

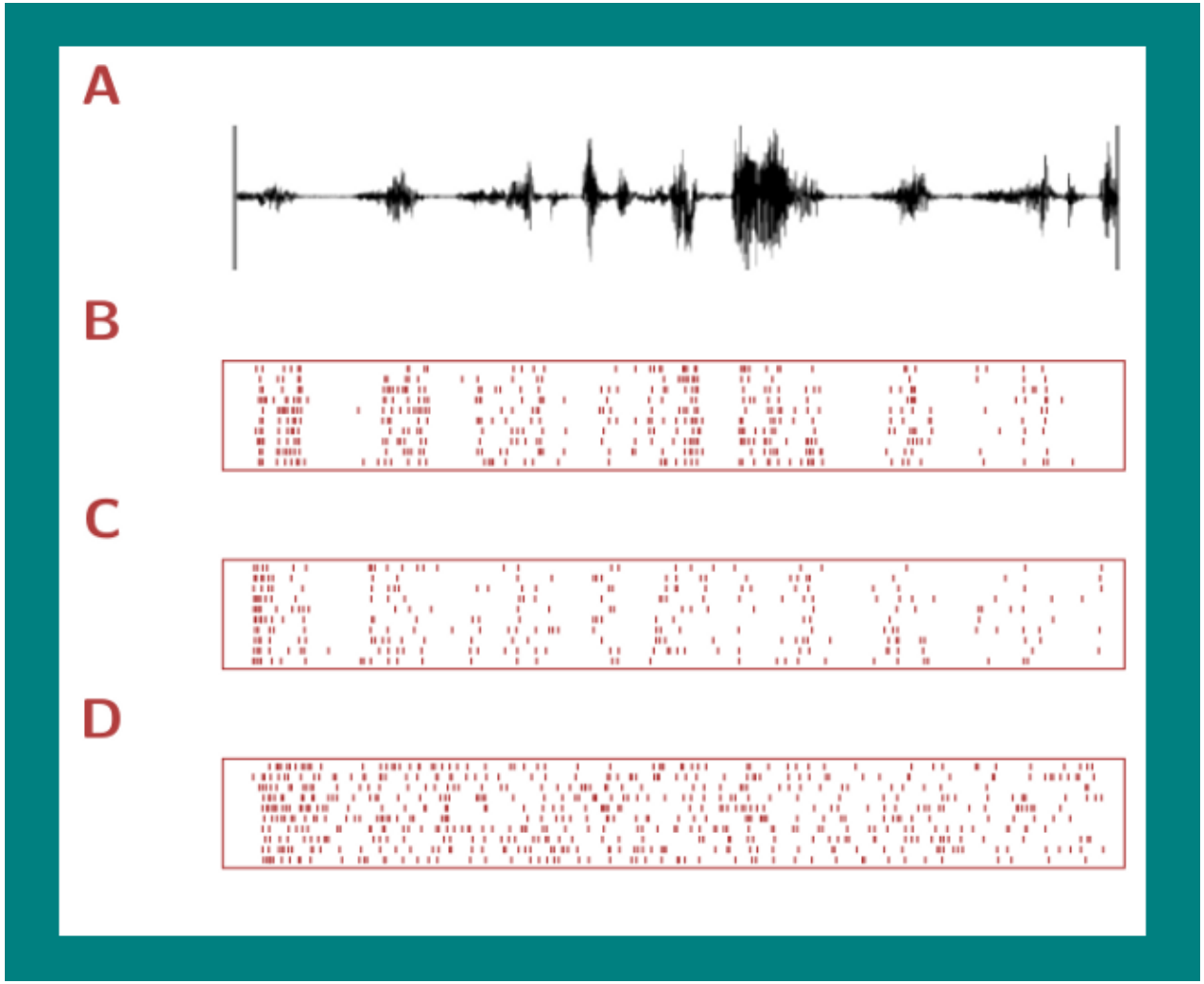
Psychology, U Bristol, 2019

Outline

This talk will be about estimating mutual information on metic spaces.

- I became interested in this topic because of the application to spike trains.
- I believe that the approach outlined here has applications elsewhere, perhaps in machine learning.

Spike trains



Shannon's entropy

$$H(X) = -\sum_x p(x) \log_2 p(x)$$

Shannon's entropy

$$H(X) = -\sum_x p(x) \log_2 p(x)$$

A	B	C	D	E	F	G	H
1/2	1/4	1/8	1/16	1/32	1/64	1/128	1/128
000	001	010	011	100	101	110	111
0	10	110	1110	11110	111110	1111110	1111111

$$\text{average code length} = \frac{1}{2} + \frac{1}{4}2 + \frac{1}{8}3 + \frac{1}{16}4 + \ldots = H(X) \approx 1.98 < 3$$

Shannon's entropy

Shannon's entropy measure's 'interestingness' rather than information.

Conditional entropy

$$H(X|S=s) = -\sum_x p(x|s)\log_2 p(x|s)$$

and

$$H(X|S) = H(X|S) = \langle H(X|s)\rangle_s$$

If X is the stimulus $H(X|S)$ quantifies noise.

Differential entropy is tricky

If $\mathcal{X} = \{\text{orange}, \text{apple}, \text{bunch of graphs}, \text{pear}, \text{some bananas}\}$ then

$$H(X) = \log 5 = 1.6$$

but if we count the grapes and bananas individually

$$H(X) = \frac{20}{25} \log \frac{25}{20} + \frac{2}{25} \log \frac{25}{2} + \frac{3}{25} \log 25 = 1.11$$

Picture from, you know, the internet.

Mutual information

$$I(X,Y) = \sum_{x,y} p_{X,Y}(x,y) \log_2 \frac{p_{X,Y}(x,y)}{p_X(x)p_Y(y)}$$

Mutual information

$$I(X,Y) = \sum_{x,y} p_{X,Y}(x,y) \log_2 \frac{p_{X,Y}(x,y)}{p_X(x)p_Y(y)}$$

gives

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

Mutual information

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

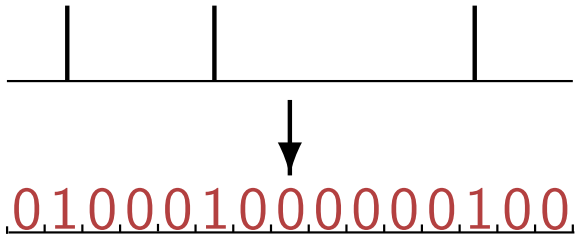
so, for example

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

info in Y=(info remaining in Y if you know X)+(mutual info)

Classical approach

- Discretize.



- Split into words.

$$010001000000100 \to 01000, 10000, 00100$$

Bialek, de Ruyer van Steveninck, Strong and other coworkers, late 1990s.

Classical approach

- Estimate probability of words. For example, say $w_8 = 01000$ then estimate

$$p(w_8) \approx \frac{\#\text{ occurrences of } w_8}{\#\text{ words}}$$

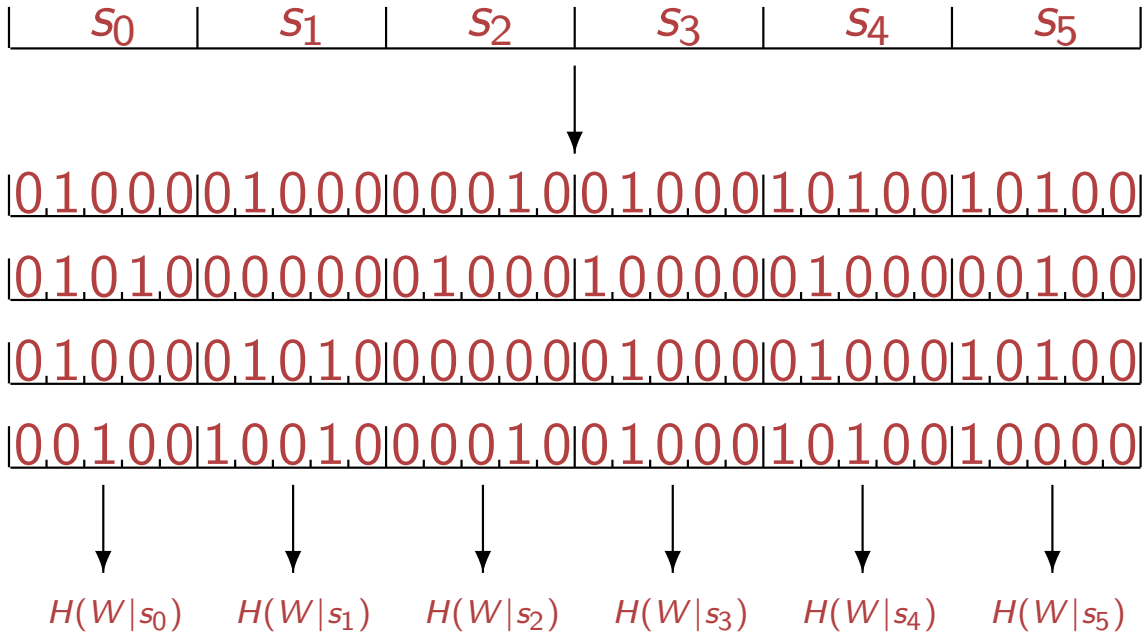
- Calculate

$$H(W) = -\sum_i p(w_i) \log_2 p(w_i) = -\langle \log_2 p(w_i) \rangle$$

Bialek, de Ruyer van Steveninck, Strong and other coworkers, late 1990s.

Classical approach

- Conditional probability.



Bialek, de Ruyer van Steveninck, Strong and other coworkers, late 1990s.

Classical approach

- Mutual information

$$H(W|S) = \langle H(W|s_i) \rangle$$

and

$$I(W;S) = H(W) - H(W|S)$$

Bialek, de Ruyer van Steveninck, Strong and other coworkers, late 1990s.

ms scale information in blow fly spike trains.

Bialek, de Ruyer van Steveninck, Strong and other coworkers, late 1990s.

Difficulties with the classical approach.

- Undersampling.
  - 100 ms words and 2 ms bins gives $2^{50} = 1125899906842624$ words.
  - Lots of clever approaches to this, for example Nemenman et al. (PRE 2004, BMC Neuroscience 2007) where a cunning prior is used for $p(w_i)$.
- Sampling bias.
  - An even distribution will never give equal counts for each word, giving different $p(w_i)$.
  - Lots of clever approaches to this too, see Panzeri et al. (J Neurophys. 2007).

Many fixes but still . . .

- Neuron - neuron mutual information.
- Maze - neuron mutual information.
- Mutual information with multiple units.

A Kozachenko-Leonenko estimator.

Here we will use a Kozachenko-Leonenko estimator.

Kraskov, Stöbauer and Grassberger (PRE 2004)

A dart board

photo from ebay (£4.20 +p.p.)

Probability mass function

$p(\mathbf{x})$ is the mass function

Probability mass function

$$\text{prob}(\text{dart lands in } B) = \int_B p(\mathbf{x}) dV$$

Estimating using the number of holes

$$\langle \text{number of holes in } B \rangle \approx \int_B p(\mathbf{x}) dV \times (\text{total number of holes})$$

where the total volume is normalized.

Estimating the probability mass function

If the mass function varies slowly:

$$\int_B p(\mathbf{x})dV \approx p(\mathbf{x}_0) \times \text{vol}\, B$$

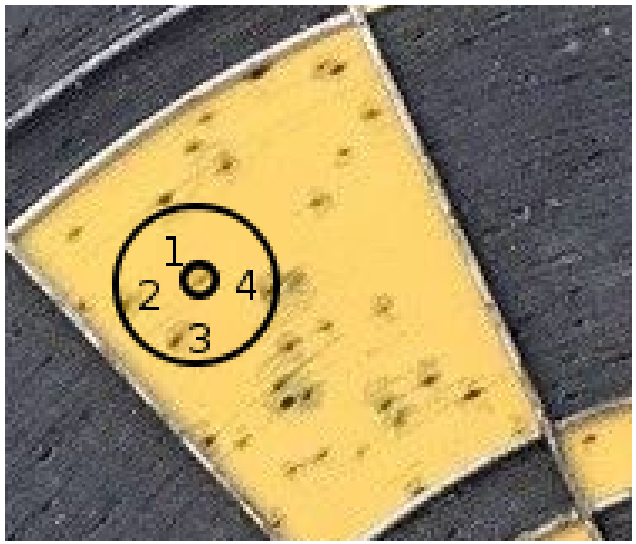
where $\mathbf{x}_0$ is a point, for example, in the middle of B where we are interested in. Now

$$\text{number of holes in } B \approx p(\mathbf{x}_0) \times \text{vol}\, B \times (\text{total number of holes})$$

Estimating using the number of holes

$$p(x_0) \approx \frac{\#B}{n \times \text{vol}\, B}$$

where n is the total number of points and $\#B$ is the number of points in B.



so

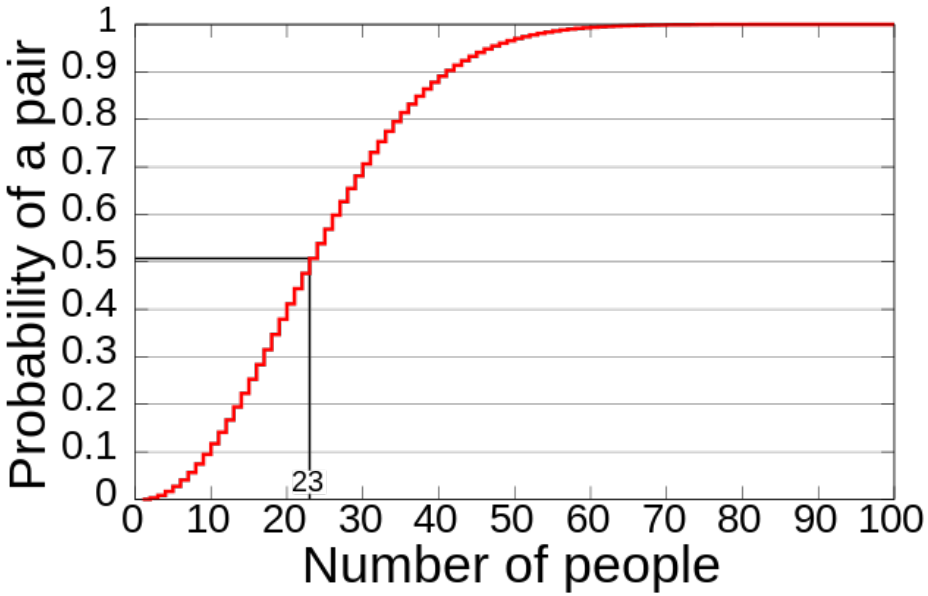
$$p(\circ) = \frac{4}{n\text{vol}\, B}$$

Estimating using the number of holes

$$p(x_0) \approx \frac{\#B}{n \times \text{vol}\, B}$$

Using this to find the mutual information gives a *Kozachenko-Leonenko* estimator.

Kozachenko-Leonenko estimators are very good



graph from wikipedia article on the birthday paradox

Problem

How do we work out the volume in the space of functions? We have no coordinates xyz to do

$$\text{vol}\, B = \int_B dxdydz$$

Problem

How do we work out the volume in the space of functions? We have no coordinates xyz to do

$$\text{vol}\, B = \int_B dxdydz$$

Even if we had coordinates would we want to use them?

Use the mass function as a measure!

$$\text{vol}\, B = \int_B p(\mathbf{x}) dV$$

A volume:

- $A \subset B$ implies $\text{vol}\, A < \text{vol}\, B$.
- $A \cap B = \phi$ implies $\text{vol}\, A \cup B < \text{vol}\, A + \text{vol}\, B$.

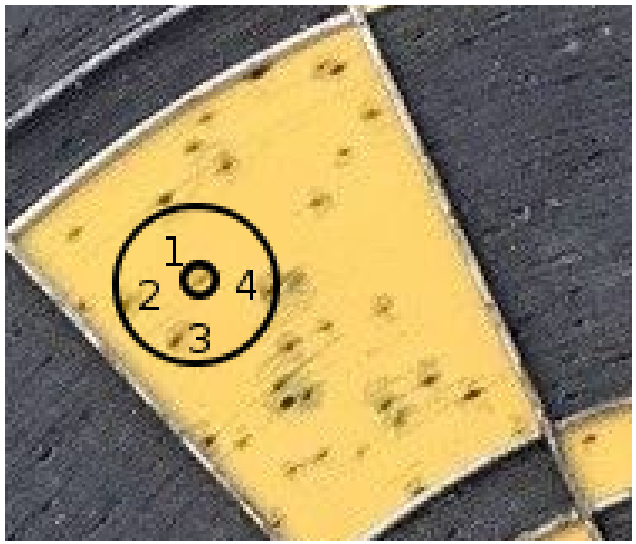
Use the mass function as a measure!

$$\text{vol}\, B = \int_B p(\mathbf{x}) dV$$

Volume by counting holes

$$\text{vol}\, B = \int_B p(\mathbf{x}) dV \approx \frac{\text{number of holes in } B}{\text{total number of holes}}$$

Volume by counting holes



A ball with volume h/n around the circled point, where n is the total number of holes and $h = 4$.

Metric

To make a ball you need a metric; not to measure the radius since the size is being defined by the volume, but to define 'the nearest h points'.

Oh no

$$p(\mathbf{x}_0) \approx \frac{\#B}{n \times \mathrm{vol}\, B} = \frac{h}{nh/n} = 1$$

and using this meaure gives $H(X) = 0$; in fact the differential entropy is not well-defined. However the mutual information is!

Mutual infomation

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

has two probability distributions: $p_Y(y)$ and $p_{Y|X}(y|x)$.

IDEA: use one to estimate volume, the other can then be estimated by counting!

Formula

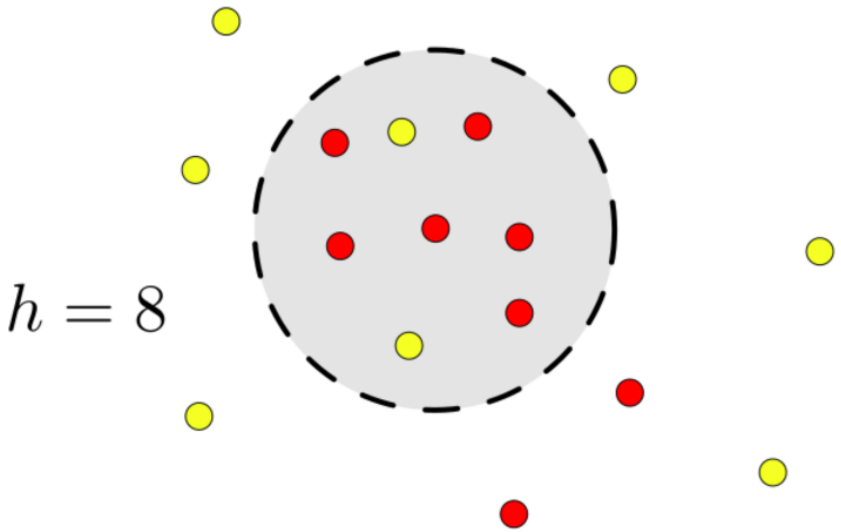
This is for the case where X is a discrete random variable and everything exciting is happening in Y space.

$$I(X,Y) = \frac{1}{n}\sum_{y_i} \log_2 \frac{n_s \# B_{y_i}(y_i)}{h}$$

where $B(y_i)$ is the ball around y and $\#_y B(y)$ is the number of points in that correspond to the same X value as y. n_s is the number of stimuli.

Formula

$$I(X,Y) = \frac{1}{n}\sum_{y_i} \log_2 \frac{n_s \#_{y_i} B(y_i)}{h}$$



h

There are two approximations:

$$\int_B p(x)dV \approx \#B \times \text{vol}\, B$$

and

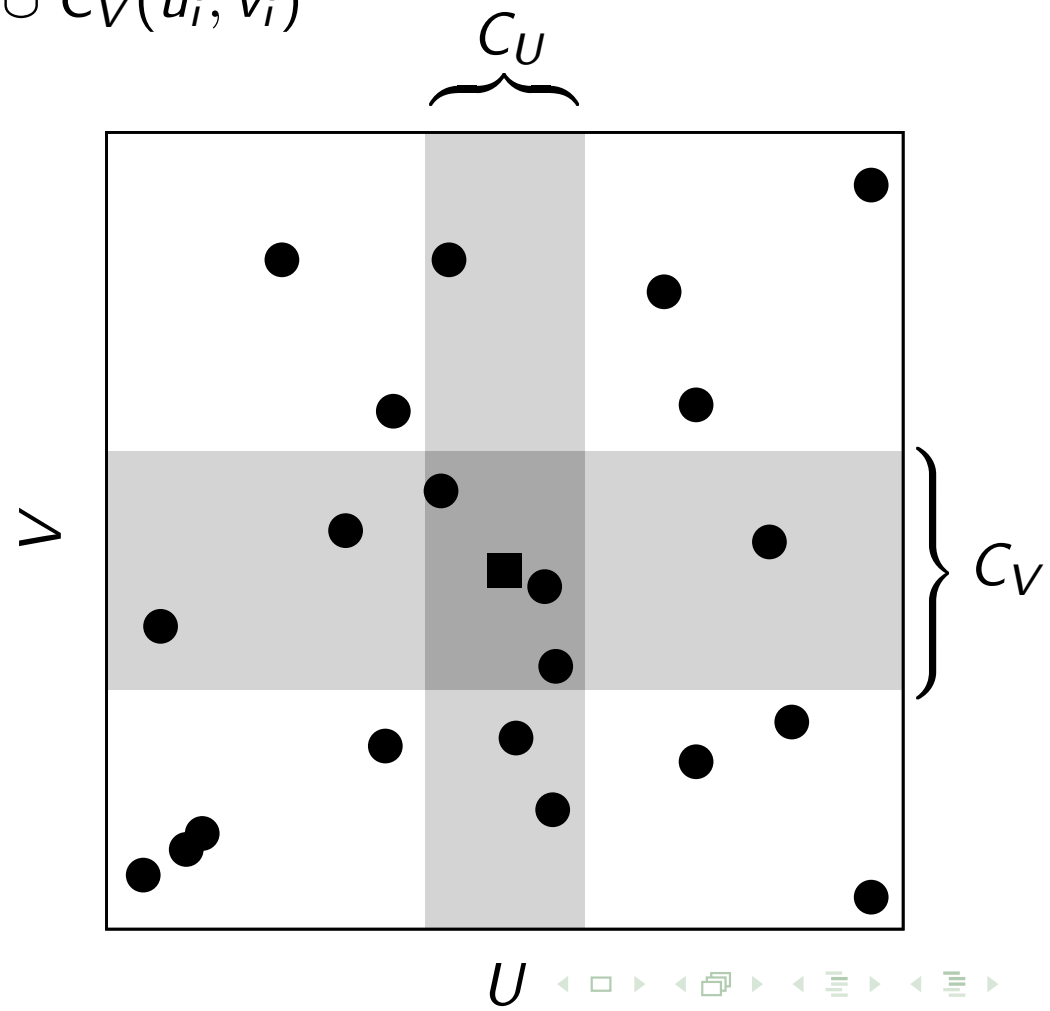
$$\int_B p(x)dV \approx \mathrm{V} \times p(x_0)$$

The first approximation gets better if the volume is bigger, the second gets worse; the correct choice of h is a compromize between these two. There is actually an approach to picking h that seems to work, based on the bias.

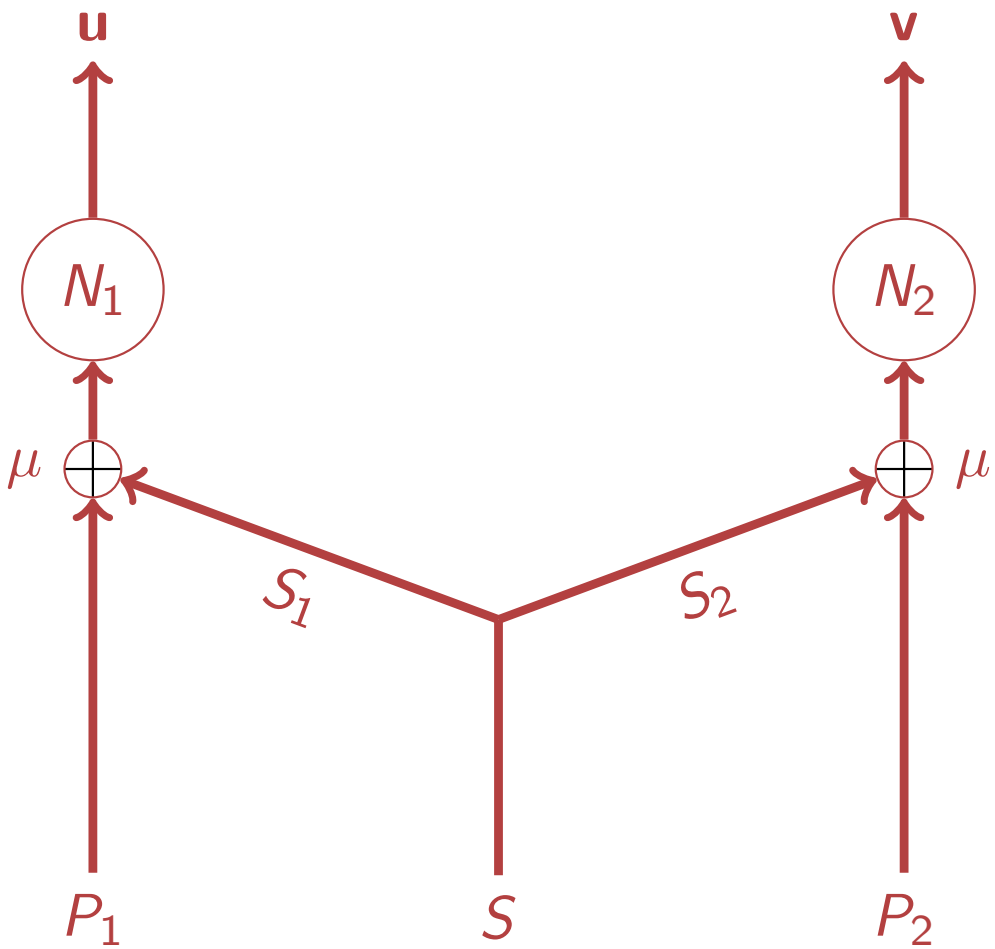
Another formula

$$I(X,Y) = \frac{1}{n}\sum_{i=1}^{n} \log_2 \frac{n\#[C(u_i,v_i)]}{h^2}$$

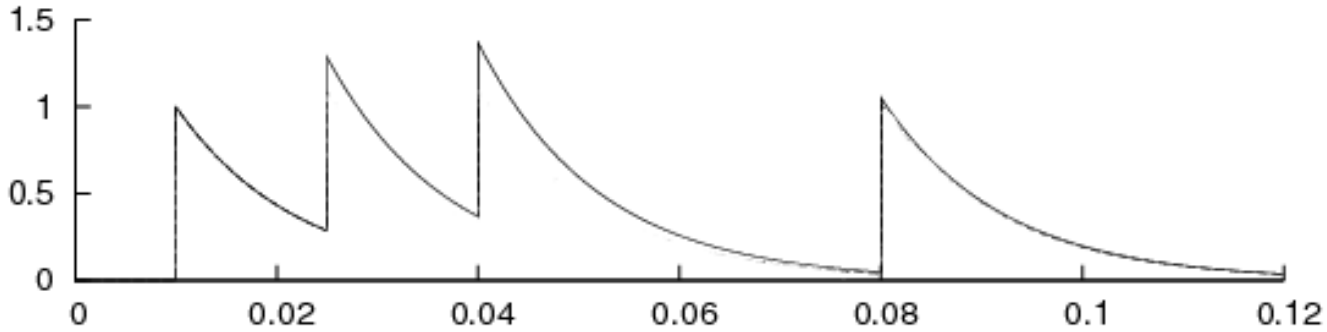
with $C(u_i,v_i) = C_U(u_i,v_i) \cup C_V(u_i,v_i)$



Fictive data



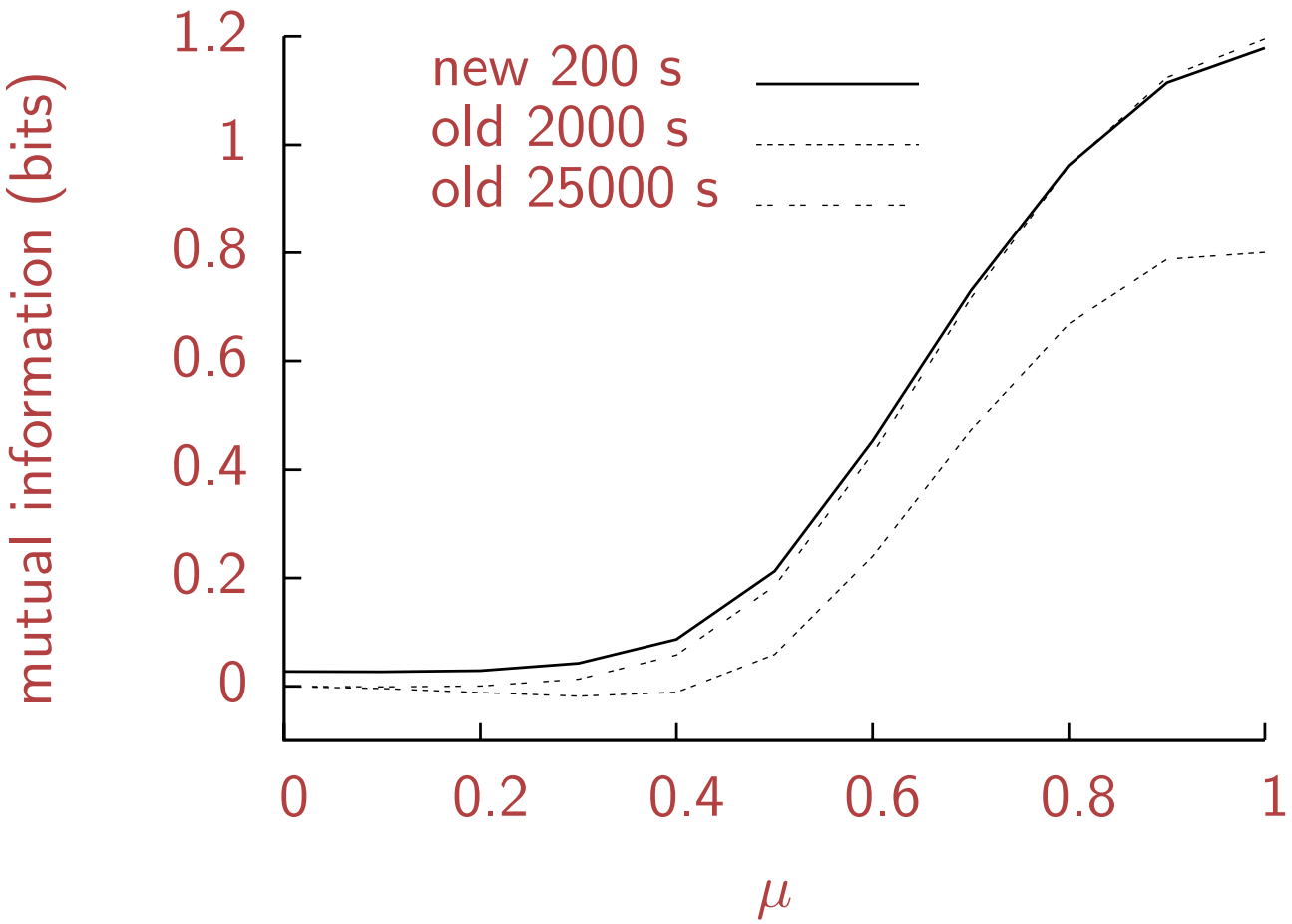
van Rossum metric



Spike trains mapped to functions and a metric on the space of functions induces a metric on the spike train space.

van Rossum (Neural Comp. 2001)

Result


mutual information (bits)
1.2
1
0.8
0.6
0.4
0.2
0
new 200 s
old 2000 s
old 25000 s
0
0.2
0.4
0.6
0.8
1
μ

Conclusion

- It works remarkably well!
- Does it work for functions?
- Does it do clustering?
- Does it do ICA?
- Are there applications to machine learning and data science?

The end

References:

- *A kernel-based calculation of information on a metric space.* R. Joshua Tobin and Conor J. Houghton, Entropy 15 (2013) 4540-4552.
- *Calculating mutual information for spike trains and other data with distances but no coordinates.* Conor Houghton, Royal Society Open Science, 2 (2015) 140391.
- *Calculating the mutual information between two spike trains.* Conor Houghton, Neural Computation (2019) 31:330-343.

Funding: the James S McDonnell Foundation.

THANK YOU!