Title: Informal Discussion of Eternal Inflation and String Theory

Date: Feb 01, 2005 02:00 PM

URL: http://pirsa.org/05020001

Abstract:

common problem in all sciences:

measure 2 quantities A, B

⇒ are they related ?

⇒ how can one quantify degree ?

temperature at town X }
rainfall — " — } ✓

temperature at Waterloo }
chinese stock market index } ?

# births }
# of breeding storks } ?

consumption of drug }
health state } ??

Applications:

clustering of objects

decomposition of signals into
~ independent components

classification

⋮

clustering of objects

decomposition of signals into
~ independent components

classification

⋮

clustering of objects

decomposition of signals into
~ independent components

classification

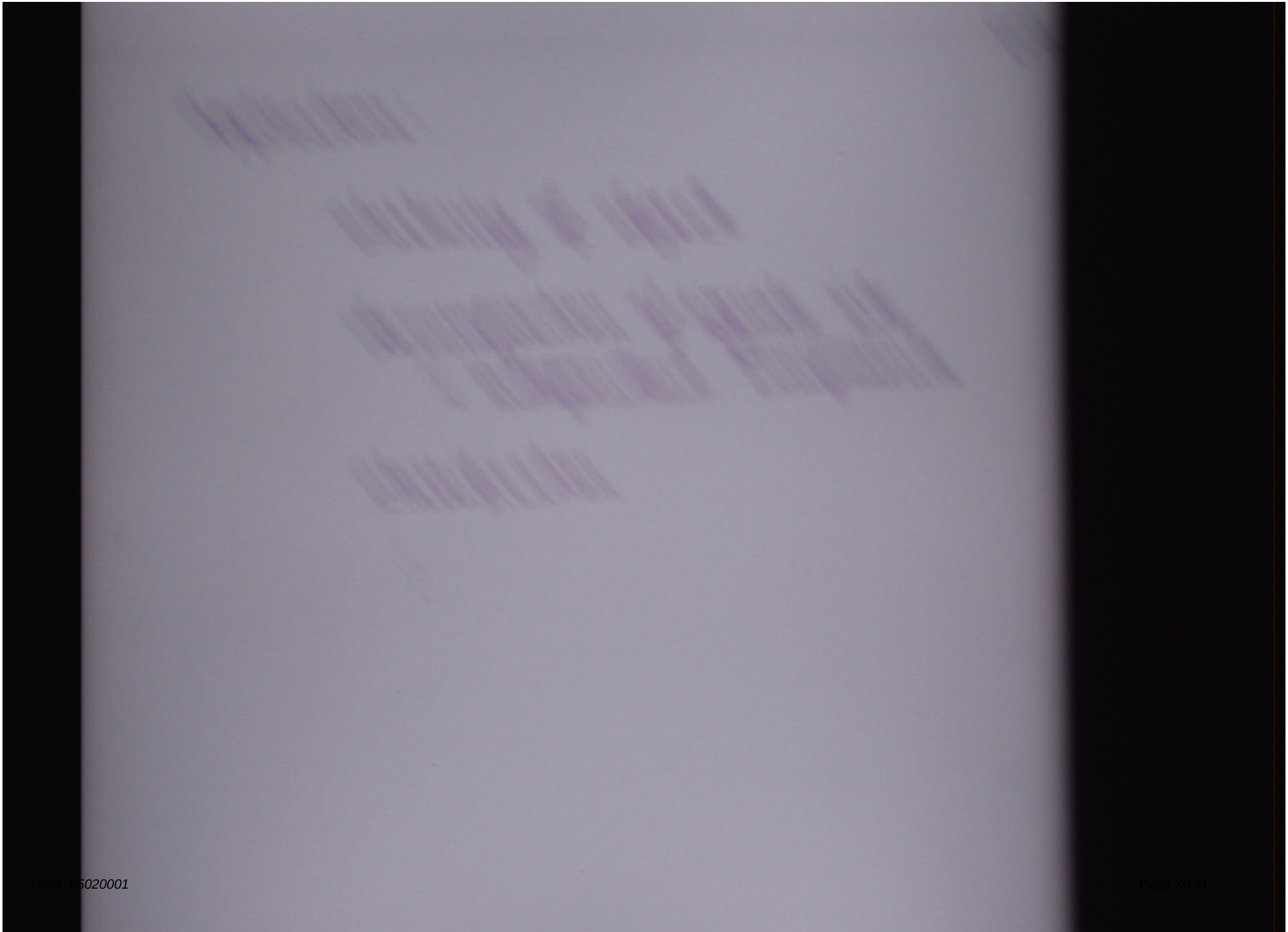
⋮

clustering of objects

decomposition of signals into
~ independent components

classification

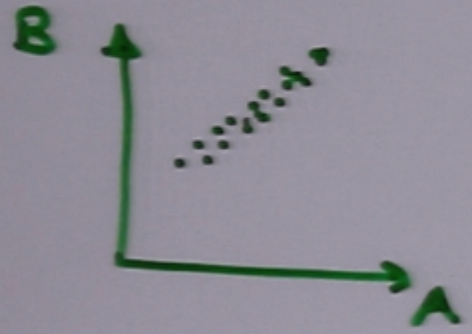
⋮



Correlation coefficient:

$$C_{AB} = \langle AB \rangle - \langle A \rangle \langle B \rangle$$

only linear relationship!




o.k.


??

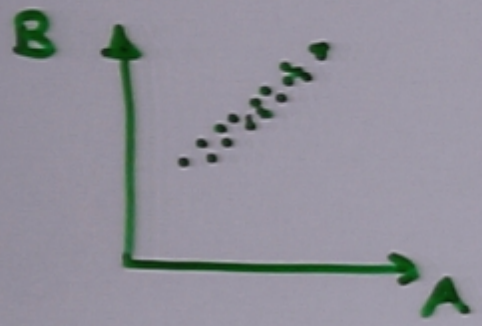
Universal,
information - theoretic
measure of dependencies:

**Mutual Information**

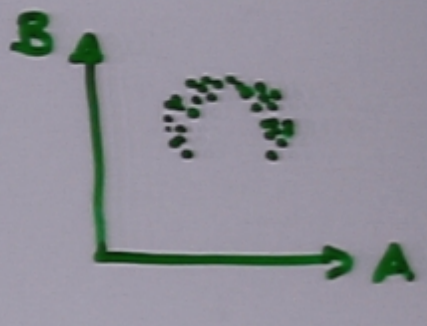
Correlation coefficient:

$$C_{AB} = \langle AB \rangle - \langle A \rangle \langle B \rangle$$

only linear relationship!




o.k.

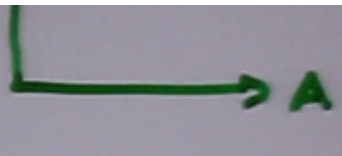



??

Universal,
information - theoretic
measure of dependencies:

**Mutual Information**


o.k.




??

Universal,
information - theoretic
measure of dependencies:

**Mutual Information**

Shannon Theory

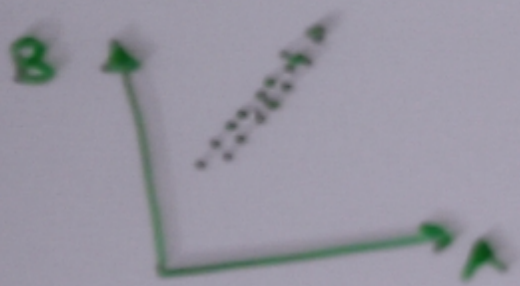
algorithmic (Kolmogorov)
inform. Theory

(1)

## Correlation coefficient:

$$C_{AB} = \langle AB \rangle - \langle A \rangle \langle B \rangle$$

only linear relationship!




o.k.


??

# Shannon information theory:

$X, Y$ = random variables

If $X, Y$ discrete:

$$H(X) = -\sum_i p(x_i) \log p(x_i)$$

Shannon entropy

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

M.I.

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

$$= \sum_{ij} p_{ij} \log \frac{p_{ij}}{p_i p_j}$$

If $X, Y$ continuous:

$X, Y$ = random variables

If $X, Y$ discrete:

$$H(X) = -\sum_i p(x_i) \log p(x_i)$$

Shannon entropy

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

M.I.

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

$$= \sum_{ij} p_{ij} \log \frac{p_{ij}}{p_i p_j}$$

If $X, Y$ continuous:

- make binning (coarse graining)

$$H(X) = -\sum_i p(x_i) \log p(x_i)$$

Shannon entropy

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

M.I.

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

$$= \sum_{ij} P_{ij} \log \frac{P_{ij}}{P_i P_j}$$

If X, Y continuous:

- make binning (coarse graining)
- take bin size $\to 0$:

$$I(X,Y) = \int dx dy \; p(x,y) \log \frac{p(x,y)}{p(x)\,p(y)}$$

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

M.I.

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

$$= \sum_{ij} P_{ij} \log \frac{P_{ij}}{P_i P_j}$$

If X, Y continuous:

- make binning (coarse graining)
- take bin size $\to 0$:

$$I(X,Y) = \int dx\,dy\; \rho(x,y) \log \frac{\rho(x,y)}{\rho(x)\,\rho(y)}$$

If X,Y discrete:

$$H(X) = -\sum_i p(x_i) \log p(x_i)$$

Shannon entropy

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

M.I.

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

$$= \sum_{ij} p_{ij} \log \frac{p_{ij}}{p_i p_j}$$

If X,Y continuous:

- make binning (coarse graining)
- take bin size $\to 0$:

Shannon entropy

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

M.I.

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

$$= \sum_{ij} p_{ij} \log \frac{p_{ij}}{p_i p_j}$$

If X, Y continuous:

- make binning (coarse graining)
- take bin size $\to 0$:

$$I(X,Y) = \int dx\,dy\, \rho(x,y) \log \frac{\rho(x,y)}{\rho(x)\rho(y)}$$

$X, Y$ = random variables

If $X, Y$ discrete:

$$H(X) = -\sum_i p(x_i) \log p(x_i)$$

Shannon entropy

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

M.I.

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

$$= \sum_{ij} p_{ij} \log \frac{p_{ij}}{p_i p_j}$$

If $X, Y$ continuous:

- make binning (coarse graining)

If X,Y discrete:

$$H(X) = -\sum_i p(x_i) \log p(x_i)$$

Shannon entropy

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

M.I.

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

$$= \sum_{ij} p_{ij} \log \frac{p_{ij}}{p_i p_j}$$

If X,Y continuous:

- make binning (coarse graining)
- take bin size → 0:

Shannon entropy

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

M.I.

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

$$= \sum_{ij} p_{ij} \log \frac{p_{ij}}{p_i p_j}$$

If X, Y continuous:

- make binning (coarse graining)
- take bin size $\to 0$:

$$I(X,Y) = \int dx\,dy\; p(x,y) \log \frac{p(x,y)}{p(x)\,p(y)}$$

N.B. : $H(X) \longrightarrow \infty$ when bin size $\to 0$ !

## Properties of MI :

1.) $I(X,Y) = I(Y,X)$ symm.

2.) $I(X,Y) \geq 0$

$= 0$ only if X,Y indep.: $p(xy) = p(x) \cdot p(y)$

3) $1 - \frac{I(X,Y)}{H(X,Y)}$ is distance ( $\Delta$- inequality)

4) $I(X,Y)$ invar under homeomorphisms

$X \to \phi(X)$ $\phi^{-1}, \psi^{-1}$ exist

$Y \to \psi(Y)$

( $H(X), H(X,Y)$ !)

$X, Y$ = random variables

If $X, Y$ discrete:

$$H(X) = -\sum_i p(x_i) \log p(x_i)$$

Shannon entropy

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

M.I.

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

$$= \sum_{ij} p_{ij} \log \frac{p_{ij}}{p_i p_j}$$

If $X, Y$ continuous:

If X,Y discrete:

$$H(X) = -\sum_i p(x_i) \log p(x_i)$$

Shannon entropy

$$I(X,Y) = H(X) + H(Y) - H(XY)$$

M.I.

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

$$= \sum_{ij} p_{ij} \log \frac{p_{ij}}{p_i p_j}$$

NB. : $H(X) \longrightarrow \infty$ when bin size $\to 0$ !

## Properties of MI :

1.) $I(X,Y) = I(Y,X)$ symm.

2.) $I(X,Y) \geq 0$

$= 0$ only if X,Y indep.

$p(xy) = p(x) \cdot p(y)$

3) $1 - \frac{I(X,Y)}{H(X,Y)}$ is distance
( $\Delta$- inequality)

4) $I(X,Y)$ invar under homeomorphisms

$X \to \phi(X)$ $\phi^{-1}, \psi^{-1}$ exist

$I(X,Y) = I(Y,X)$ symm.

2.) $I(X,Y) \geq 0$

$= 0$ only if X,Y indep.

$\rho(xy) = \rho(x) \cdot \rho(y)$

3) $1 - \frac{I(X,Y)}{H(X,Y)}$ is distance

( $\Delta$- inequality)

4) $I(X,Y)$ invar under homeomorphisms

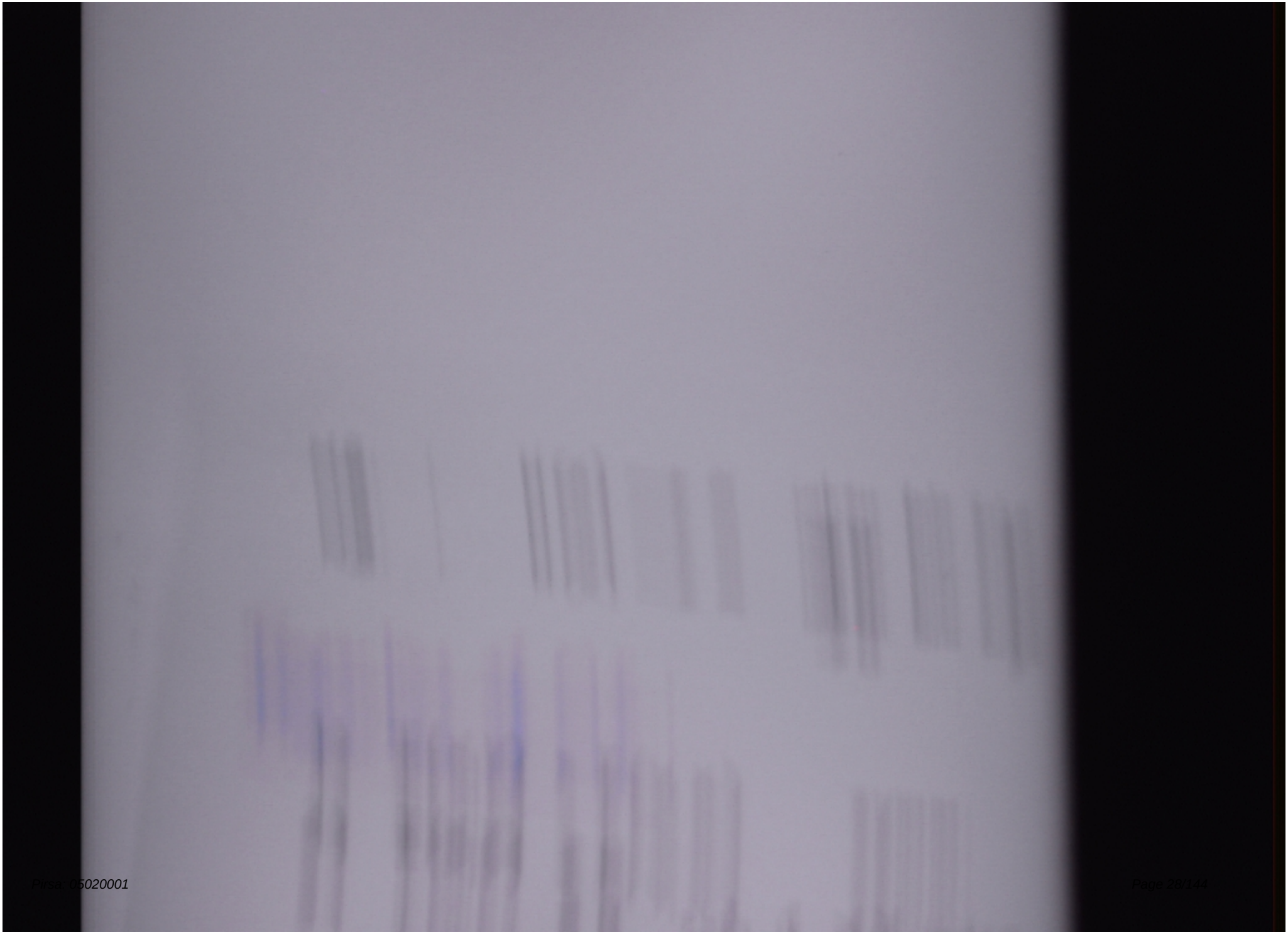
$X \to \phi(X)$

$Y \to \Psi(Y)$ $\quad \phi^{-1}, \Psi^{-1}$ exist

(not true for $H(X), H(X,Y)$ !)

5) for fixed covar matrix C :

$I(X,Y)$ is minimal for Gaussian



2.) $I(X,Y) \geq 0$

$= 0$ only if X,Y indep.:

$\rho(xy) = \rho(x) \cdot \rho(y)$

3) $1 - \frac{I(X,Y)}{H(X,Y)}$ is distance

( $\Delta$- inequality)

4) $I(X,Y)$ invar under homeomorphisms

$X \to \phi(x)$ $\quad \phi^{-1}, \psi^{-1}$ exist

$Y \to \psi(Y)$

(not true for $H(X), H(X,Y)$ !)

5) for fixed covar matrix C :

$I(X,Y)$ is minimal for Gaussian

2.) $I(X,Y) \geq 0$

$= 0$ only if X,Y indep.:

$\rho(xy) = \rho(x) \cdot \rho(y)$

3) $1 - \frac{I(X,Y)}{H(X,Y)}$ is distance

( $\Delta$- inequality)

4) $I(X,Y)$ invar under homeomorphisms

$X \to \phi(X)$ $\quad \phi^{-1}, \psi^{-1}$ exist

$Y \to \psi(Y)$

(not true for $H(X), H(X,Y)$ !)

5) for fixed covar matrix $C$ :

$I(X,Y)$ is minimal for Gaussian

$$\Rightarrow \quad I(X_1 X_2 \cdots X_m) \geq \frac{1}{2} \log \frac{\det C}{C_{11} C_{22} \cdots C_{mm}}$$

# MI for $\geq 3$ Variables:

- there are various definitions in the literature !

ICA literature:

$$I(X_1 \cdots X_n) = H(X_1) + \cdots + H(X_n) - H(X_1 \cdots X_n)$$

Grouping property:

def.: $W = (X, Y)$ : joint random variable

$$I(X, Y, Z, \ldots) = I(W, Z, \ldots) + I(X, Y)$$

# MI for $\geq 3$ Variables:

- there are various definitions in the literature !

ICA literature :

$$I(X_1 \cdots X_n) = H(X_1) + \cdots + H(X_n) - H(X_1 \cdots X_n)$$

Grouping property :

def.: $W = (X, Y)$ : joint random variable

$$I(X, Y, Z, \ldots) = I(W, Z, \ldots) + I(X, Y)$$

$$I(X_1 \cdots X_n) = H(X_1) + \cdots + H(X_n) - H(X_1 \cdots X_n)$$

**Grouping property** :

def.: $W = (X, Y)$ : joint random variable

$$I(X, Y, Z, \ldots) = I(W, Z, \ldots) + I(X, Y)$$

$\Rightarrow$ hierarchical cluster decomposition of MI:

MI = MI within clusters + MI between clusters

$$I(X_1 \cdots X_n) = H(X_1) + \cdots + H(X_n) - H(X_1 \cdots X_n)$$

**Grouping property** :

def.: $W = (X, Y)$ : joint random variable

$$I(X, Y, Z, \ldots) = I(W, Z, \ldots) + I(X, Y)$$

⇒ hierarchical cluster decomposition of MI:

MI = MI within clusters + MI between clusters

$X = x_1 x_2 x_3 \cdots x_N$ : symbol string
$x_i \in$ "alphabet" $\mathcal{A}$

$U$ = universal (Turing-) computer,
i.e. PC

$C_U(X)$ = length (in bits) of shortest program which prints $X$ and then halts $U$

"Kolmogorov complexity of $X$"

Thms:

- $\lim_{N\to\infty} \frac{1}{N} C_U(X)$ is indep. of $U$ for nearly all $X$

- if $X$ is randomly drawn from stochastic process with entropy $H$

# Algorithmic Inform. Theory:

$X = x_1 x_2 x_3 \cdots x_N$ : symbol string
$x_i \in$ "alphabet" $\mathcal{A}$

$U$ = universal (Turing-) computer,
i.e. PC

$C_U(X)$ = length (in bits) of shortest program which prints $X$ and then halts $U$

"Kolmogorov complexity of $X$"

Thms:

- $\lim_{N\to\infty} \frac{1}{N} C_U(X)$ is indep. of $U$ for nearly all $X$

- if $X$ is randomly drawn from

$U$ = universal (Turing-) computer,
i.e. PC

$C_U(X)$ = length (in bits) of shortest program which prints $X$ and then halts $U$

"Kolmogorov complexity of $X$"

Thms:

- $\lim_{N\to\infty} \frac{1}{N} C_U(X)$ is indep. of $U$ for nearly all $X$

- if $X$ is randomly drawn from stochastic process with entropy $H$ per letter:

$$\Rightarrow \lim_{N\to\infty} \frac{1}{N} C_U(X) = H$$

$C_u(X)$ = length (in bits) of shortest program which prints $X$ and then halts $U$

"Kolmogorov complexity of $X$"

Thus:

- $\lim_{N\to\infty} \frac{1}{N} C_u(X)$ is indep. of $U$ for nearly all $X$

- if $X$ is randomly drawn from stochastic process with entropy $H$ per letter:

$$\Rightarrow \lim_{N\to\infty} \frac{1}{N} C_u(X) = H$$

## Algorithmic Inform. Theory:

$X = x_1 x_2 x_3 \cdots x_N$ : symbol string
$x_i \in$ "alphabet" $A$

$U$ = universal (Turing-) computer,
i.e. PC

$C_U(X)$ = length (in bits) of shortest program which prints $X$ and then halts $U$

"Kolmogorov complexity of $X$"

Thms:

- $\lim_{N\to\infty} \frac{1}{N} C_U(X)$ is indep. of $U$ for nearly all $X$

- if $X$ is randomly drawn from stochastic process with entropy $H$

$U$ = universal (Turing-) computer,
i.e. PC

$C_U(X)$ = length (in bits) of shortest program which prints $X$ and then halts $U$

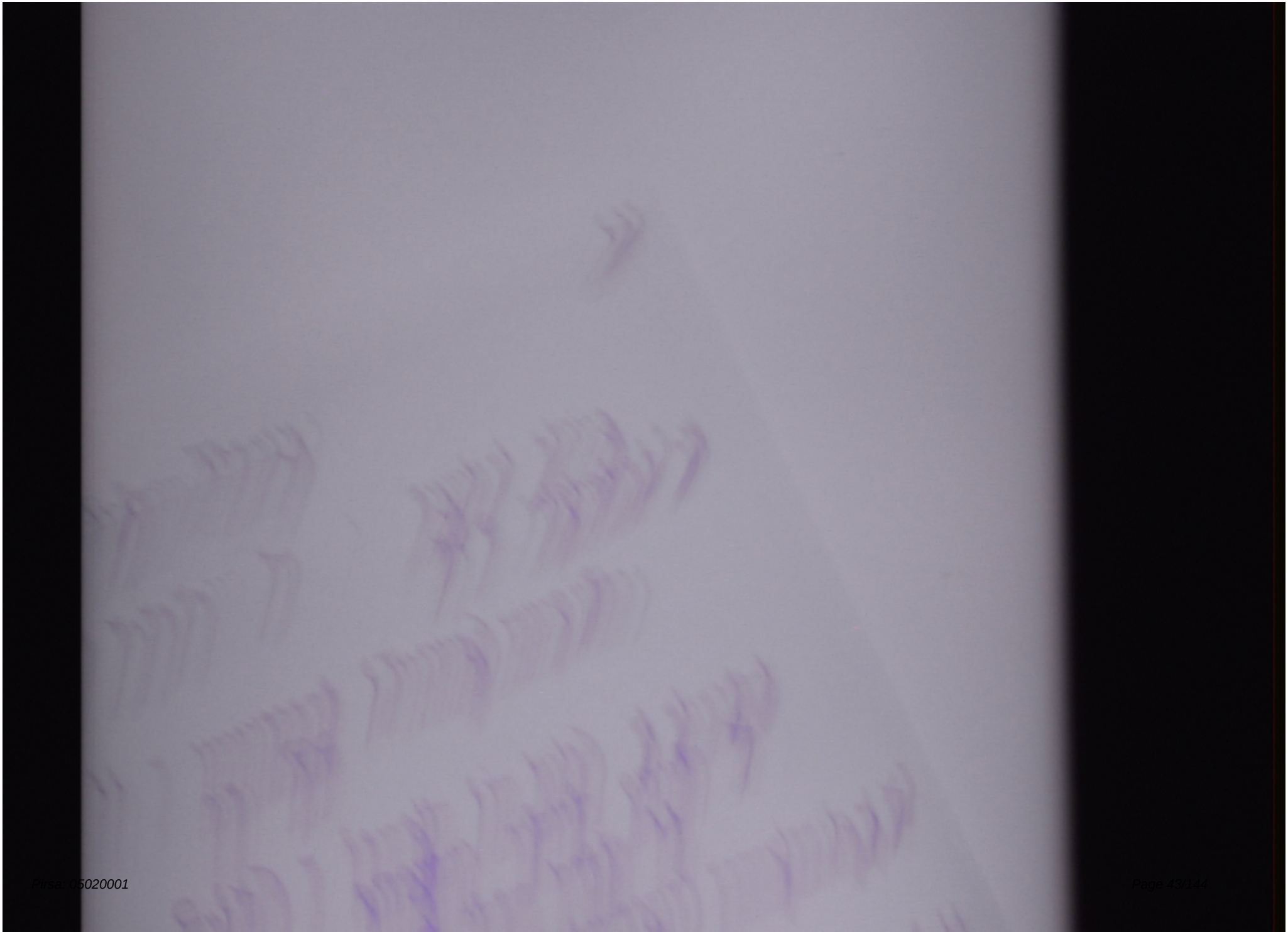
"Kolmogorov complexity of $X$"

Thms:

- $\lim_{N\to\infty} \frac{1}{N} C_U(X)$ is indep. of $U$ for nearly all $X$

- if $X$ is randomly drawn from stochastic process with entropy $H$ per letter:

$$\Rightarrow \lim_{N\to\infty} \frac{1}{N} C_U(X) = H$$



- $C_u(x)$ is not computable, i.e.

  $\nexists$ <u>algorithm</u> for calculating it

Lower estimates on $C(x)$:

given by any compression algor.,

e.g. zip, gzip, compress, bzip2,

...

Mutual information between

$X = x_1 x_2 \cdots x_N$

$Y = y_1 \cdots y_M$ :

$$I_c(X,Y) = C(X) + C(Y) - C(\overrightarrow{XY})$$

concatenation $x_1 \cdots x_N y_1 \cdots y_N$

notice : $XY \neq YX$,

$C(XY) = C(YX) + \mathcal{O}(\ln N)$

- $C_u(x)$ is not computable, i.e.
  $\nexists$ algorithm for calculating it

Lower estimates on $C(x)$:
given by any compression algor.,
e.g. zip, gzip, compress, bzip2,
...

Mutual information between
$X = x_1 x_2 \cdots x_N$
$Y = y_1 \cdots y_M$ :

$$I_c(X,Y) = C(X) + C(Y) - C(XY)$$

$XY$: concatenation $x_1 \cdots x_N y_1 \cdots y_N$

notice : $XY \neq YX$,
$C(XY) = C(YX) + O(\ln N)$

∄ algorithm for [illegible]

Lower estimates on $C(x)$:

given by any compression algor.,

e.g. zip, gzip, compress, bzip2,

...

Mutual information between

$X = x_1 x_2 \cdots x_N$

$Y = y_1 \cdots y_M$ :

$$I_c(X,Y) = C(X) + C(Y) - C(\underset{\nearrow}{XY})$$

concatenation $x_1 \cdots x_N y_1 \cdots y_N$

notice : $XY \neq YX$,

$C(XY) = C(YX) + O(\ln N)$

⇒ all properties of Shannon-MI for

~~∃~~ algorithm for calculating it

Lower estimates on $C(x)$:

given by any compression algor.,

e.g. zip, gzip, compress, bzip2,

...

Mutual information between

$X = x_1 x_2 \cdots x_N$

$Y = y_1 \cdots y_M$ :

$$I_c(X,Y) = C(X) + C(Y) - C(XY)$$

$XY$: concatenation $x_1 \cdots x_N y_1 \cdots y_N$

notice : $XY \neq YX$,

$C(XY) = C(YX) + O(\ln N)$

Lower estimates on $C(x)$:

given by any compression algor.,

e.g. zip, gzip, compress, bzip2,

...

Mutual information between

$X = x_1 x_2 \cdots x_N$

$Y = y_1 \cdots y_M$ :

$$I_c(X,Y) = C(X) + C(Y) - C(\overrightarrow{XY})$$

concatenation $x_1 \cdots x_N y_1 \cdots y_N$

notice : $XY \neq YX$,

$C(XY) = C(YX) + O(\ln N)$

⇒ all properties of Shannon-MI for discrete variables hold up to terms $\sim \ln N$

Main difference between algorithmic & Shannon versions:

$H(X)$ = property of <u>random variable</u>

$C(X)$ = property of <u>single realization</u>

Let $x$ be produced by stochastic process $\mathbf{X}$:

estimation of $C(x)$ via compression alger.

- 4 - $H(\mathbf{X})$: via estimate of <u>statistics</u>, e.g. block frequencies, from $x$

$\Rightarrow$ basically same estimates

$H(X)$ = property of random variable

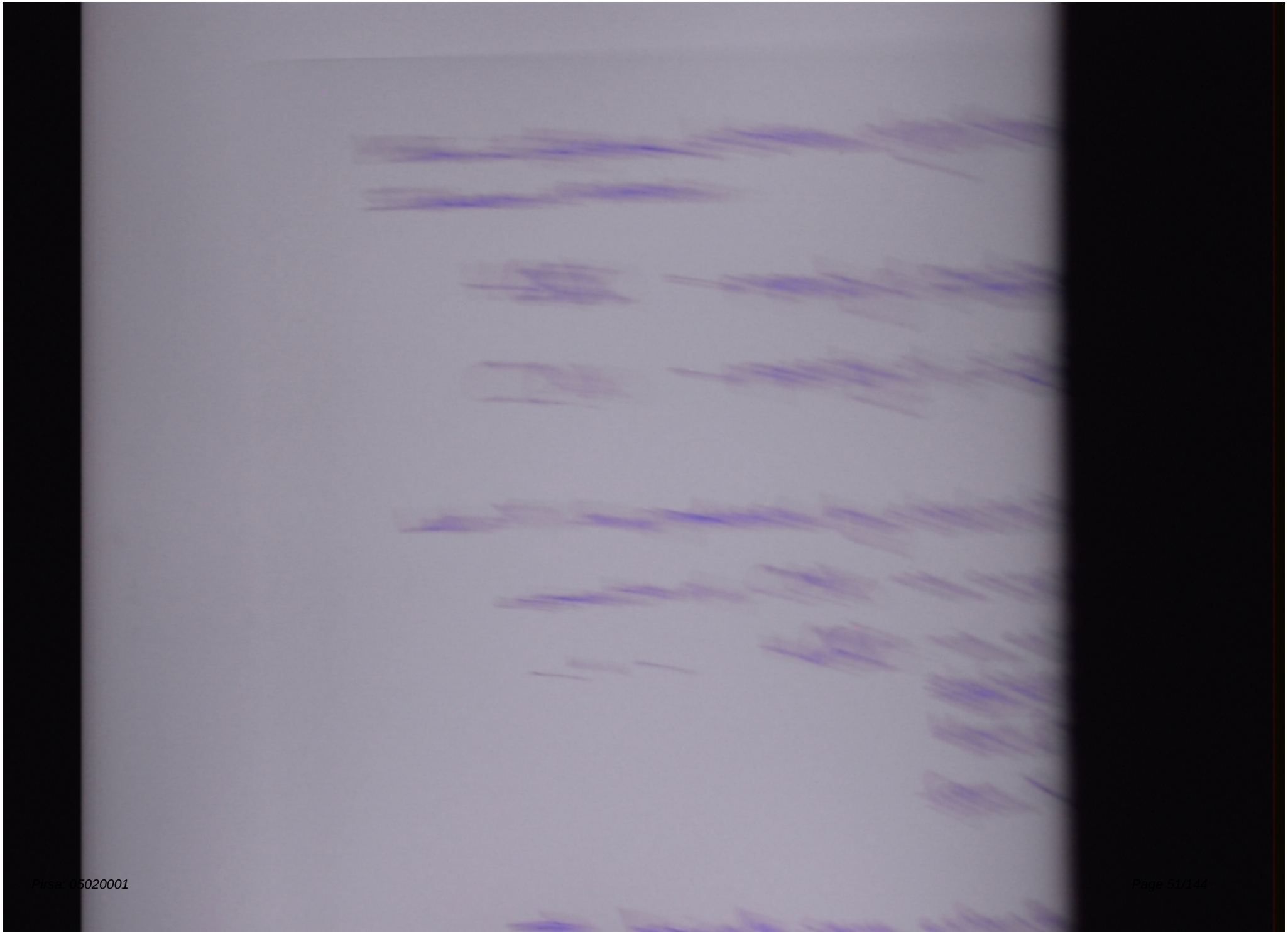
$C(X)$ = property of single realization

Let $X$ be produced by stochastic process $\mathbf{X}$:

estimation of $C(X)$ via compression algor.

- 4 - $H(X)$: via estimate of statistics, e.g. block frequencies, from $X$

$\Rightarrow$ basically same estimate



## Phylogenetic trees

X
Y
Z
⋮

} mitochondrial genomes of 34 mammals (~ 16,000 bases each)

$I_c(X,Y)$ = algorithmic Mi between X,Y obtained with standard compr (e.g. bzip2)

$D(X,Y) = 1 - \frac{I_c(X,Y)}{C(X,Y)}$ = normalized distance

# Phylogenetic trees

X
Y
Z
⋮

} mitochondrial genomes of 34 mammals (~ 16,000 bases each)

$I_c(X,Y)$ = algorithmic MI between X, Y obtained with standard compressor (e.g. bzip2)

$$D(X,Y) = 1 - \frac{I_c(X,Y)}{C(X,Y)}$$ = normalized distance

X
Y
Z
⋮

} mitochondrial genomes of 34 mammals

(~ 16,000 bases each)

$I_c(X,Y)$ = algorithmic MI between X, Y obtained with standard compressor (e.g. bzip2)

$D(X,Y) = 1 - \frac{I_c(X,Y)}{C(X,Y)}$ = normalized distance

X
Y
Z
⋮

} mitochondrial genomes of 34 mammals (~ 16,000 bases each)

$I_c(X,Y)$ = algorithmic MI between X, Y obtained with standard compressor (e.g. bzip2)

$D(X,Y) = 1 - \frac{I_c(X,Y)}{C(X,Y)}$ = normalized distance

X
Y
Z
⋮

} mitochondrial genomes of 34 mammals

(~ 16,000 bases each)

$I_c(X,Y)$ = algorithmic MI between X, Y obtained with standard compressor (e.g. bzip2)

$$\mathcal{D}(X,Y) = 1 - \frac{I_c(X,Y)}{C(X,Y)}$$ = normalized distance

X
Y
Z
⋮

} mitochondrial genomes of 34 mammals

(~ 16,000 bases each)

$I_c(X,Y)$ = algorithmic MI between X, Y obtained with standard compressor (e.g. bzip2)

$$\mathcal{D}(X,Y) = 1 - \frac{I_c(X,Y)}{C(X,Y)}$$ = normalized distance

X
Y
Z
⋮

} mitochondrial genomes of 34 mammals

(~ 16,000 bases each)

$I_c(X,Y)$ = algorithmic MI between X, Y obtained with standard compressor (e.g. bzip2)

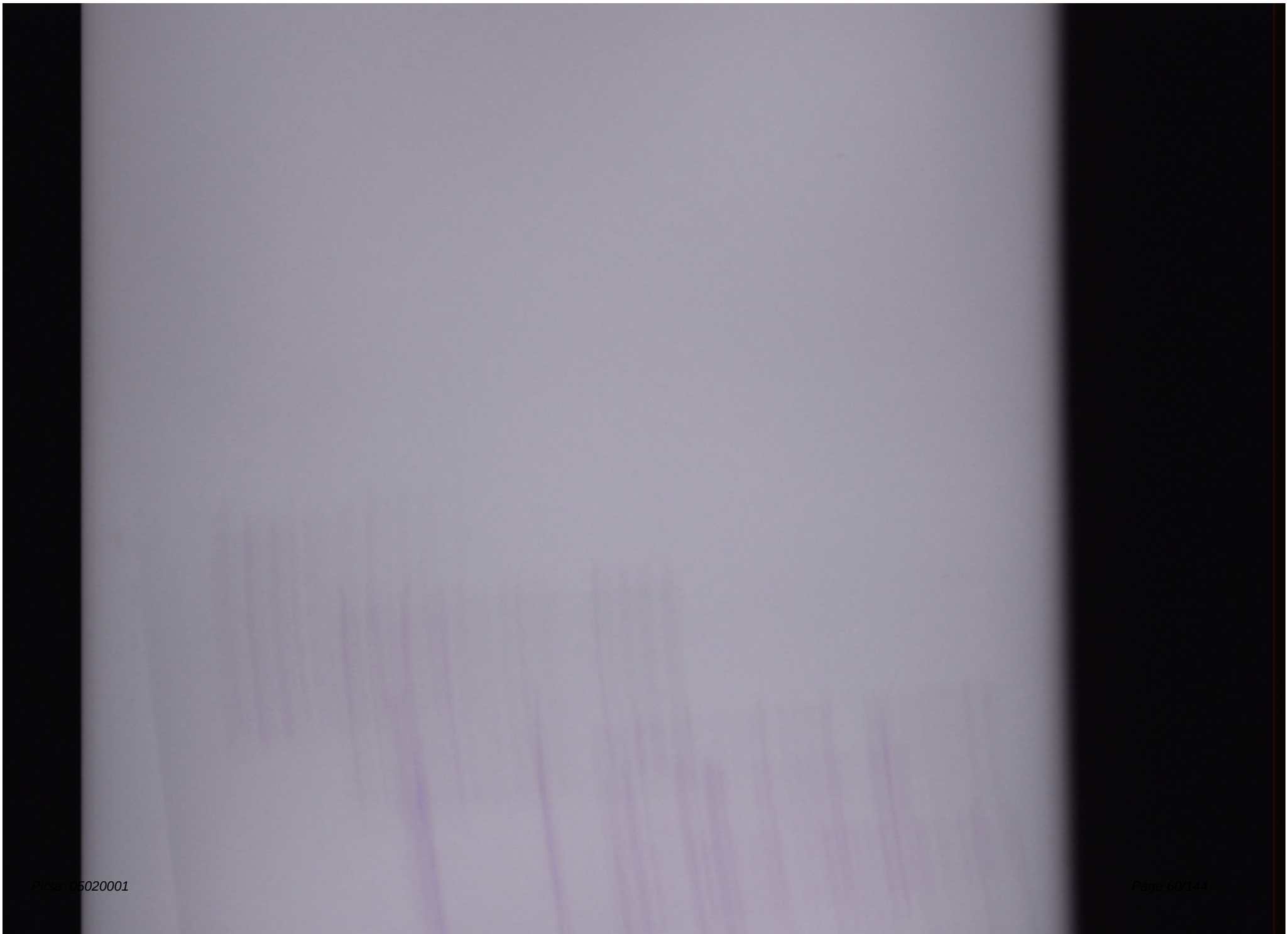
$D = 1 - \frac{I_c(X,Y)}{C(X,Y)}$ = normalized distance

X
Y
Z
⋮

} mitochondrial genomes of 34 mammals (~ 16,000 bases each)

$I_c(X,Y)$ = algorithmic MI between X, Y obtained with standard compressor (e.g. bzip2)

$D(X,Y) = 1 - \frac{I_c(X,Y)}{C(X,Y)}$ = normalized distance



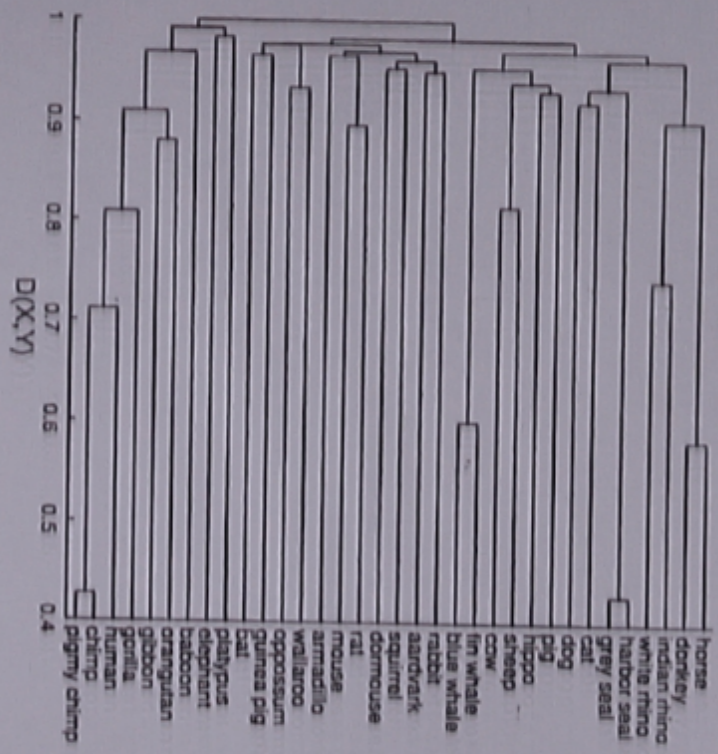



Fig. 1 – Phylogenetic tree for 34 mammals. The heights of nodes are the distances between the joi
daughter clusters.

Algorithm:

① start with $K = 34$ genomes

② form $K \times K$ matrix of distances

③ search minimal distance, say between species $X_j$, $X_\ell$

④ concatenate $X' = X_j X_\ell$
eliminate species $X_j$, $X_\ell$
&
replace by $X'$
(thereby $K \to K-1$)

⑤ if $K > 1$, goto ②

= typical hierarchical clustering alg.

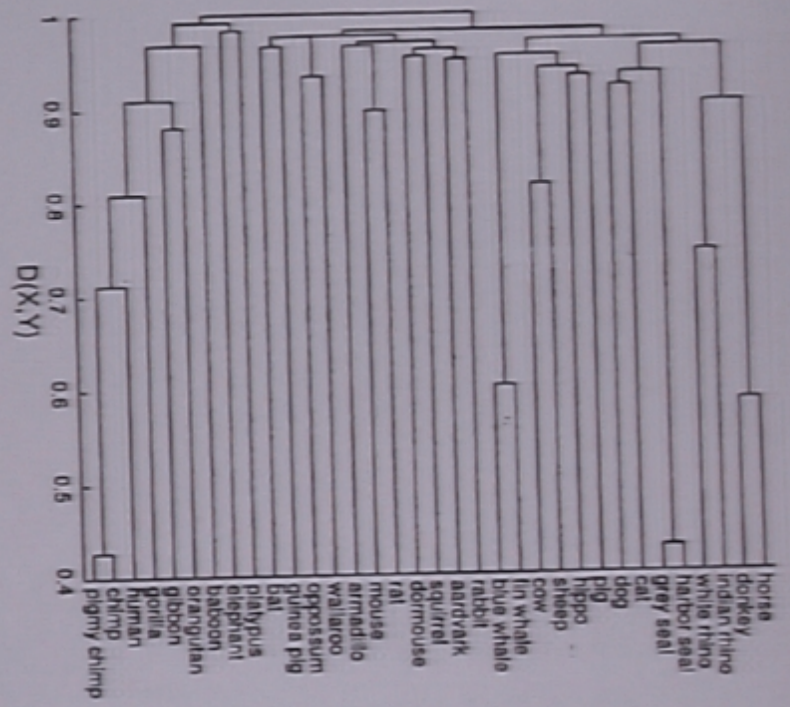



Fig. 1 – Phylogenetic tree for 34 mammals. The heights of nodes are the distances between t
daughter clusters.

Algorithm:

① start with $K = 34$ genomes

② form $K \times K$ matrix of distances ←

③ search minimal distance, say between species $X_j$, $X_\ell$

④ concatenate $X' = X_j X_\ell$

eliminate species $X_j$, $X_\ell$

&

replace by $X'$

(thereby $K \to K-1$)

⑤ if $K > 1$, goto ② ——

= typical hierarchical clustering alg.,
except that replacement

$X_j, X_\ell \to X'$

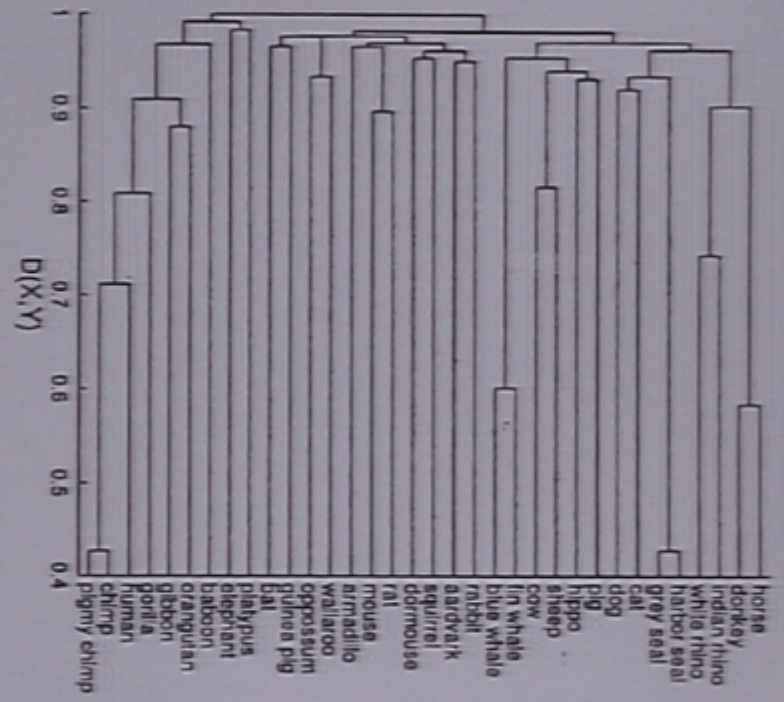
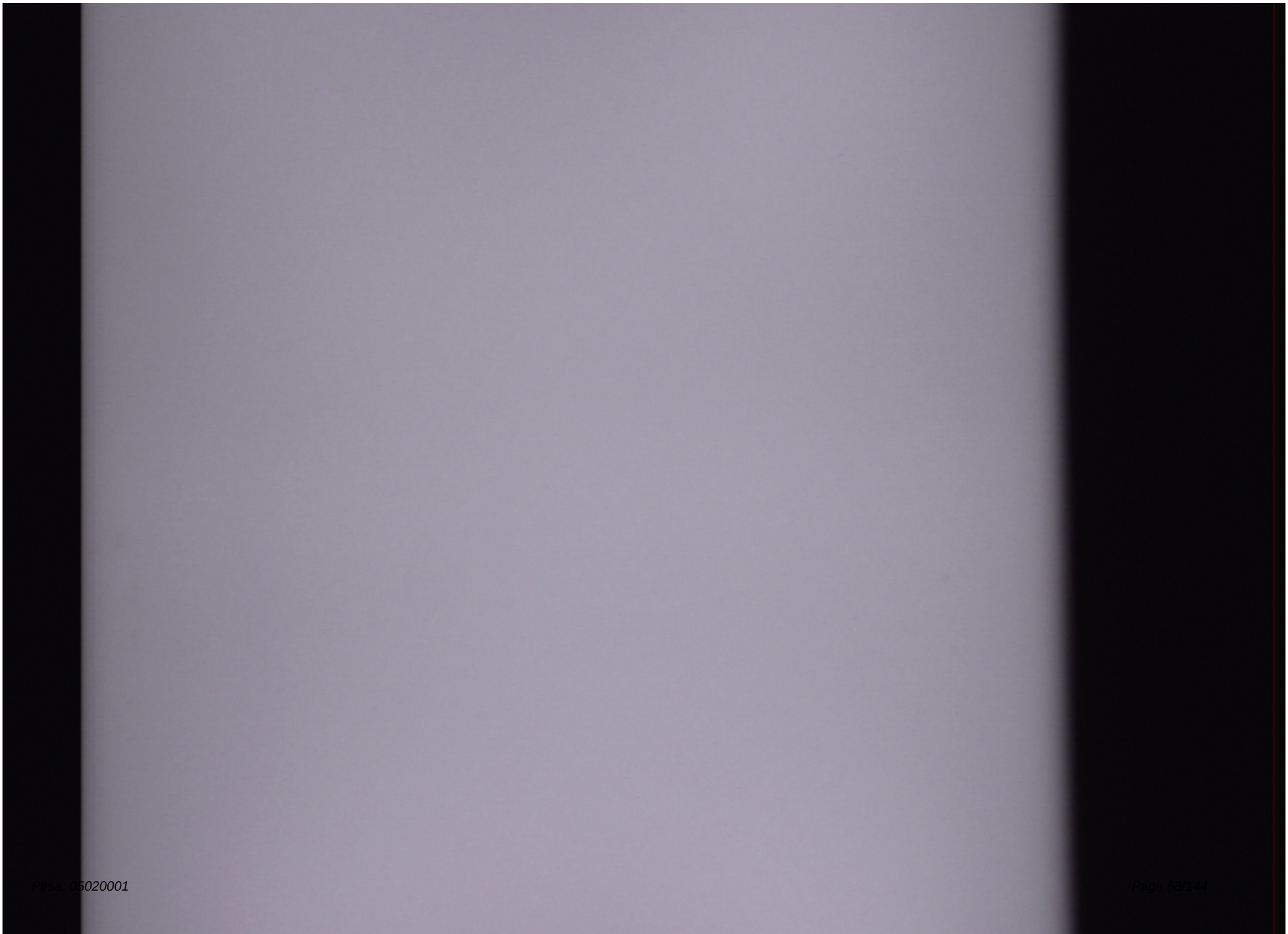



Fig. 1 – Phylogenetic tree for 34 mammals. The heights of nodes are the distances between the jo
daughter clusters.



# Shannon MI of

# Continuous random variables

# Estimating MI :

finite sample $(x,y)_1 \cdots (x,y)_N$ iid

$\Rightarrow \hat{I}(X,Y)$ ?

- binning
  - fixed grid : very bad !
  - adaptive grid

    Fraser & Swinney
    Darbellay & Vajda

    still large systematic error
    large statistical ---
    difficult in high dim's
    ... but fast !

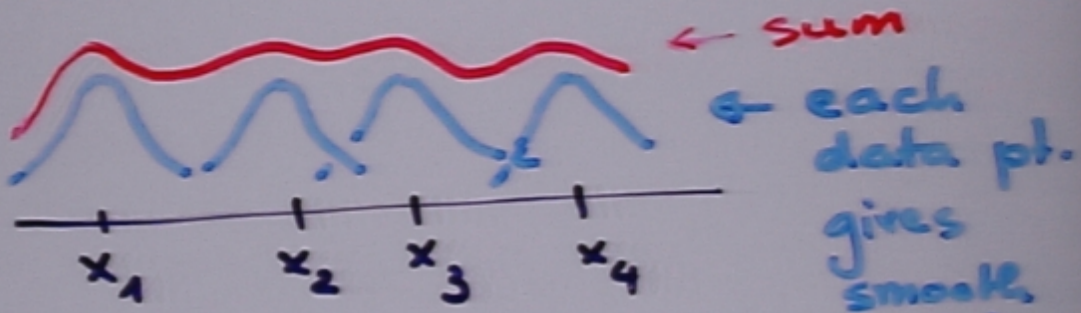
- Kernel methods

- adaptive grid

  Frazer & Swinney
  Darbellay & Vajda

  still large systematic error
  large statistical --"--
  difficult in high dim's
  ... but fast !

- Kernel methods

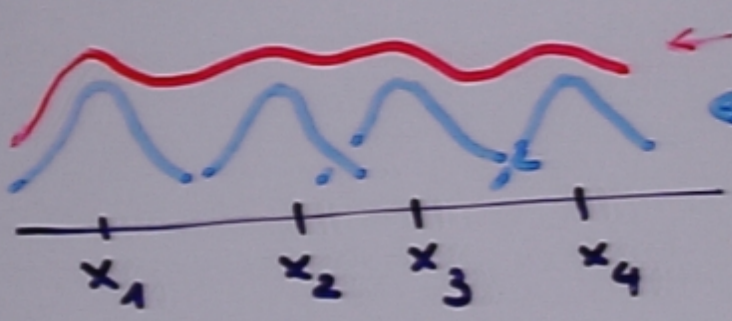


- adaptive grid

  Frazer & Swinney
  Darbellay & Vajda

  still large systematic error
  large statistical ---
  difficult in high dim's
  ... but fast !

- Kernel methods



very large systematic errors,
unless width of kernels
chosen optimally :

too wide → poor resolution
too narrow → entropies
underestimated

optimal choice ??

- estimators based on
nearest neighbour statistics
!!

too wide → poor resolution

too narrow → entropies underestimated

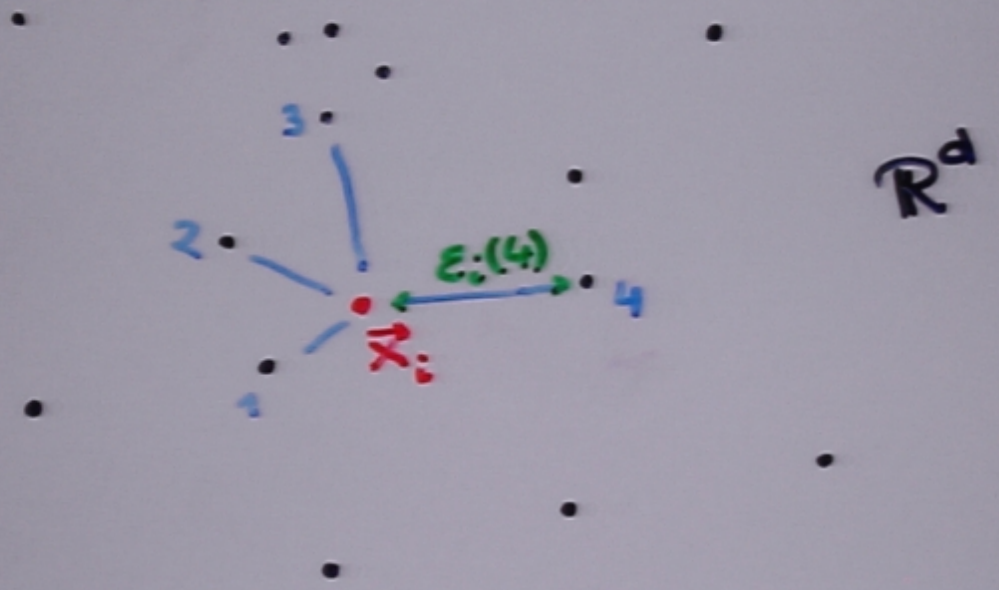
optimal choice ??

- estimators based on nearest neighbour statistics

!!

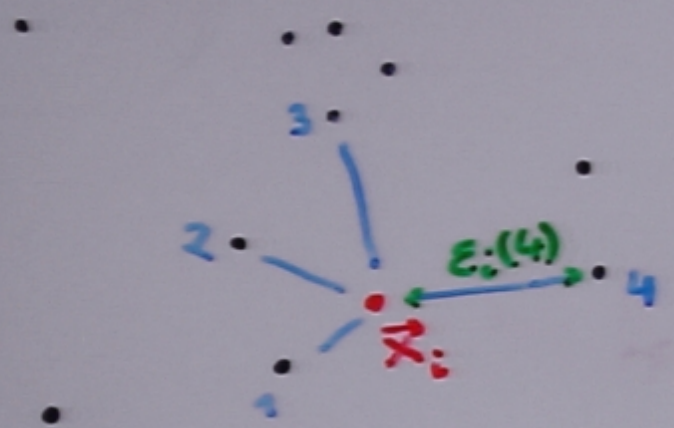
entropy:

$$H(X) = -\int dx\, \rho(x) \log \rho(x)$$




Kozachenko-Leonenko:

$$\hat{H}_k(X) = -\psi(k) + \psi(N) + \log c_d + \frac{d}{N} \sum^{N} \log \varepsilon_i(k)$$

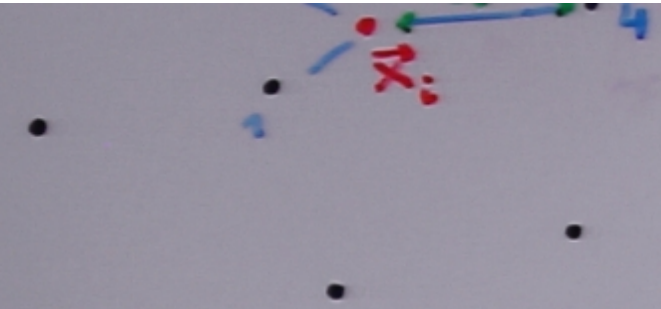



$\mathbb{R}^d$

Kozachenko - Leonenko :

$$\hat{H}_k(X) = -\psi(k) + \psi(N) + \log c_d + \frac{d}{N} \sum_{i=1}^{N} \log \varepsilon_i(k)$$

$$\psi(x) = \frac{d}{dx} \ln \Gamma(x) \quad \text{(digamma-fct.)}$$



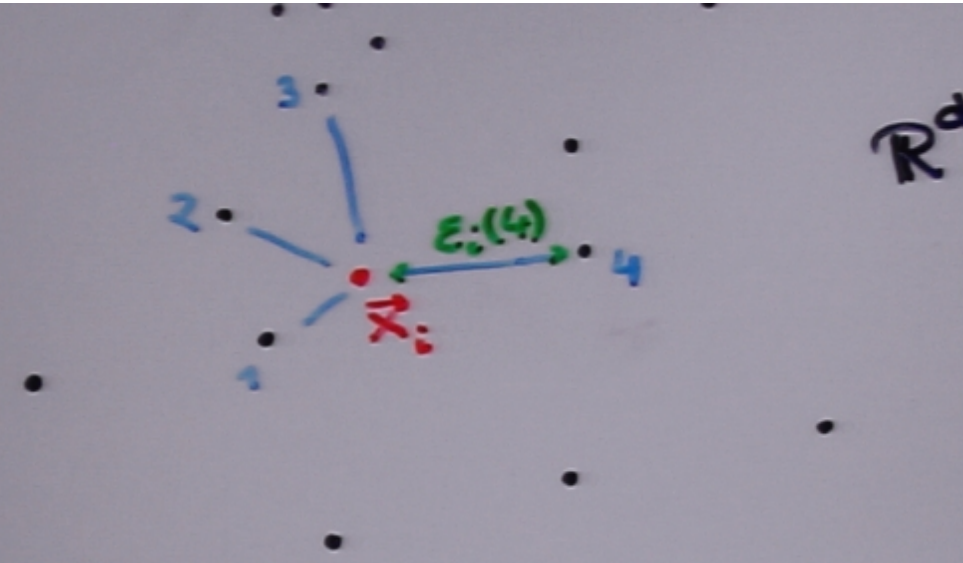

Kozachenko-Leonenko:

$$\hat{H}_k(X) = -\psi(k) + \psi(N) + \log c_d + \frac{d}{N}\sum_{i=1}^{N} \log \varepsilon_i(k)$$

$\psi(x) = \frac{d}{dx} \ln \Gamma(x)$ (digamma-fct.)

$c_d$ = volume of d-dim unit ball

$\varepsilon_i(k)$ = distance to k-te neighbor of $x_i$



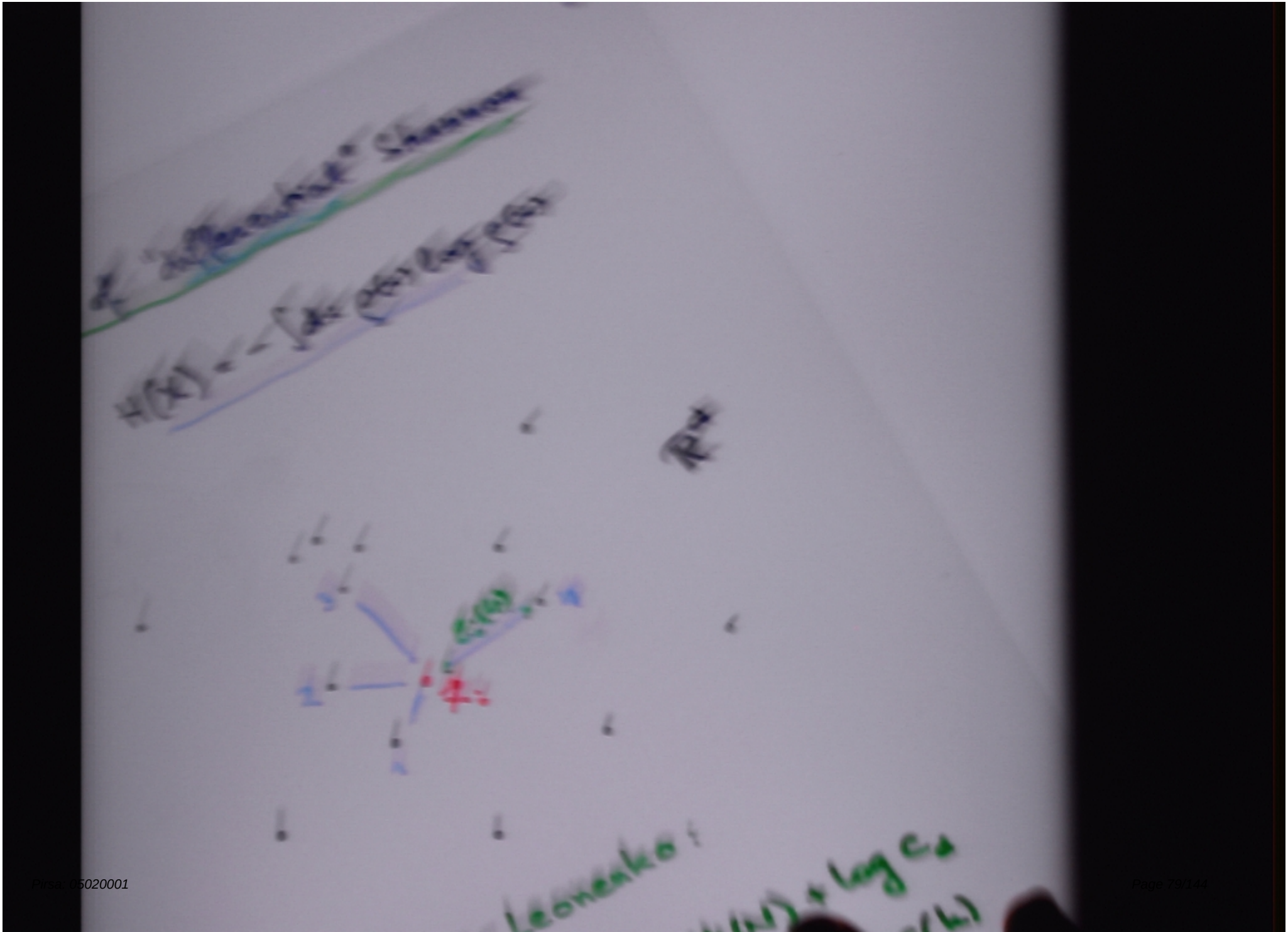

Kozachenko-Leonenko:

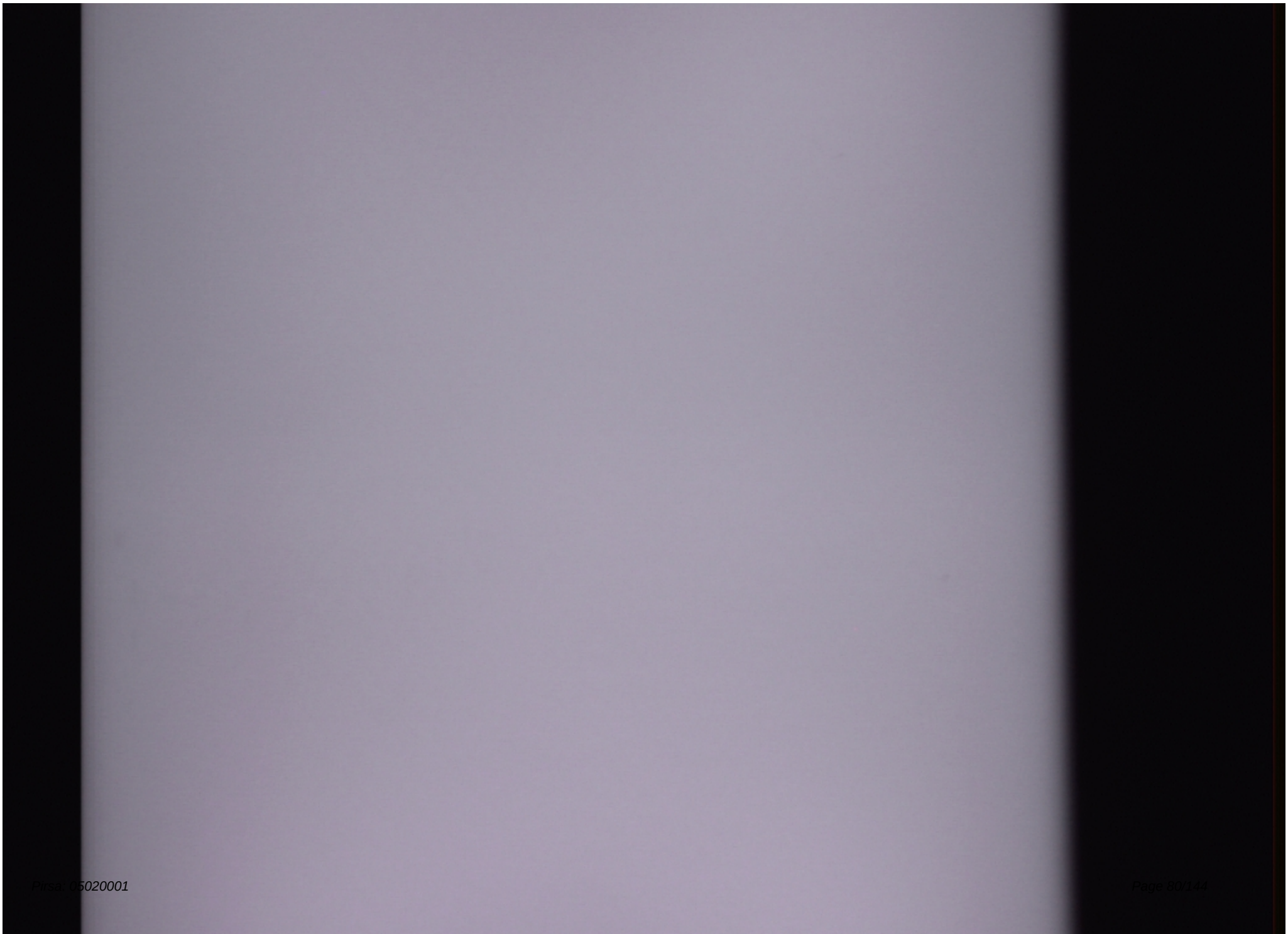
$$\hat{H}_k(x) = -\psi(k) + \psi(N) + \log c_d + \frac{d}{N}\sum_{i=1}^{N} \log \varepsilon_i(k)$$

$\psi(x) = \frac{d}{dx} \ln \Gamma(x)$ (digamma-fct.)

$c_d$ = volume of d-dim unit ball

$\varepsilon_i(k)$ = distance to k-te neighbor of $x_i$





## Estimation of $I(X,Y)$:

- $\hat{I}_k(X,Y) = \hat{H}_k(X) + \hat{H}_k(Y) - \hat{H}_k(X,Y)$

  $\hat{H}_k(X)$: bad !! $\hat{H}_k(X,Y)$

  $\varepsilon(k)$ small — $\varepsilon(k)$ large

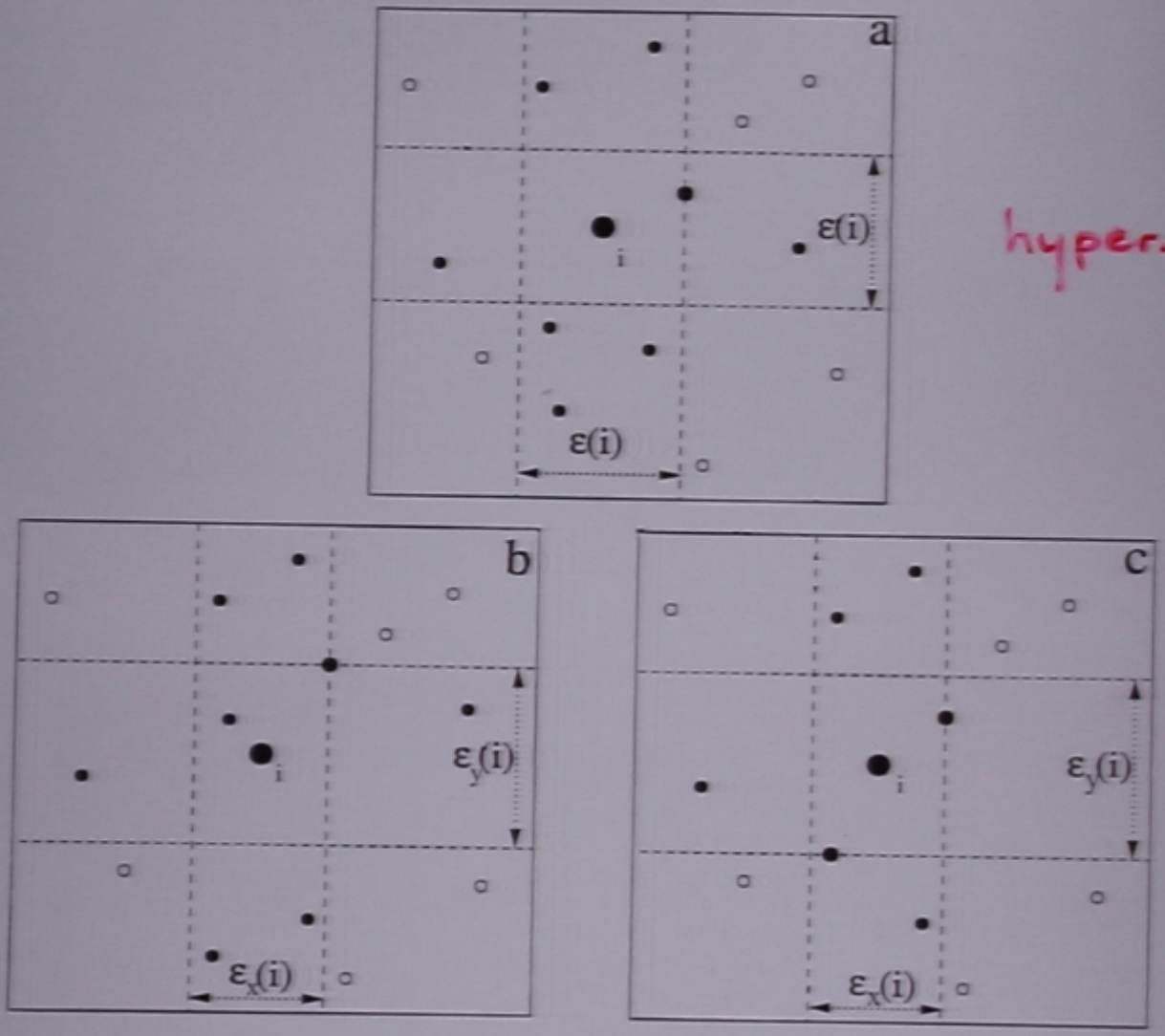
  no cancelling of errors

- better: use the same $\varepsilon_i(k)$ for marginal & joint spaces !

  $k$ : # neighbours in joint space

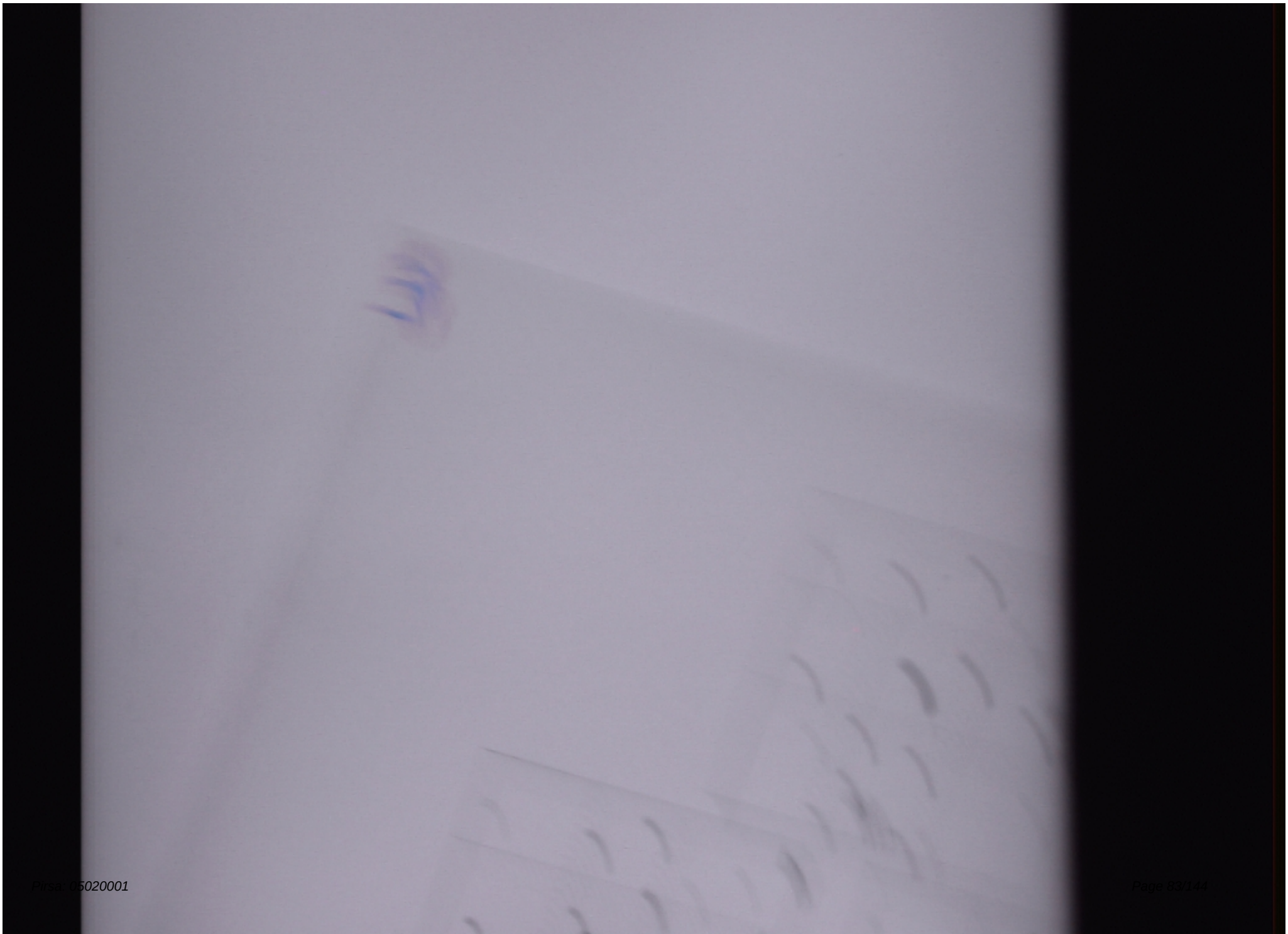
  $n_x(i)$: — " — in X space

  $n_y$ — " — Y




hyper-cub

rec



better: use the <u>same</u> $\varepsilon_i(k)$ for marginal & joint spaces !

$k$ : # neighbours in joint space

$n_x(i)$ : " " in X space

$n_y$ " " Y

$$\Rightarrow \hat{I}(X,Y) = \psi(k) - \frac{1}{k} + \\ - \langle \psi(n_x) + \psi(n_y) \rangle \\ + \psi(N)$$

no cancelling of errors

- better: use the same $\varepsilon_i(k)$ for marginal & joint spaces !

$k$ : # neighbours in joint space

$n_x(i)$ : " " in X space

$n_y$ " " Y

$$\Rightarrow \hat{I}(X,Y) = \psi(k) - \frac{1}{k} + \\ - \langle \psi(n_x) + \psi(n_y) \rangle \\ + \psi(N)$$

$$H(X) = -\sum_{x} p(x) \log_2 p(x)$$




Kosambi-Teorema:

$$\hat{H}(x) = -\Psi(r) + \Psi(n) + \log c^q + \frac{n}{q} \sum_{i=1}^{n} \log \varepsilon(r_i)$$

no cancelling of errors

- better: use the <u>same</u> $\varepsilon_i(k)$ for marginal & joint spaces !

$k$ : # neighbours in joint space

$n_x(i)$ : " " in X space

$n_y$ : " " Y

$$\Rightarrow \hat{I}(X,Y) = \psi(k) - \frac{1}{k} + \\ - \langle \psi(n_x) + \psi(n_y) \rangle \\ + \psi(N)$$

no cancelling of errors

- better: use the <u>same</u> $\varepsilon_i(k)$ for marginal & joint spaces !

$k$ : # neighbours in joint space

$n_x(i)$ : " " in X space

$n_y$ " " Y

$$\Rightarrow \hat{I}(X,Y) = \psi(k) - \frac{1}{k} + \\ - \langle \psi(n_x) + \psi(n_y) \rangle \\ + \psi(N)$$

one uses (hyper-) cubes
(= balls in max norm)

other uses (hyper-) rectangles

other version slightly different

Both versions:

- small statistical errors
- fast, if efficient data structures used
- robust against outliers
- if degeneracies due to discretization
  ⇒ randomize by adding small noise

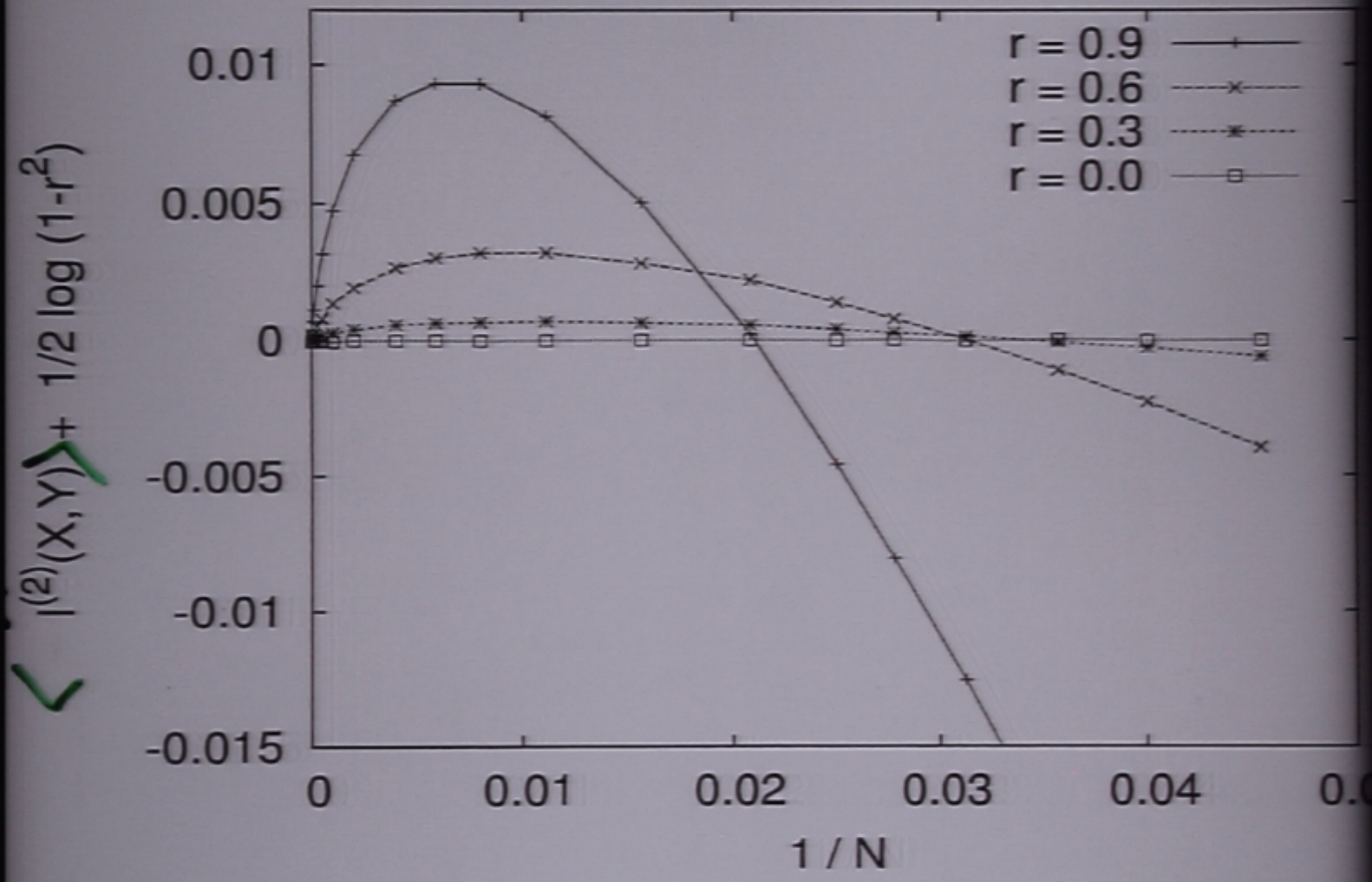
other version slightly different

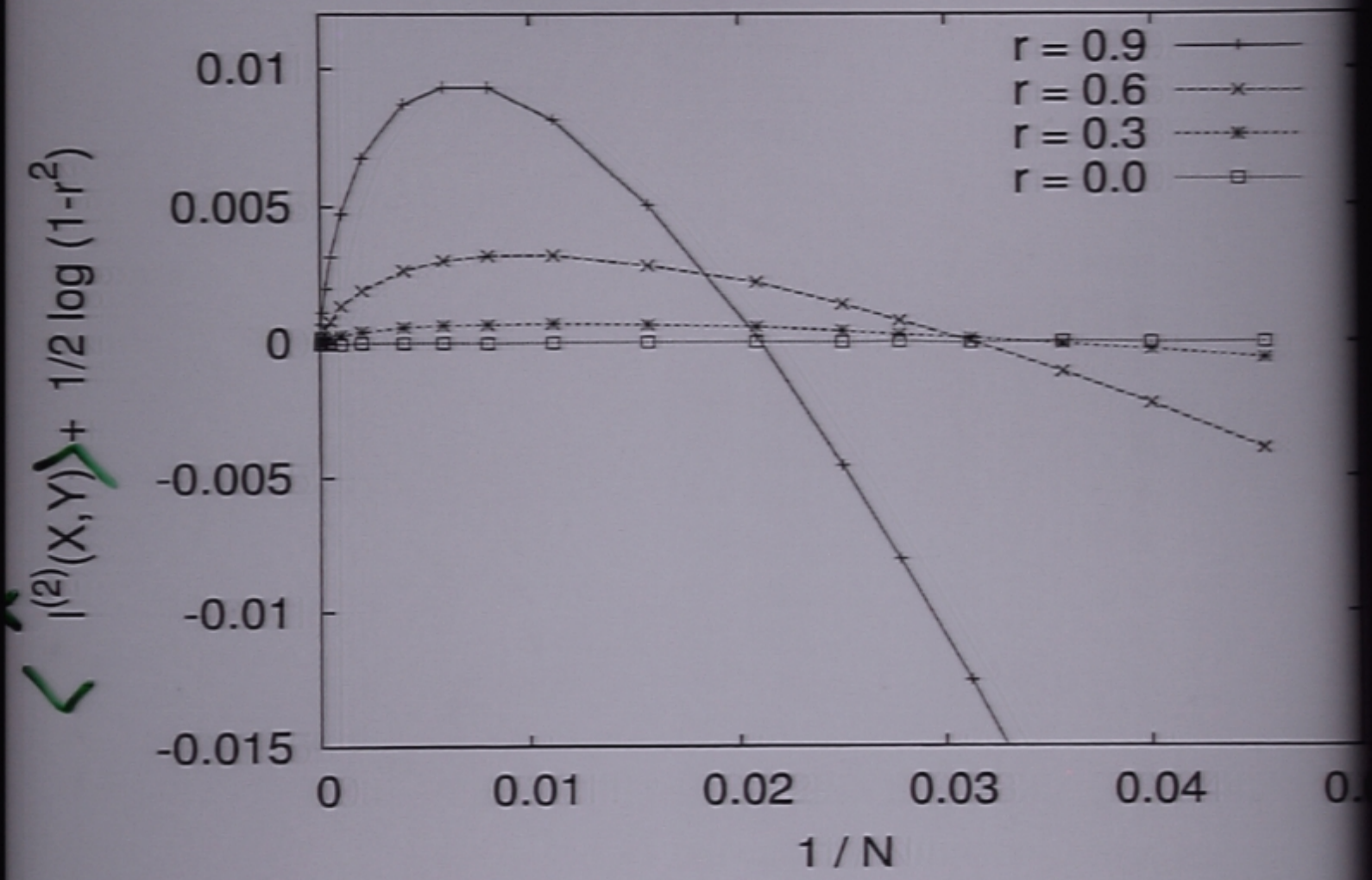
Both versions:

- small statistical errors
- fast, if efficient data structures used
- robust against outliers
- if degeneracies due to discretization
  ⇒ randomize by adding small noise
- small systematic errors
- no systematic errors, if X,Y

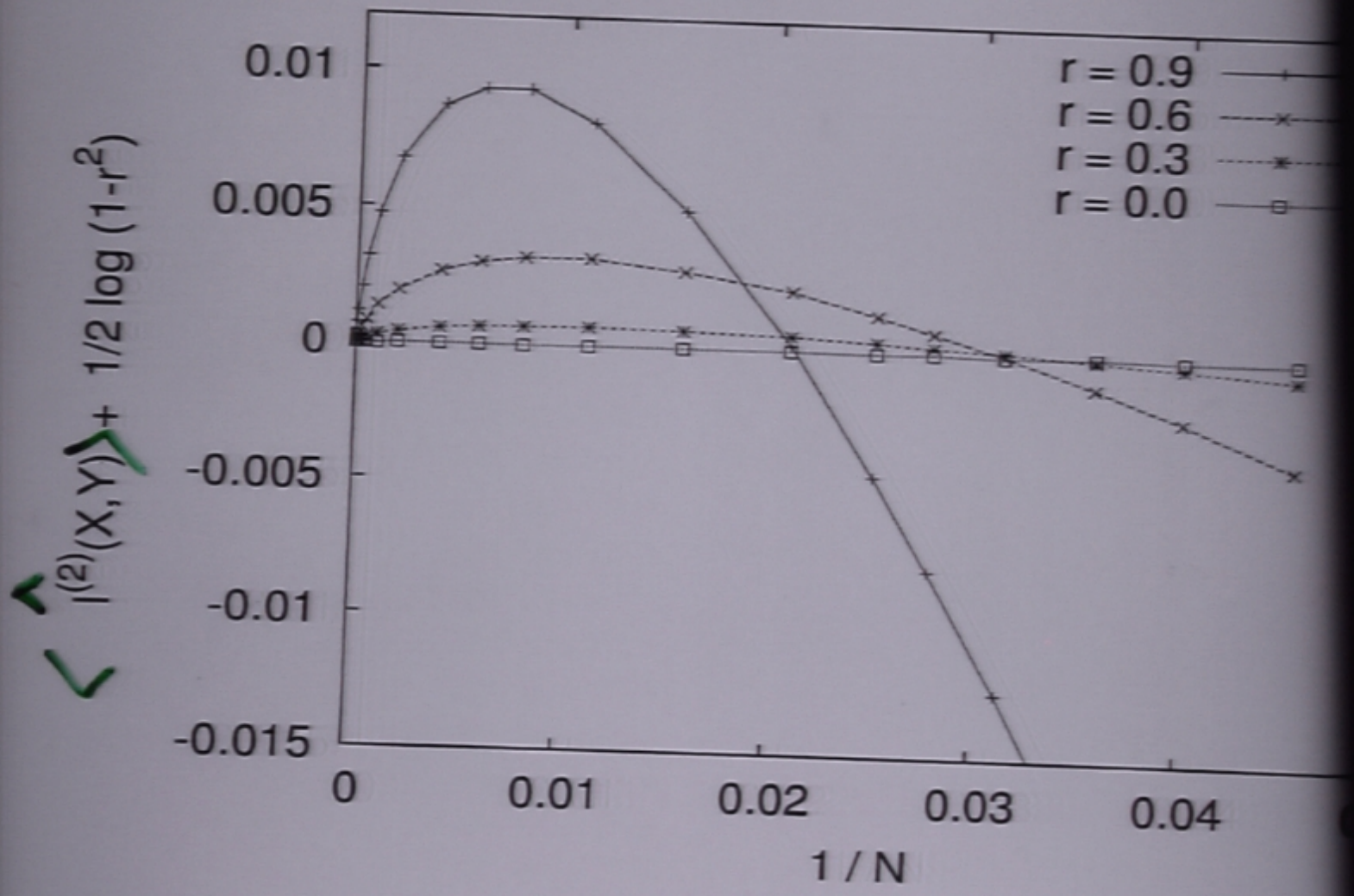
Both versions:

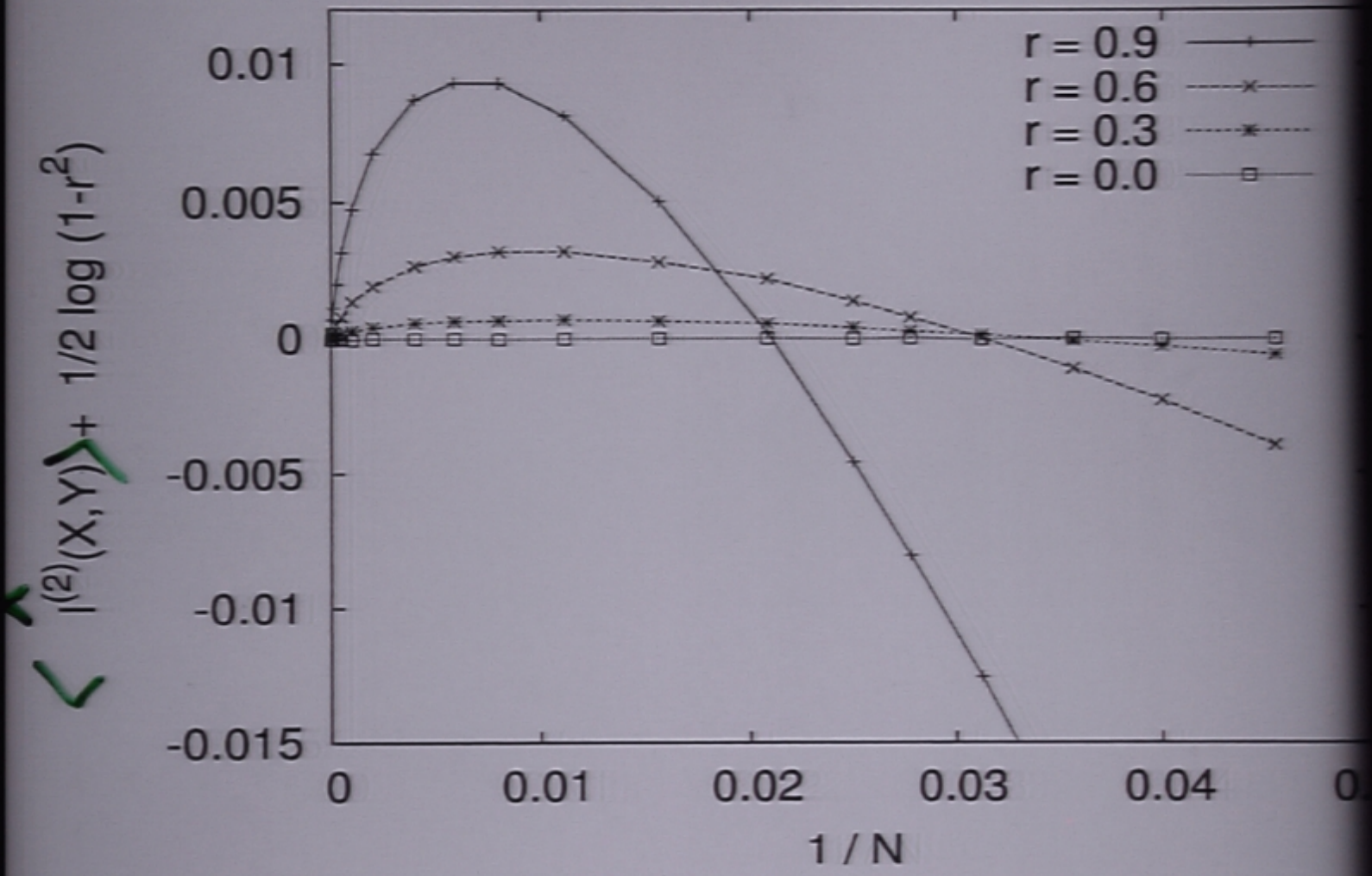
- small statistical errors
- fast, if efficient data structures used
- robust against outliers
- if degeneracies due to discretization
  ⇒ randomize by adding small noise
- small systematic errors
- no systematic errors, if X,Y independent (numerical)

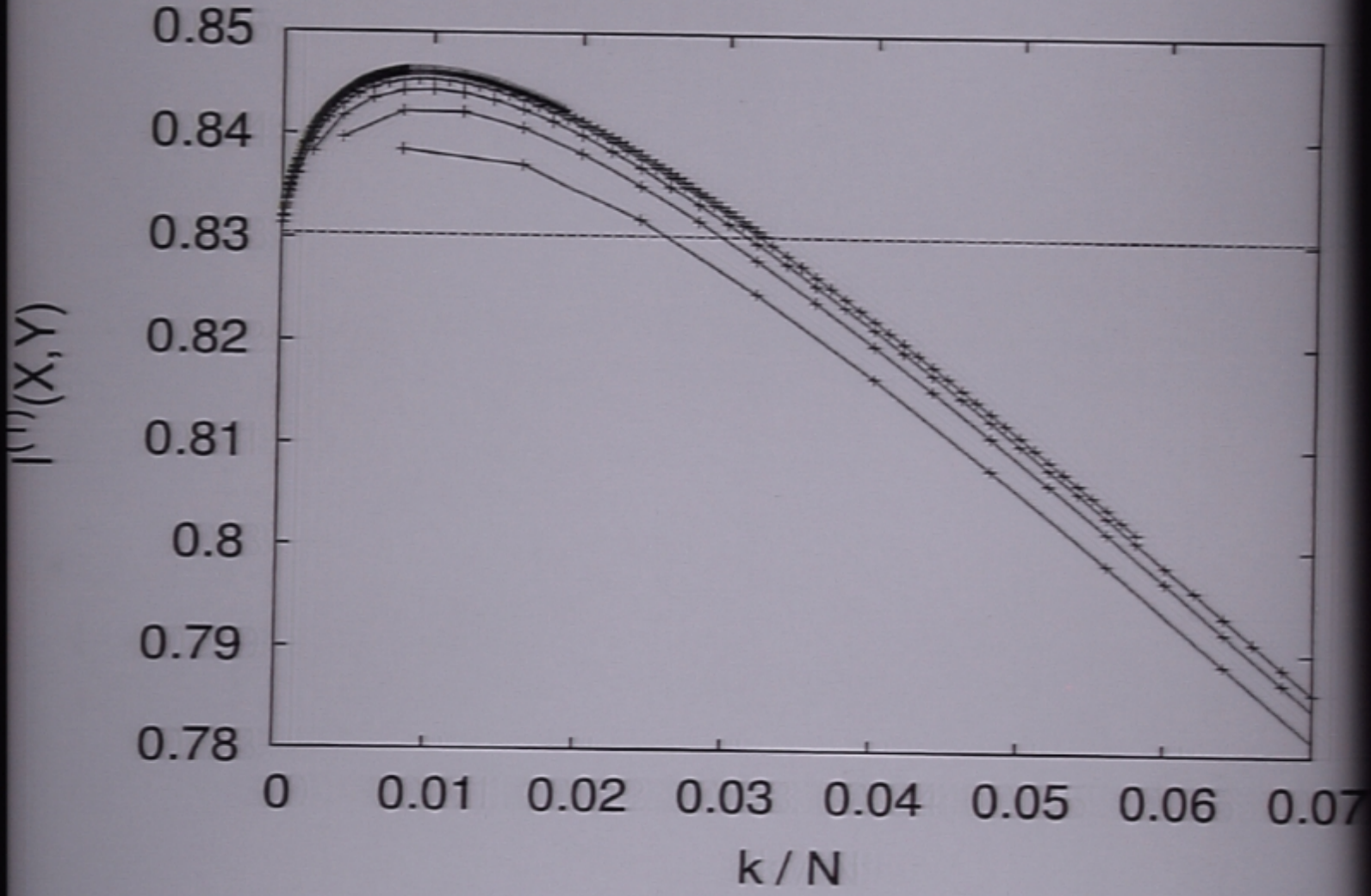

r = 0.9
r = 0.6
r = 0.3
r = 0.0
0.01
0.005
0
-0.005
-0.01
-0.015
0
0.01
0.02
0.03
0.04
1 / N
$I^{(2)}(X,Y) + 1/2 \log(1-r^2)$

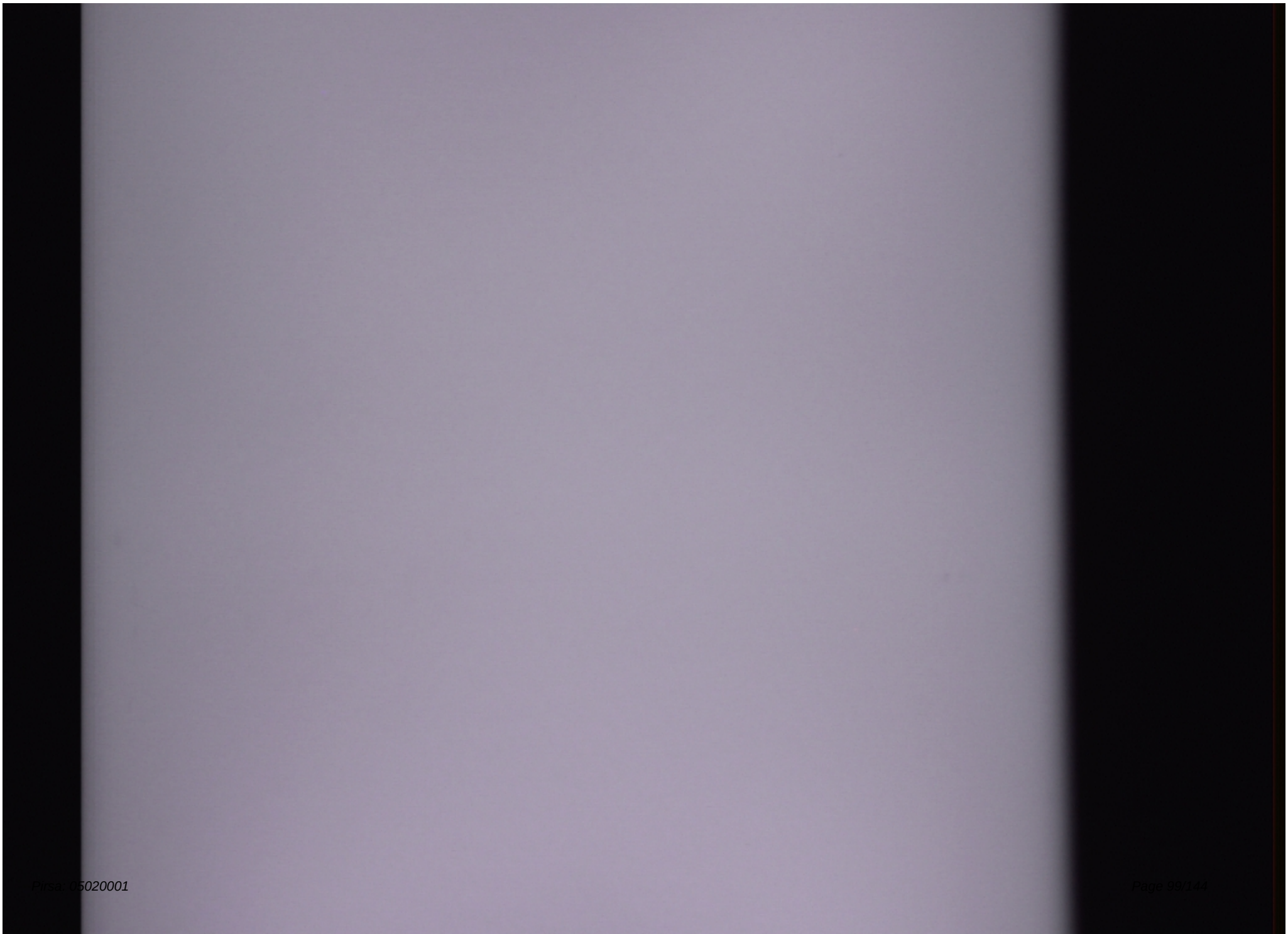

r = 0.9
r = 0.6
r = 0.3
r = 0.0
0.01
0.005
0
-0.005
-0.01
-0.015
0
0.01
0.02
0.03
0.04
1 / N
$\langle I^{(2)}(X,Y) \rangle_+ + 1/2 \log(1-r^2)$

$\langle \hat{I}^{(2)}(X,Y) \rangle + 1/2 \log (1-r^2)$
-0.015
-0.01
-0.005
0
0.005
0.01
0
0.01
0.02
0.03
1 / N
r = 0.9
r = 0.6
r = 0.3
r = 0.0


r = 0.9
r = 0.6
r = 0.3
r = 0.0
0.01
0.005
0
-0.005
-0.01
-0.015
0
0.01
0.02
0.03
0.04
1 / N
< I(2)(X,Y) >+ 1/2 log (1-r2)


r = 0.9
r = 0.6
r = 0.3
r = 0.0
0.01
0.005
0
-0.005
-0.01
-0.015
0
0.01
0.02
0.03
0.04
1 / N
< I(2)(X,Y) > + 1/2 log (1-r2)

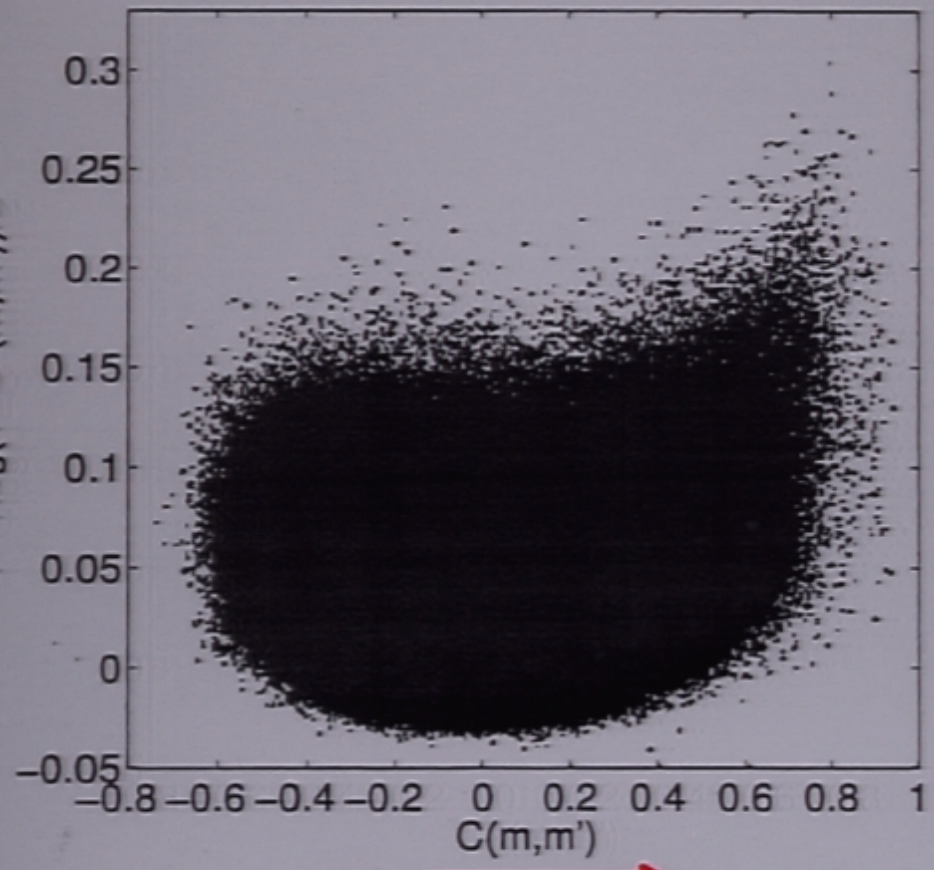

0.85
0.84
0.83
0.82
0.81
0.8
0.79
0.78
0
0.01
0.02
0.03
0.04
0.05
0.06
0.07
k / N



$m' = 1, \ldots 6000$ ... in yeast

dependencies measured via expression ratios of 300 closely related genomes (mostly a few mutations)

deviation of $\hat{I}$ from lower bd.

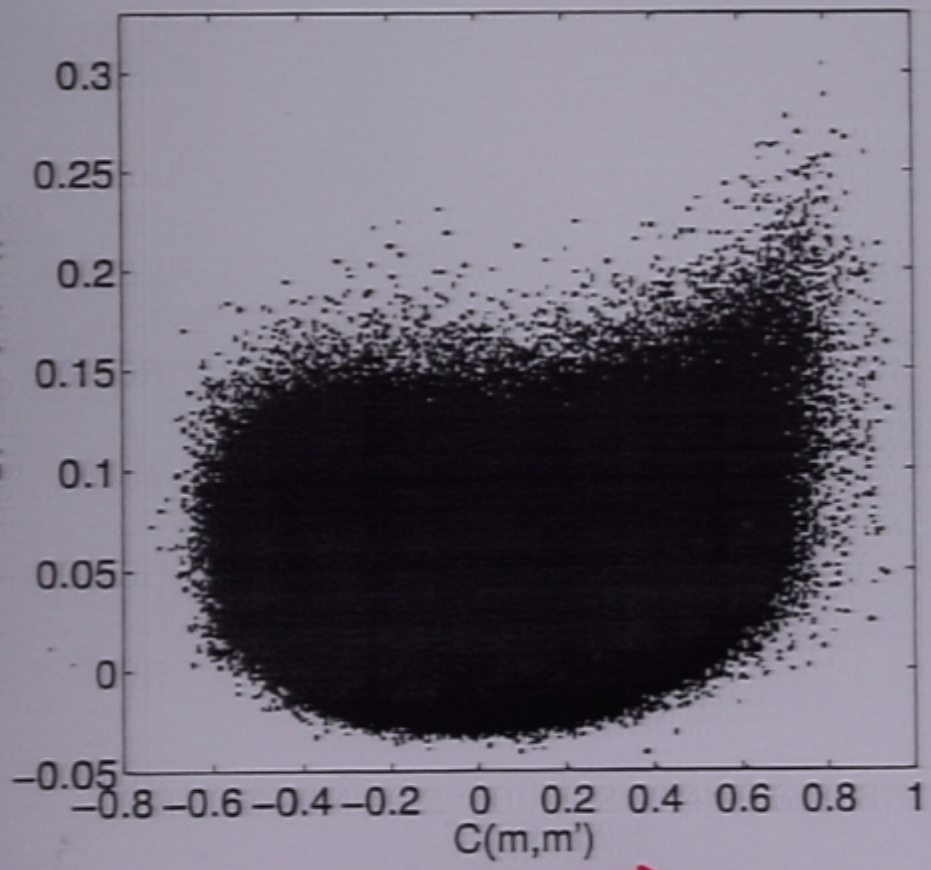



linear correl.

$m = 1, \ldots 6000$ } open reading frames
$m' = 1, \ldots 6000$ } in yeast

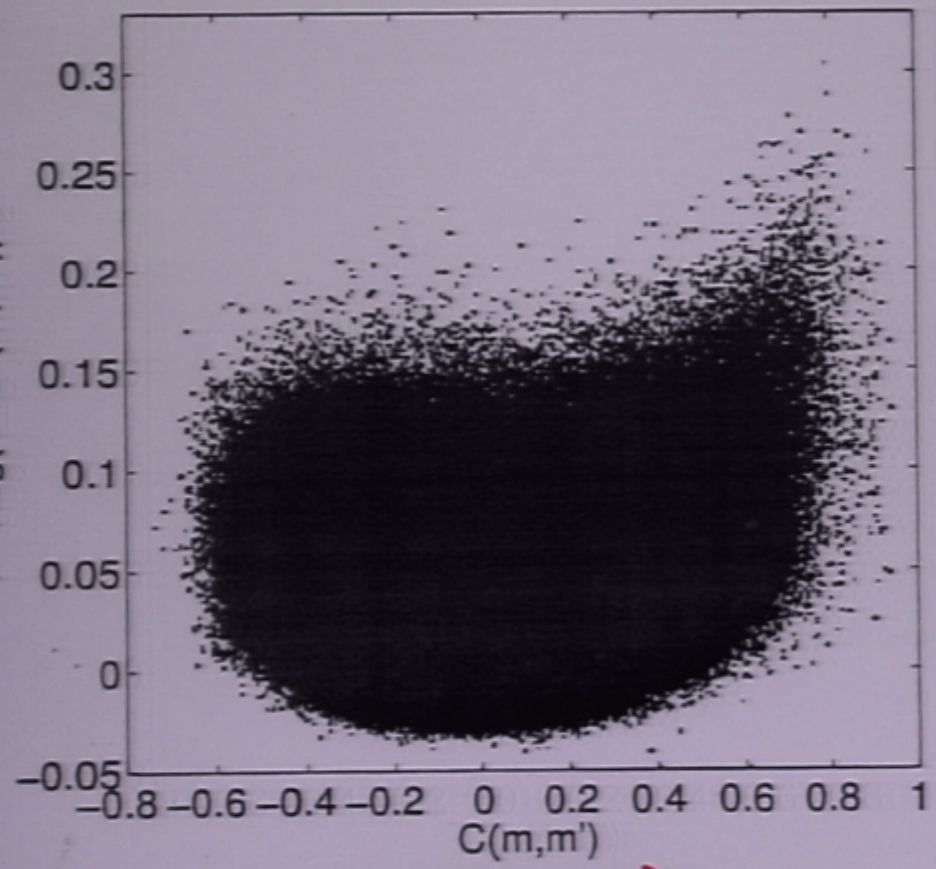
dependencies measured via expression ratios of 300 closely related genomes (mostly a few mutations)

deviation of $I^{(1)}$ from lower bd.



$m = 1, \ldots 6000$ } open reading frames
$m' = 1, \ldots 6000$ } in yeast

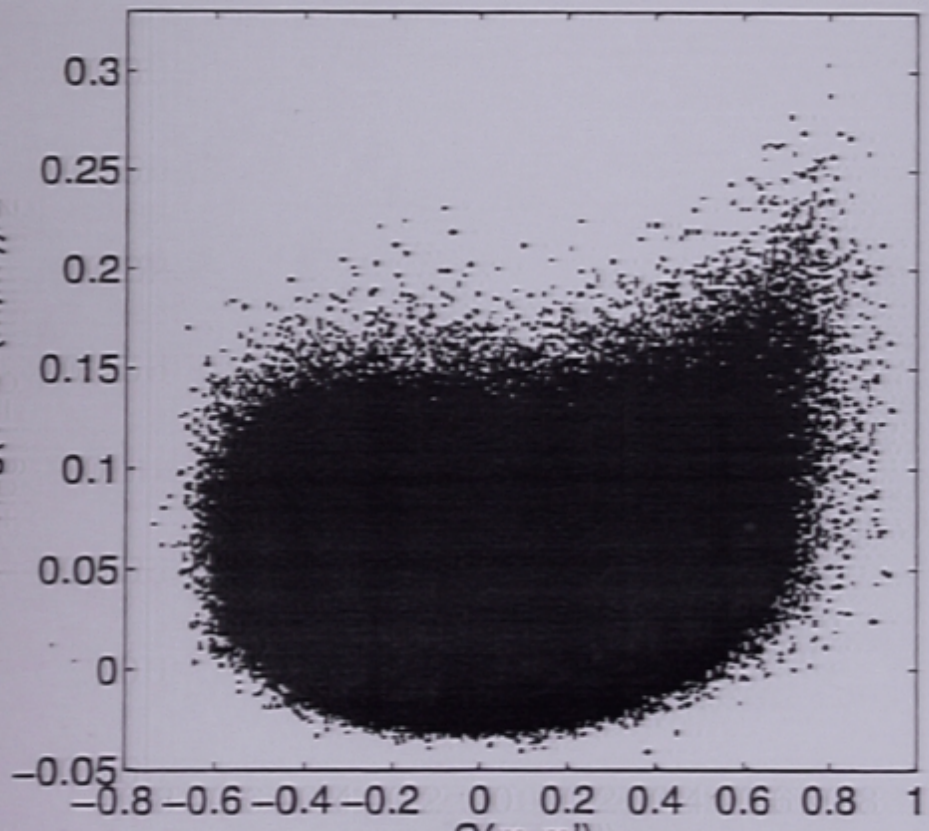
dependencies measured via expression ratios of 300 closely related genomes (mostly a few mutations)



$m = 1, \ldots 6000$
$m' = 1, \ldots 6000$
} open reading frames in yeast

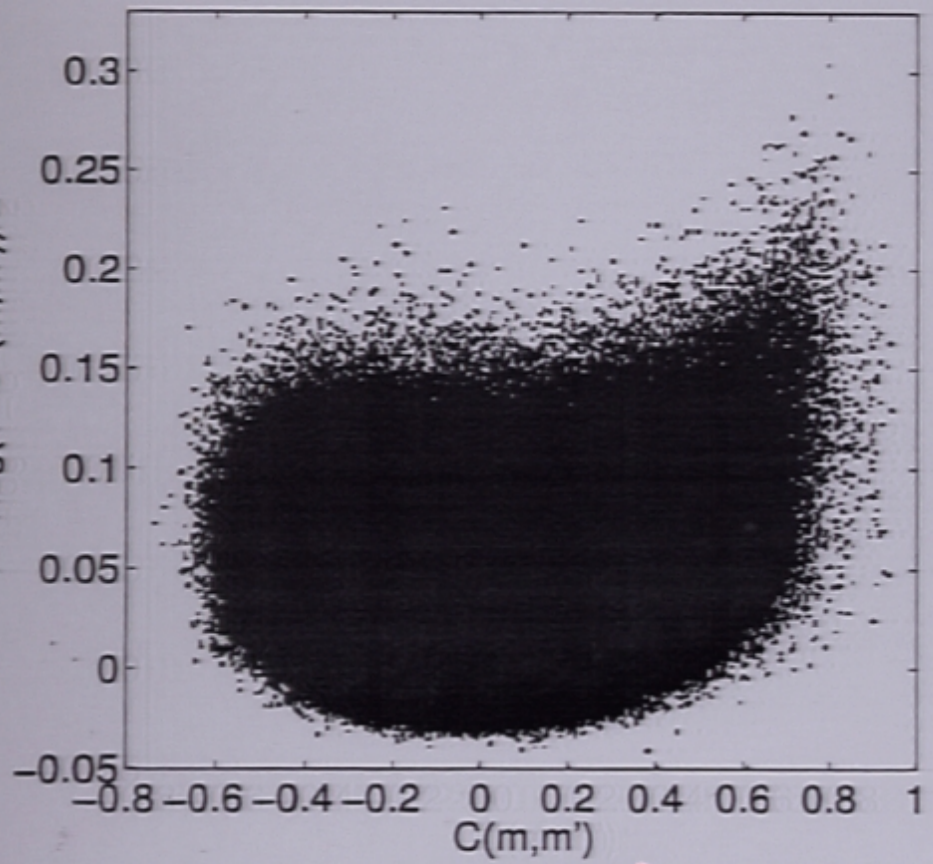
dependencies measured via expression ratios of 300 closely related genomes (mostly a few mutations)

deviation of $\hat{I}$ from lower bd.



dependencies measured via expression ratios of 300 closely related genomes (mostly a few mutations)

deviation of $\hat{I}$ from lower bd.




linear corrd. coeff.

Mutual Information based

Least dependent

~~Independent~~ Component

Analysis

(MILCA)

Sciences :

measured data
= superposition of several sources

Examples :

- signal + noise
- several speakers ("party problem")
- partial overlap between cellular phone freq. bands
- spectra emitted from chemical mixtures

Can sources be disentangled,
just from the observed data
alone ??

(no a priori knowledge of
how they were mixed,
no knowledge about number
of different sources, ... )

"blind source separation"
BSS

Only possible, if

Can sources be disentangled,
just from the observed data
alone ??

(no a priori knowledge of
how they were mixed,
no knowledge about number
of different sources, ... )

"blind source separation"

BSS

Only possible, if

no knowledge about number of different sources, ... )

"blind source separation"

BSS

Only possible, if

- sources have different statistics
- or/and are independent

# Simplest possibility:

- linear superposition
- statistically strictly independent sources
- # sources = # channels
- different "training tuples" (ie., signal vectors at diff. times) are i.i.d

# Simplest possibility:

- linear superposition
- statistically strictly independent sources
- # sources = # channels
- different "training tuples" (ie., signal vectors at diff. times) are i.i.d

$$x_i(t) = \sum_{k=1}^{n} W_{ik}\, s_k(t) \qquad i = 1, \dots n$$

measure-ments

mixing matrix (instantan., stationary, non-singular)

sources (i.i.d., independant)

$W = ?$

$s = ?$

$$x_i(t) = \sum_{k=1}^{n} W_{ik}\, s_k(t) \qquad i = 1, \ldots n$$

measure-ments

mixing matrix (instantan., stationary, non-singular)

sources (i.i.d., independent)

$W = ?$

$s = ?$

obvious formal sol'n to simplest problem:

$$\vec{s}(t) = W^{-1}\,\vec{x}(t)$$

such that

$s_i(t)$, $s_k(t)$ independent $\forall\ (i,k)$

What does this mean?

E.g. $\langle s_i\, s_k\rangle - \langle s_i\rangle\langle s_k\rangle = 0$ (uncorrelated)

$C_{ij}(x) \equiv \langle x_i\, x_j\rangle - \langle x_i\rangle\langle x_j\rangle$

$$\vec{s}(t) = W^{-1}\,\vec{x}(t)$$

such that

$s_i(t)$ , $s_k(t)$ independent $\forall\ (i,k)$

What does this mean?

E.g. $\langle s_i\, s_k\rangle - \langle s_i\rangle\langle s_k\rangle = 0$ (uncorrelated)

$$C_{ij}(x) \equiv \langle x_i\, x_j\rangle - \langle x_i\rangle\langle x_j\rangle$$

$W^{-1}\, C(x)\, W \doteq$ diagonal

$\Rightarrow$ PCA

If only covariance matrix is used as dependency measure ⇒ PCA

BUT: uncorrelated ≠ independent!

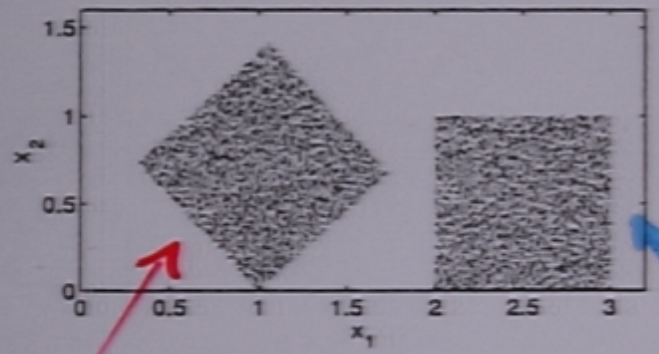



**Abbildung 3.1:** Zwei gleichförmig verteilte Zufallsvariable, wobei sich die linke von der rechten Abbildung nur durch eine Rotation unterscheidet. Beide Bilder stellen unkorrelierte Variable dar, aber nur rechts sind sie auch unabhängig.

x,y are uncorrelated

x,y are statistically independent

## ICA:

assumes same linear mixing ansatz as PCA, but

goes beyond 2nd order statistics for finding W

PCA does not give unique $A = W^{-1}$:

- permutation of indices
- re-scaling of axes

Use A such that $C(s) = \text{const} = \mathbb{1}$

⇒ "pre-whitening"

assumes same linear mixing ansatz as PCA, but

goes beyond 2nd order statistics

for finding W

PCA does not give unique $A = W^{-1}$:

- permutation of indices
- re-scaling of axes

Use A such that $C(s) = \text{const} = \mathbb{1}$

⇒ "pre-whitening"

ansatz as PCA, but

goes beyond 2nd order statistics

for finding W

PCA does not give unique $A = W^{-1}$:

- permutation of indices
- re-scaling of axes

Use A such that $C(s) = \text{const} = \mathbf{1}$

$\Rightarrow$ "pre-whitening"

- after pre-whitening, there is still an unspecified

## goes beyond 2nd order statistics for finding W

PCA does not give unique $A = W^{-1}$:

- permutation of indices
- re-scaling of axes

Use A such that $C(s) = \text{const} = \mathbf{1}$

$\Rightarrow$ "pre-whitening"

- after pre-whitening, there is still an unspecified rotation $\mathbb{R} \in O(n)$ free

ICA proper specifies R

up to

- permutation of sources
- common re-scaling of all sources

Jutten & Herault ~1980

Bell & Sejnowski ~1990

Cardoso ≥1990

⋮

A. Hyvärinen, J. Karhunen, E. Oja

"Independent Compon Anal"

- permutation of sources
- common re-scaling of all sources

Jutten & Herault ~1980

Bell & Sejnowski ~1990

Cardoso ≥1990

⋮

A. Hyvärinen, J. Karhunen, E. Oja
"Independent Compon Anal"
Wiley 2001

many public domain & proprietary

## MILCA

- Basic Milca:

  minimize $\hat{I}(s_1 \cdots s_n)$ by rotations in subspaces (after whitening)

  choose in each subspace optimal angle by fitting $\hat{I}$ with short Fourier series

- Significance test:

  check variation of $\hat{I}(s_1 \cdots s_n)$ with angles,

  to find angles which cannot be given unambiguously

  (no unique BSS possible)

- Basic Milca:

  minimize $\hat{I}(s_1 \cdots s_n)$ by rotations
  in subspaces (after whitening)

  choose in each subspace optimal
  angle by fitting $\hat{I}$ with short
  Fourier series

- Significance test:

  check variation of $\hat{I}(s_1 \cdots s_n)$
  with angles,
  to find angles which cannot be
  given unambiguously
  (no unique BSS possible)

- **Independence test :**

  check whether final

  $\hat{I}(s_i; s_j) \cong 0 \quad \forall\, i, j$

- **Multi-variate MiLCA**

  ⇒ if $\exists$ cluster $\{s_a \cdots s_m\}$

  of dependent sources,

  then consider this as <u>one</u>

  <u>k-variate source</u>

  Use grouping property for a complete clustering of all sources ("mutual information based clustering", MIC :

if $\exists$ cluster $\{S_1 \cdots S_n\}$
of dependent sources,
then consider this as one
k-variate source

Use grouping property for a
complete clustering of all sources
("mutual information based
clustering", MIC :
A. Kraskov et al)

reconstruct contributions of
each relevant cluster to every
channel

- MiLCA with instantaneous mixing time correlated signals

$x_i(t)$, $x_i(t')$ not indep.

$\vec{s}(t) = A\,\vec{x}(t)$ as before (no delay)

Df:

$\underline{s}_i(t) = (s_i(t-d\tau) \cdots s_i(t))$

d+1 - dim. delay vectors

$$\hat{I}(\underline{s}_1(t) \cdots \underline{s}_n(t)) \stackrel{!}{=} \min$$

minimize also dependence between time-shifted sources !

- MILCA with delayed mixing & delayed $\hat{I}$

our best algorithm !

$$\hat{s}_i(t) = \sum_{j=1}^{n} \sum_{k=0}^{d} a_{ij}^{k} \; x_j(t - k\tau)$$

$i = 1 \dots nd$

$$\hat{I}(\hat{s}_1 \dots \hat{s}_{nd}) \stackrel{!}{=} \min$$

ECG of pregnant woman

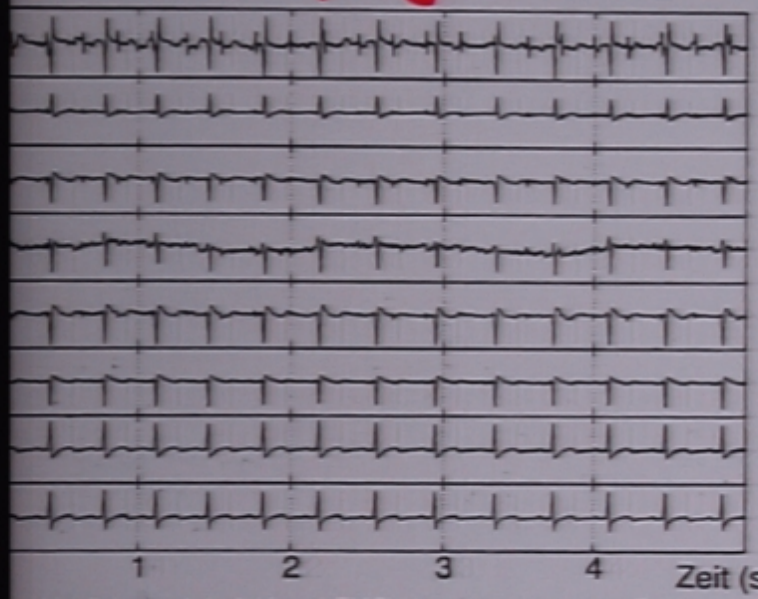



**Abbildung 4.1:** 8-Kanal EKG einer schwangeren Frau.

basic MILC

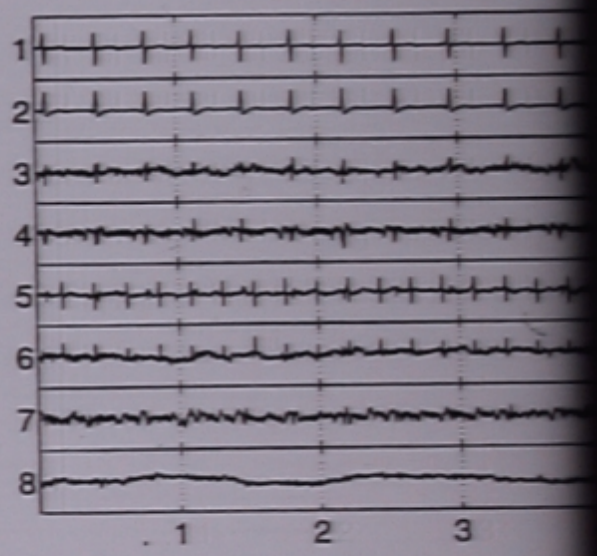
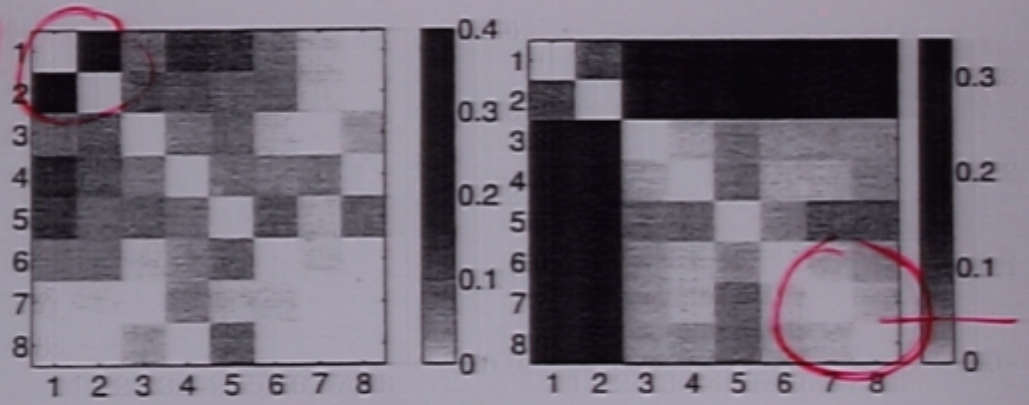



**Abbildung 4.2:** MILCA-Ausgabe: Komponenten nach der M
für die EKG-Signale aus Abb.(4.1).

strongly corrd. cluster




no u
dec
of

**Abbildung 4.3:** Linkes Bild: $\hat{I}$ zwischen allen paarweisen Kombinationen des Signals aus Abb.(4.2). Rechtes Bild: Quadratwurzel der Variabilität $\sigma_{ij}$ von $\hat{I}_{ij}(\phi)$. In beiden Bildern ist

ECG of pregnant woman

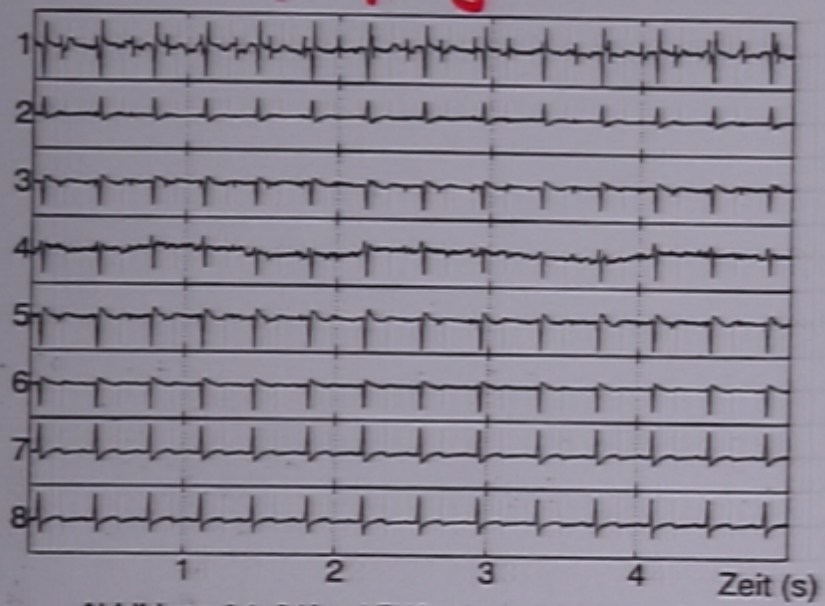



**Abbildung 4.1:** 8-Kanal EKG einer schwangeren Frau.

basic

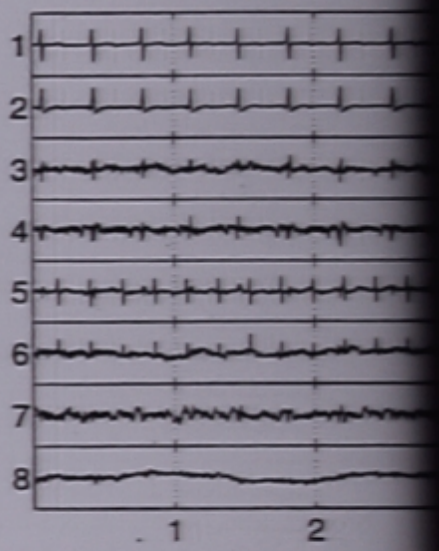



**Abbildung 4.2:** MILCA-Ausgabe: Komponenten für die EKG-Signale aus Abb.(4.1).

strongly corrd. cluster

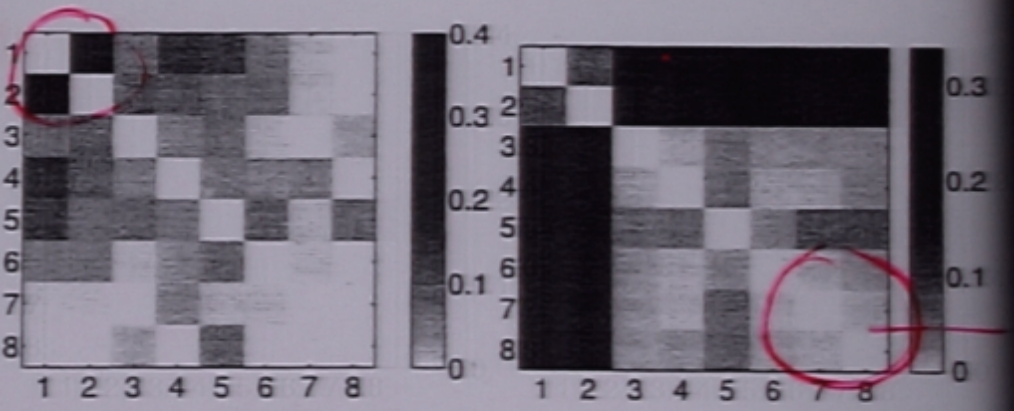



**Abbildung 4.3:** Linkes Bild: $\tilde{I}$ zwischen allen paarweisen Kombinationen des Signals a
Abb.(4.2). Rechtes Bild: Quadratwurzel der Variabilität $\sigma_{ij}$ von $\tilde{I}_{ij}(\phi)$. In beiden Bildern
die Diagonale auf Null gesetzt.

gnant woman


einer schwangeren Frau.

basic MiLCA

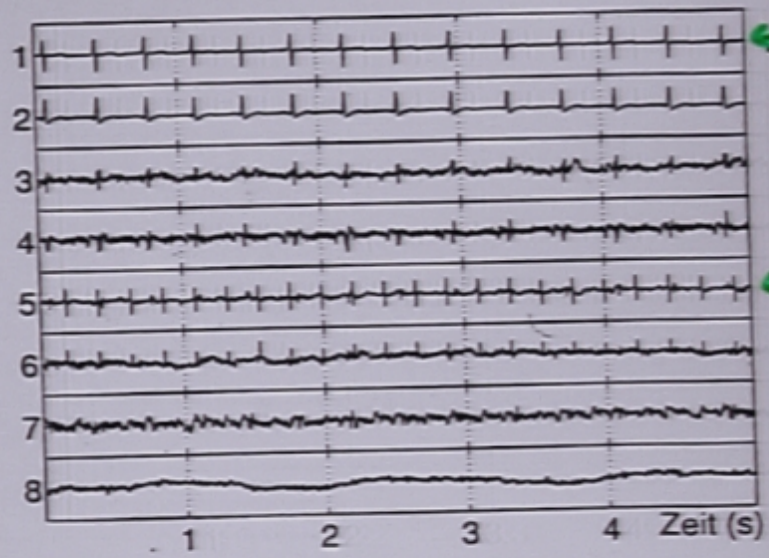
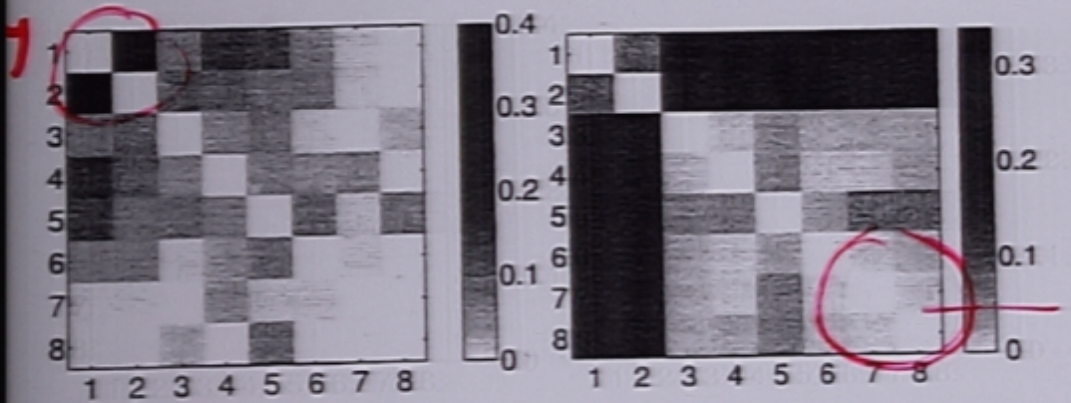



**Abbildung 4.2:** MILCA-Ausgabe: Komponenten nach der Minimierung von $I(X_1 \ldots X_8)$ für die EKG-Signale aus Abb.(4.1).



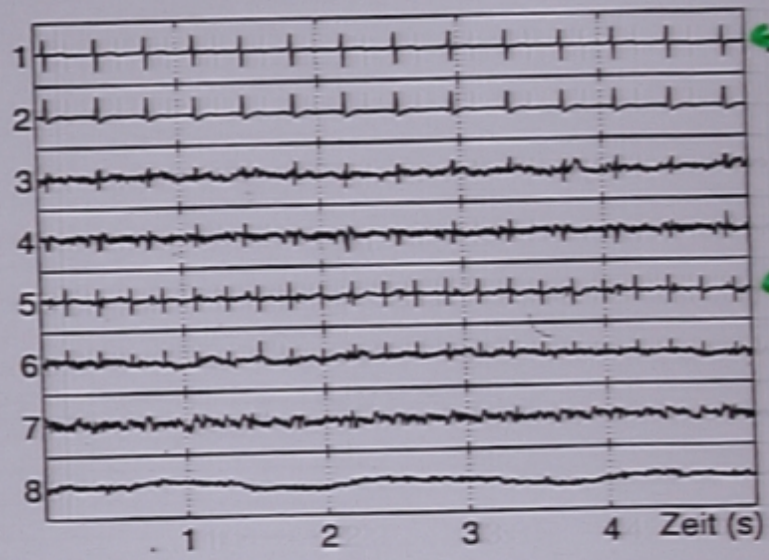

no unique decomposition of noise sources

**Abbildung 4.3:** Linkes Bild: $\tilde{I}$ zwischen allen paarweisen Kombinationen des Signals aus Abb.(4.2). Rechtes Bild: Quadratwurzel der Variabilität $\sigma_{ij}$ von $\tilde{I}_{ij}(\phi)$. In beiden Bildern ist die Diagonale auf Null gesetzt.

gnant woman


einer schwangeren Frau.

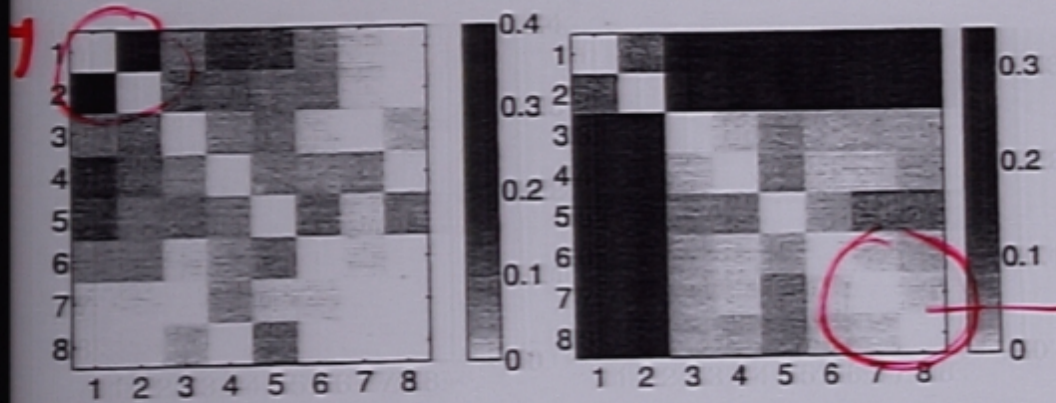
basic MiLCA




← mother

← Fetus

**Abbildung 4.2:** MILCA-Ausgabe: Komponenten nach der Minimierung von $I(X_1 \ldots X_8)$ für die EKG-Signale aus Abb.(4.1).




no unique decomposition of noise sources

**Abbildung 4.3:** Linkes Bild: $\tilde{I}$ zwischen allen paarweisen Kombinationen des Signals aus Abb.(4.2). Rechtes Bild: Quadratwurzel der Variabilität $\sigma_{ij}$ von $\tilde{I}_{ij}(\phi)$. In beiden Bildern ist die Diagonale auf Null gesetzt.

After delay embedding (3 times):

8 channels → 24 channels

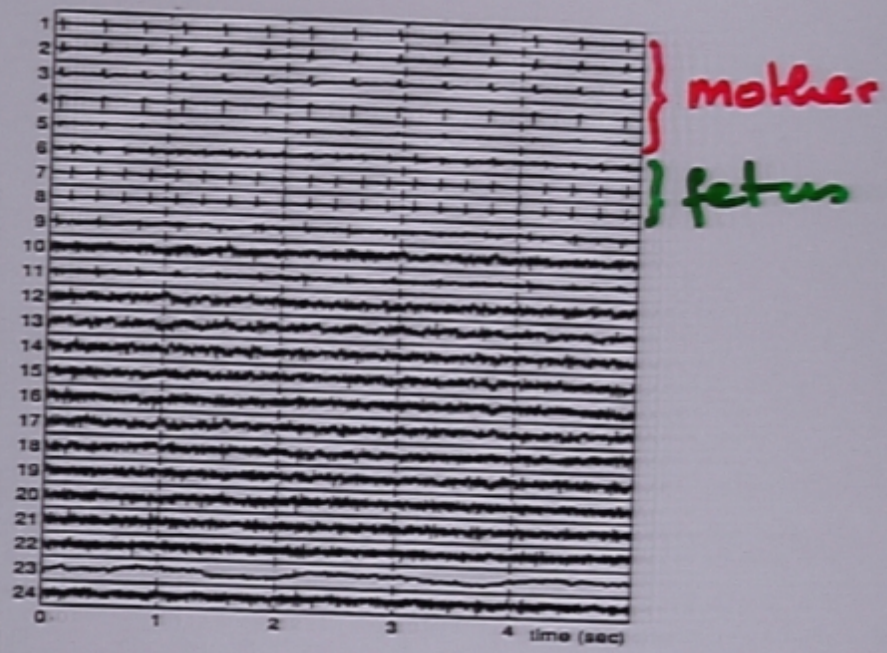



FIG. 17: MILCA output from the embedded eight channel ECG (k=100, m=3)


FIG. 18: Dendrogram for Fig. 17. Heights of each cluster correspond to $I(X_i, X_j)$ of the cluster $ij$ (k=3).

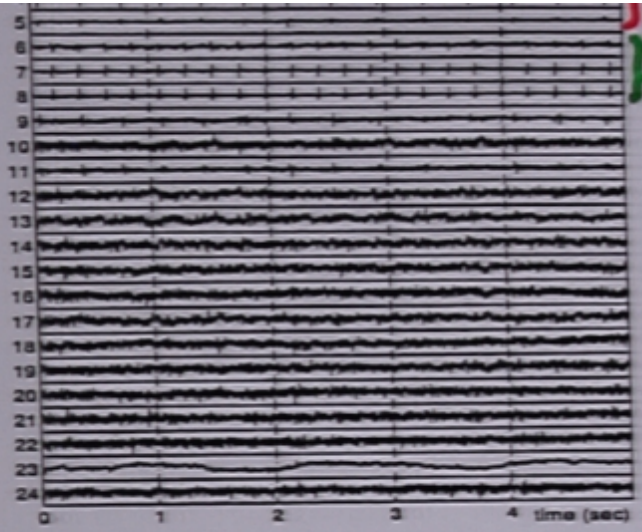



FIG. 17: MILCA output from the embedded eight channel ECG (k=100, m=3)


FIG. 18: Dendrogram for Fig. 17. Heights of each cluster correspond to $I(X_i, X_j)$ of the cluster $ij$ ($k$=3).

8 channels → 24

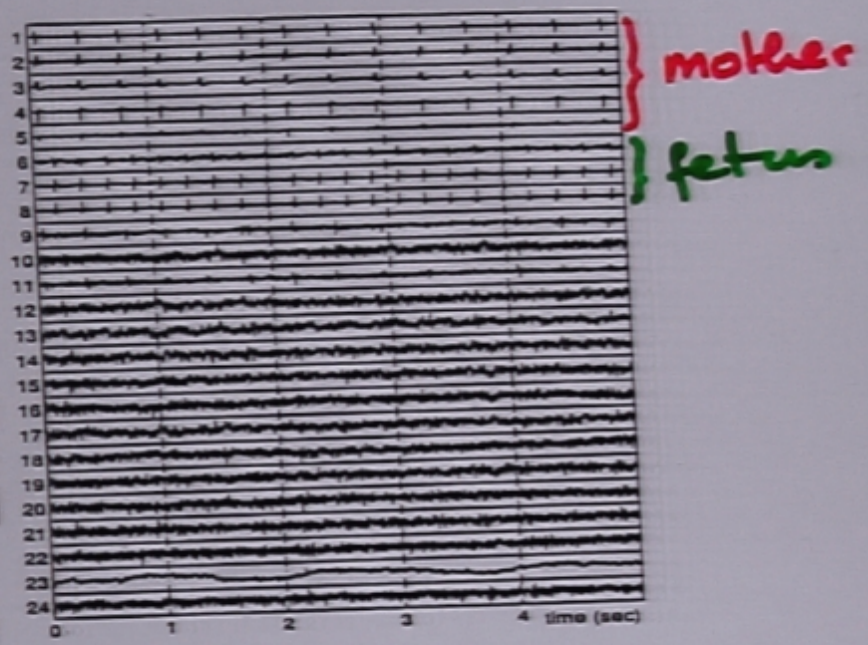



FIG. 17: MILCA output from the embedded eight channel ECG (k=100, m=3)


FIG. 18: Dendrogram for Fig. 17. Heights of each cluster correspond to $I(X_i, X_j)$ of the cluster $ij$ ($k$=3).

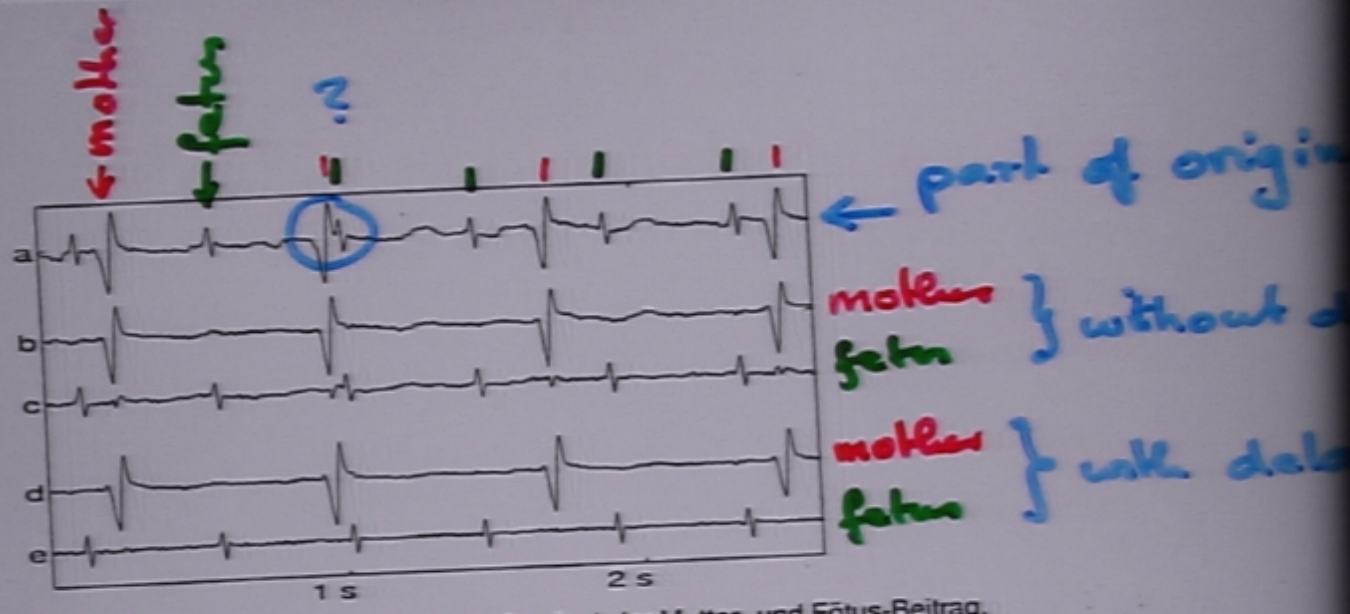



**Abbildung 4.10:** (a) Ausschnitt des originalen EKGs; (b,c) der Mutter- und Fötus-Beitrag, erhalten ohne Verzögerungseinbettung; (d,e) beide Beiträge bei Verwendung der Verzögerungseinbettung.

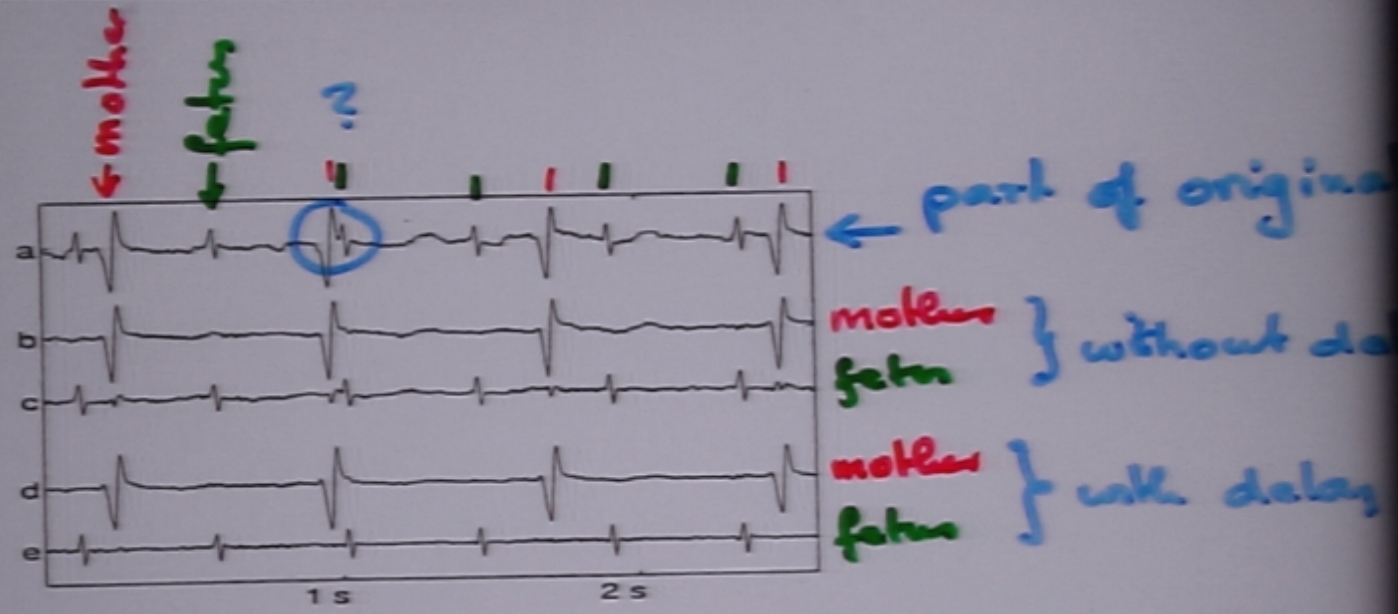



**Abbildung 4.10:** (a) Ausschnitt des originalen EKGs; (b,c) der Mutter- und Fötus-Beitrag, erhalten ohne Verzögerungseinbettung; (d,e) beide Beiträge bei Verwendung der Verzögerungseinbettung.