April 12, 2005

Middlesex University, London NW4 4BT, UK
School of Computing Science
Roman V. Belavkin (R.Belavkin@mdx.ac.uk)

Common Lisp Implementation of FastICA
\[
X \approx \mathbf{WV}
\]

Find \( \mathbf{W} \) such that

1. \( \mathbf{W}^{-1} \approx \mathbf{V} \)

2. All \( \forall i \) but one \( P(x_i) \) are non-Gaussian

\[
\left( \begin{array}{c} \mathbf{W}x_1 \\ \vdots \\ \mathbf{W}x_n \end{array} \right) \sim \left( \begin{array}{c} x_1 \\ \vdots \\ x_n \end{array} \right)
\]

Assuming

\[
X \mathbf{W} = \lambda
\]

\[
L(\mathbf{W}x_1, \ldots, \mathbf{W}x_n) = \lambda, \quad L(x_1, \ldots, x_n) = X
\]

**BLIND SOURCE SEPARATION (or ICA)**
Common Lisp Implementation

ICAI.Iisp (5K) + PCA.Iisp (8K) + Matrices.Iisp (10K) = 23K of code:

(defun one-unit-fica (X &key......)
  (do* ((m (num-rows X))
        (ws (list w))
        (w (find-one-weight X ws))
        (W (make-array (list m m))))
       ((= (length ws) m)
        (let ((W (make-array (list m m))))
          (dotimes (i m W)
            (dotimes (j m)
              (setf (aref W i j)
                (aref (nth j ws) i 0))))))))
MIXING SOURCES

\[ Y_0 = W_0 \times X_0 \]

\[ Y_1 = W_1 \times X_1 \]

\[ Y_2 = W_2 \times X_2 \]

\[ = W \times X \]

Original Independent Sources

Mixed Signals
DEMIXING SIGNALS

Reconstructed Independent Components

$A \times$ Mixed Signals
RECONSTRUCTING MIXTURE

Reconstructed Independent Components

\[ \text{Reconstructed Mixture} = \mathbf{A}^{-1} \times \text{Reconstructed Independent Components} \]
• Use ICA output as an input to Bayesian learning agents (perception with reduction of dimensions and minimal loss of information):

\[ P(Y | S, U) = \alpha P(Y, S, U) \]

\[ X \rightarrow S \in \mathbb{R}^m \]

set of actions

reduce dimensions (ICA)

\[ Y \in \mathbb{R}^m \]

perception inputs

\[ X \in \mathbb{R}^n \]

perception inputs

• Encourage MSc students to use ICA in their projects in Business Information Systems programme.