Collective classification in network data

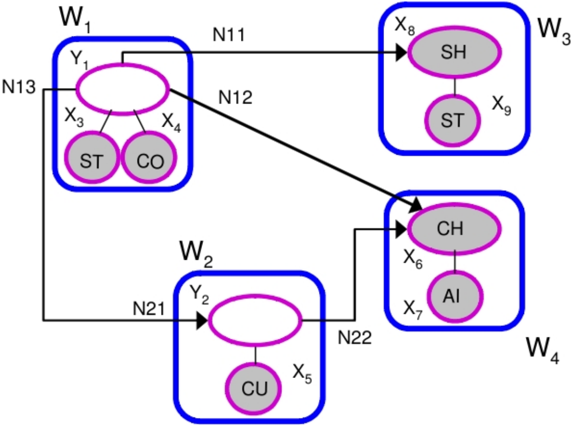
Seminar on graphs, UCSB 2009

Outline

1. Problem
2. Methods
   - Local methods
   - Global methods
3. Experiments

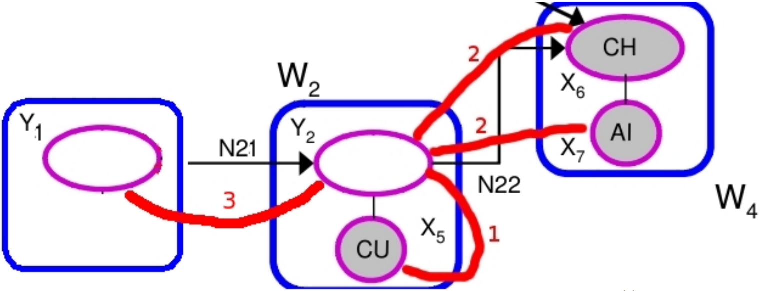
Outline

Example


W1
Y1
N11
N13
N12
X3
X4
ST
CO
X8
SH
ST
X9
W3
CH
X6
AI
X7
W4
W2
Y2
N21
N22
CU
X5

Correlations

1. Correlation between label and attributes (classic IR hypothesis)
2. Correlation between label and labels and attributes of known neighbors
3. Correlation between labels of unknown neighbors



Collective classification (CC)

Definition

CC: Combined classification of inter-linked objects using label-attribute correlations and label-label neighbor correlations.

A major difference to general classification is that inference for all unknown instances is simultaneous.

Inference

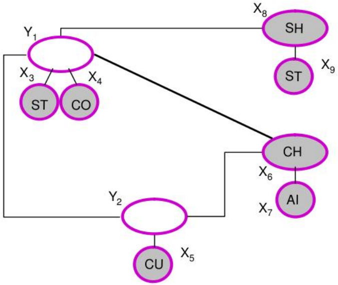
Definition

Given a joint distribution of the unknown labels, compute the marginal distribution for a single node's label.

1. Exact inference is intractable for arbitrary networks.
2. Algorithms: variable elimination, junction tree.
3. Most research is focused on approximate inference.

A more formal view on the problem

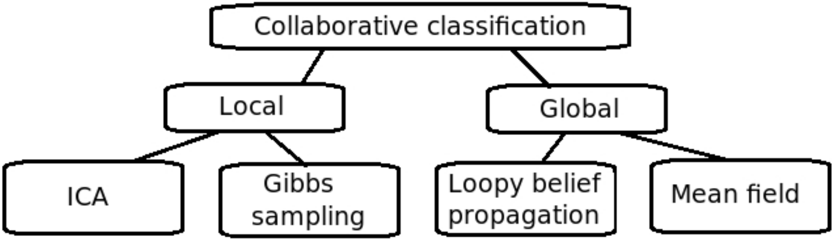
1. The network structure is modeled as a graph G=(V,E).
2. Each node is a variable defined over a given domain.
3. V contains two types of variables: X and Y
4. Goal: Label the nodes in Y



Outline

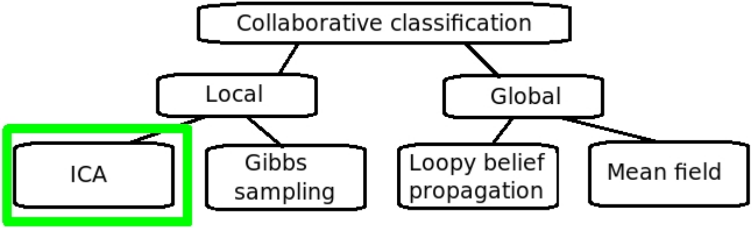
1. Problem
2. Methods
   - Local methods
   - Global methods
3. Experiments

Local and global



Outline

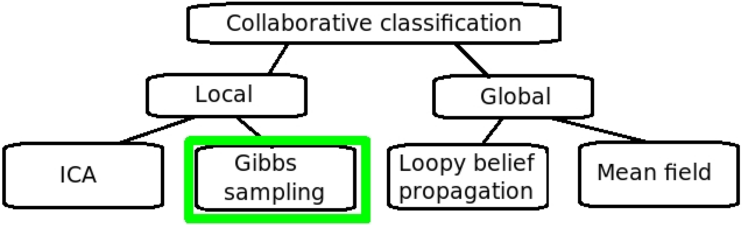
Iterative classification algorithm(ICA)



ICA mechanics

1. Classify a node Y_i based on its neighbors N_i
2. Use a local classifier $f(N_i)$ to compute the best value of y_i
3. Iteratively apply to all Y_i using the best estimates of unknowns in N_i
4. Use the labeling that stabilizes over time

Gibbs sampling(GS)



Gibbs sampling - basic idea

1. Sample from a multivariate joint distribution (unknown explicitly)
2. Generates a series of samples based on conditional distributions of each variable
3. Example: Sample values from $f(X, Y)$
   1. Start with initial $X = x_0$
   2. Sample $y_0 = p(Y|X = x_0)$
   3. Sample $x_1 = p(X|Y = y_0)$...
   4. $(x_0, y_0), (x_1, y_1)$... are samples from $p(X, Y)$ if $p(*|*)$ are the true conditionals
4. Simpler to sample from conditional distributions than to integrate over a joint (especially if the latter is unavailable)

Gibbs sampling for CC

1. The joint distribution is $p(Y_1, Y_2, \ldots Y_n)$
2. Assume that we know the conditionals $p(Y_k | Y_1 = y_1, \ldots Y_{k-1} = y_{k-1}, Y_{k+1} = y_{k+1} \ldots)$
3. Perform GS and estimate the marginals $p(Y_i), Y_i \in Y$ based on the samples

Assume we know the conditionals?

1. Assume we can **estimate** the conditional $p(Y_i|N_i)$ using a local classifier
2. Assume independence of indirect neighbors $p(Y_i|N_i) = p(Y_i|Y)$
3. No guarantee that the estimated conditionals are the true conditionals

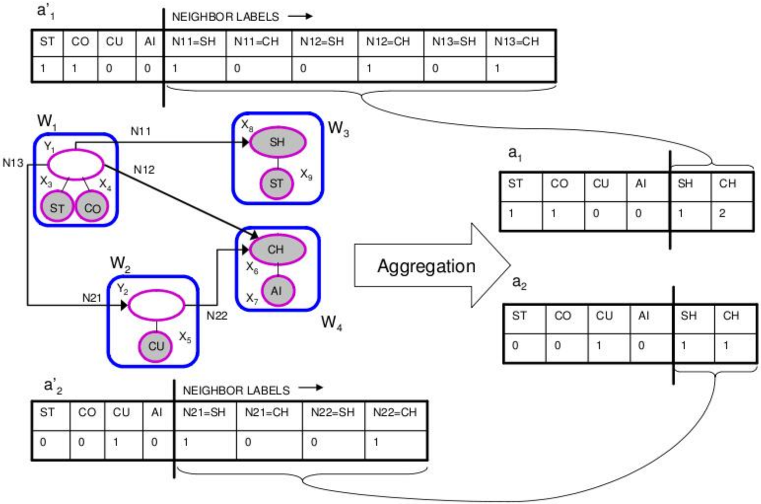
The mechanics of GS for CC

1. Initialize assignments of Y_i
2. Perform a "burn-in" number of sample steps
3. Sample and count label assignments
4. Estimate marginals based on counts.
   Decide on labels.

Challenges of ICA and GS

1. Feature construction for local classifiers
    1. Classifiers normally require fixed-length FVs
    2. Choice of aggregation - max, count, exists, etc.
2. Local classifiers(Decision trees, Log. Regression, SVM, etc.). Training.
3. Nodes ordering - robust to simple random, based on label diversity etc.
4. Performance (running time)

Feature construction



Aggregation: count, avg, exists, proportion, graph based, etc.

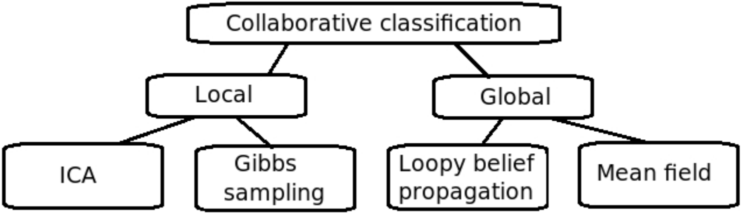
Local classifiers

Reference	local classifier used
Neville & Jensen [44]	naïve Bayes
Lu & Getoor [35]	logistic regression
Jensen, Neville, & Gallagher [25]	naïve Bayes, decision trees
Macskassy & Provost [36]	naïve Bayes, logistic regression, weighted-vote relational neighbor, class distribution relational neighbor
McDowell, Gupta, & Aha [39]	naïve Bayes, k-nearest neighbors

Outline

1. Problem
2. Methods
   - Local methods
   - Global methods
3. Experiments

Global methods



Additional notation

1. L is the set of labels, $G(V, E)$ is the network of objects
2. Three types of clique potentials(distributions)
3. ψ_i for each $Y_i \in Y$ is a mapping $\psi_i : L \rightarrow R^+$
4. ψ_{ij} for each $(Y_i, X_j) \in E$ is a mapping $\psi_{ij} : L \rightarrow R^+$
5. η_{ij} for each $(Y_i, Y_j) \in E$ is a mapping $\eta_{ij} : LxL \rightarrow R^+$

Back to our example

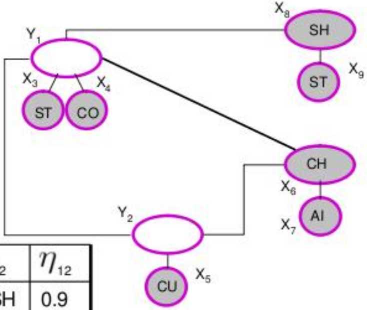
y_1	Ψ_{13}
SH	0.6
CH	0.4

y_1	Ψ_{16}
SH	0.1
CH	0.9

y_1	Ψ_{18}
SH	0.8
CH	0.2

y_1	Ψ_1
SH	0.5
CH	0.5

y_1	Ψ_{14}
SH	0.4
CH	0.6



y_1	y_2	η_{12}
SH	SH	0.9
SH	CH	0.1
CH	SH	0.1
CH	CH	0.9

y_2	Ψ_2
SH	0.5
CH	0.5

y_2	Ψ_{26}
SH	0.1
CH	0.9

y_2	Ψ_{25}
SH	0.1
CH	0.9

Just a little bit more notation

1. "Known" potential of a label y_i
   $\phi_i(y_i) = \psi_i(y_i) \sum_{(Y_i, X_j) \in E} \psi_{ij}(y_i)$
2. It is computed without considering "unknown" neighbors

Back to our example

y_1	Ψ_{13}
SH	0.6
CH	0.4

y_1	Ψ_{16}
SH	0.1
CH	0.9

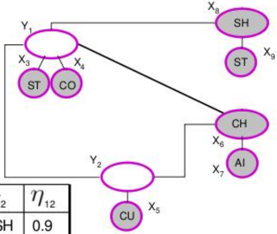
y_1	Ψ_{18}
SH	0.8
CH	0.2

$\Phi_1 = \Psi_1 * \Psi_{13} * \Psi_{14} * \Psi_{16} * \Psi_{18} =$

y_1	Φ_1
SH	0.0096
CH	0.0216

y_1	Ψ_1
SH	0.5
CH	0.5

y_1	Ψ_{14}
SH	0.4
CH	0.6



$\Phi_2 = \Psi_2 * \Psi_{25} * \Psi_{26} =$

y_2	Φ_2
SH	0.005
CH	0.405

y_1	y_2	η_{12}
SH	SH	0.9
SH	CH	0.1
CH	SH	0.1
CH	CH	0.9

y_2	Ψ_2
SH	0.5
CH	0.5

y_2	Ψ_{26}
SH	0.1
CH	0.9

y_2	Ψ_{25}
SH	0.1
CH	0.9

Pairwise Markov random field

Definition

A pairwise MRF is given by the pair $< G(V, E), \Psi >$, G is a graph, Ψ is a set of potentials ψ, η, ϕ.
For an assignment y of all Y the MRF is associated with
$P(y|x) = \alpha \prod_{Y_i in Y} \phi_i(y_i) \prod_{(Y_i, Y_j) \in E} \eta_{ij}(y_i, y_j)$

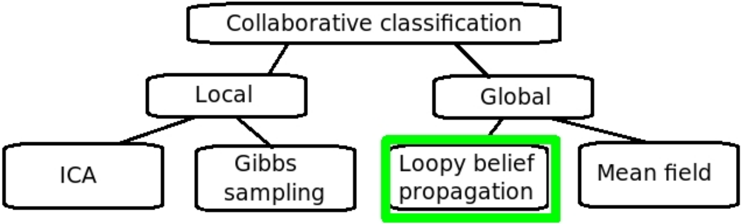
Interpretation

1. The MRF defines a joint p.d.f. of all "unknown" labels
2. Each $P(y|x)$ is the probability of a given world y
3. Same as before obtaining the marginal for $P(Y_i = y_i)$ would require summing over exponential number of terms
4. #P problem $\rightarrow$ approximation

Global CC as a variational method

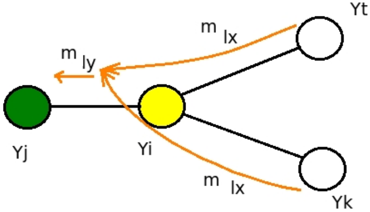
1. Instead of working with the actual distribution defined by the MRF, work with an approximate "trial" distribution
2. The "trial" distribution should be simpler (to compute/store)
3. It should be easier to extract marginals from the "trial" distribution
4. The "trial" should be fitted to the actual distribution

Loopy belief propagation (LBP)



LBP

1. Loopy belief propagation is defined on a pairwise MRF
2. It is a discrete time message passing algorithm
3. At each step a message $m_{i\to j}(y_j)$ is passed from unknown node Y_i to Y_j
4. $m_{i\to j}(y_j) = \alpha \sum_{y_i \in L} \eta_{i,j}(y_i, y_j)\phi_i(y_i) \prod_{Y_k \in N_i \cap Y \setminus Y_j} m_{k\to i}(y_i)$



LBP example

y_1	Ψ_{13}
SH	0.6
CH	0.4

y_1	Ψ_{16}
SH	0.1
CH	0.9

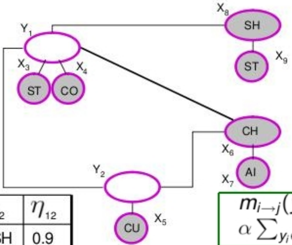
y_1	Ψ_{18}
SH	0.8
CH	0.2

$\Phi_1 = \Psi_1 * \Psi_{13} * \Psi_{14} * \Psi_{16} * \Psi_{18} =$

y_1	Φ_1
SH	0.0096
CH	0.0216

y_1	Ψ_1
SH	0.5
CH	0.5

y_1	Ψ_{14}
SH	0.4
CH	0.6



$\Phi_2 = \Psi_2 * \Psi_{25} * \Psi_{26} =$

y_2	Φ_2
SH	0.005
CH	0.405

Message i->j

$$m_{i\to j}(y_j) = \alpha \sum_{y_i \in L} \eta_{i,j}(y_i, y_j)\phi_i(y_i) \prod_{Y_k \in N_i \cup Y \setminus Y_j} m_{k\to i}(y_i)$$

y_1	y'_2	η_{12}
SH	SH	0.9
SH	CH	0.1
CH	SH	0.1
CH	CH	0.9

y_2	Ψ_2
SH	0.5
CH	0.5

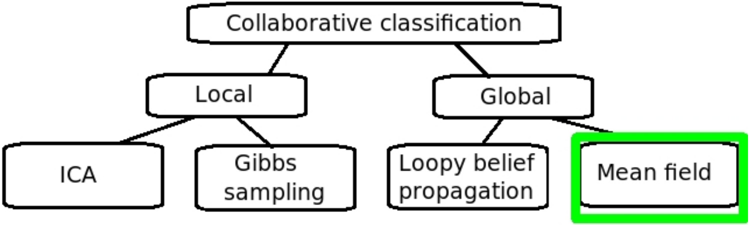
y_2	Ψ_{26}
SH	0.1
CH	0.9

y_2	Ψ_{25}
SH	0.1
CH	0.9

LBP mechanics

1. Initially all messages are set to 1
2. Perform message passing until messages stabilize
3. Compute beliefs
   $b_i(y_i) = \alpha\phi_i(y_i) \prod_{Y_j \in N_i \cap Y} m_{j \to i}(y_i)$
4. $b_i(y_i)$ is the approximation of the marginal probability of y_i for node Y_i

Relaxation labeling via mean-field (MF)



MF

1. MF is defined on MRF
2. MF can be described by the following fixed point equation:
   $b_i(y_i) = \alpha\phi_i(y_i) \prod_{Y_j \in N_j \cap Y} \prod_{y_j \in L} \eta_{ji}^{b_j(y_j)}(y_i, y_j)$
3. Iterative method for computing the fixed point equation

Outline

Experiments

1. Comparison of content-based (CO) and CC classification
2. Comparison of local classifiers for Local CC. Logistic regression (LR) versus Naive Bayes (NB)
3. Comparison of Global and Local CC
4. Eight different classifiers:
    1. CO + NB/LR
    2. ICA + NB/LR
    3. GS + NB/LR
    4. LBP
    5. MF

Experimental setup

1. Real world data
   1. CORA - $|V| = 2708, |E| = 5429, |L| = 7$
   2. Citeseer - $|V| = 3312, |E| = 4732, |L| = 6$
2. Synthetic data $|V| = 1000, |L = 5|$
3. Varying homophily and link density for synthetic data
4. 10-fold cross validation

Choice of features

1. Document terms for both CO and local CC methods
2. Count aggregation of terms
3. MRF with clique and node potentials for Global CC

Sampling for fold validation

1. Create folds for training and evaluation
2. "Snowball sampling" (SS) evaluation
   1. Select a random core node
   2. Expand, choosing a node based on the class distribution
   3. Expand |X|/k times
   4. Create split.
   5. Use the |X|/k sample for testing and the rest for training
3. Random sampling (RS) - Partition |X| in k folds randomly

Sampling challenges

1. SS may result in one and the same node appearing in multiple folds
2. Average the accuracy of each instance and than average over all training
3. Matched (M) average accuracy - only for instances that appear in at least one SS split

Learning the parameters

1. For CO and Local CC - local classifiers parameters
2. For MF and LBP - clique potentials
3. Gradient-based optimization approaches on the labeled nodes in the training splits

Experimental results - real-world datasets

	Cora			Citeseer		
Algorithm	SS	RS	M	SS	RS	M
CO-NB	0.7285	0.7776	0.7476	0.7427	0.7487	0.7646
ICA-NB	**0.8054**	**0.8478**	**0.8271**	0.7540	**0.7683**	**0.7752**
GS-NB	0.7613	0.8404	0.8154	**0.7596**	0.7680	0.7737
CO-LR	0.7356	0.7695	0.7393	0.7334	0.7321	0.7532
ICA-LR	0.8457	0.8796	0.8589	**0.7629**	**0.7732**	0.7812
GS-LR	**0.8495**	**0.8810**	**0.8617**	0.7574	0.7699	**0.7843**
LBP	0.8554	0.8766	0.8575	**0.7663**	**0.7759**	0.7843
MF	**0.8555**	**0.8836**	**0.8631**	0.7657	0.7732	**0.7888**

1. CC dominates CO

Experimental results - real-world datasets

	Cora			Citeseer		
Algorithm	SS	RS	M	SS	RS	M
CO-NB	0.7285	0.7776	0.7476	0.7427	0.7487	0.7646
ICA-NB	**0.8054**	**0.8478**	**0.8271**	0.7540	**0.7683**	**0.7752**
GS-NB	0.7613	0.8404	0.8154	**0.7596**	0.7680	0.7737
CO-LR	0.7356	0.7695	0.7393	0.7334	0.7321	0.7532
ICA-LR	0.8457	0.8796	0.8589	**0.7629**	**0.7732**	0.7812
GS-LR	**0.8495**	**0.8810**	**0.8617**	0.7574	0.7699	**0.7843**
LBP	0.8554	0.8766	0.8575	**0.7663**	**0.7759**	0.7843
MF	**0.8555**	**0.8836**	**0.8631**	0.7657	0.7732	**0.7888**

1. CC dominates CO
2. LR dominates NB

Experimental results - real-world datasets

	Cora			Citeseer		
Algorithm	SS	RS	M	SS	RS	M
CO-NB	0.7285	0.7776	0.7476	0.7427	0.7487	0.7646
ICA-NB	**0.8054**	**0.8478**	**0.8271**	0.7540	**0.7683**	**0.7752**
GS-NB	0.7613	0.8404	0.8154	**0.7596**	0.7680	0.7737
CO-LR	0.7356	0.7695	0.7393	0.7334	0.7321	0.7532
ICA-LR	0.8457	0.8796	0.8589	**0.7629**	**0.7732**	0.7812
GS-LR	**0.8495**	**0.8810**	**0.8617**	0.7574	0.7699	**0.7843**
LBP	0.8554	0.8766	0.8575	**0.7663**	**0.7759**	0.7843
MF	**0.8555**	**0.8836**	**0.8631**	0.7657	0.7732	**0.7888**

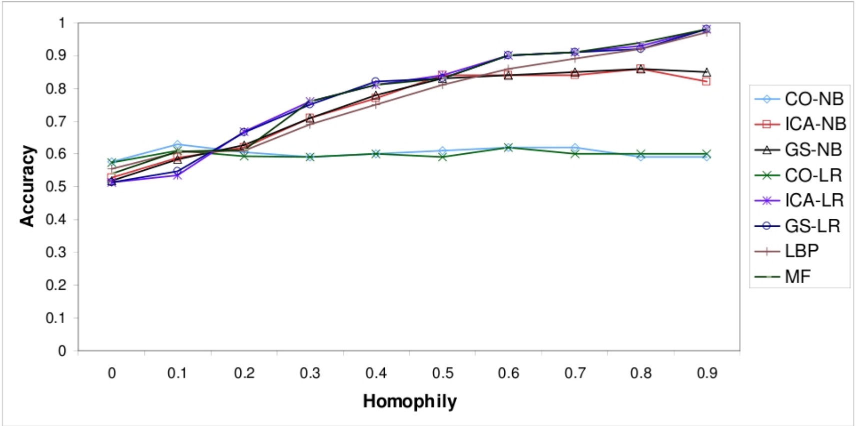
1. CC dominates CO
2. LR dominates NB
3. ICA and GS comparable by accuracy

Experimental results - real-world datasets

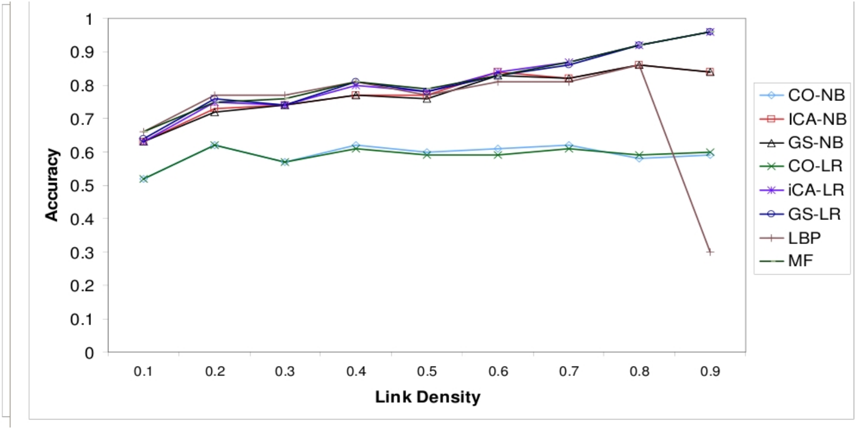
	Cora			Citeseer		
Algorithm	SS	RS	M	SS	RS	M
CO-NB	0.7285	0.7776	0.7476	0.7427	0.7487	0.7646
ICA-NB	**0.8054**	**0.8478**	**0.8271**	0.7540	**0.7683**	**0.7752**
GS-NB	0.7613	0.8404	0.8154	**0.7596**	0.7680	0.7737
CO-LR	0.7356	0.7695	0.7393	0.7334	0.7321	0.7532
ICA-LR	0.8457	0.8796	0.8589	**0.7629**	**0.7732**	0.7812
GS-LR	**0.8495**	**0.8810**	**0.8617**	0.7574	0.7699	**0.7843**
LBP	0.8554	0.8766	0.8575	**0.7663**	**0.7759**	0.7843
MF	**0.8555**	**0.8836**	**0.8631**	0.7657	0.7732	**0.7888**

1. CC dominates CO
2. LR dominates NB
3. ICA and GS comparable by accuracy
4. Slight dominance of Global over Local

Experimental results - synthetic datasets



Experimental results - synthetic datasets



Practical observations

1. MF and LBP are hard to work with. Initialization and convergence issues.
2. ICA is faster than GS (14m vs. 3h on Citeseer with NB)
3. ICA converges in <10 iterations, while GS requires 200 "burn-in" + 800 samples