An Iterative Feature Filter for Sensor Timeseries in Pervasive Computing Applications

Davide Bacciu
Dipartimento di Informatica
Università di Pisa

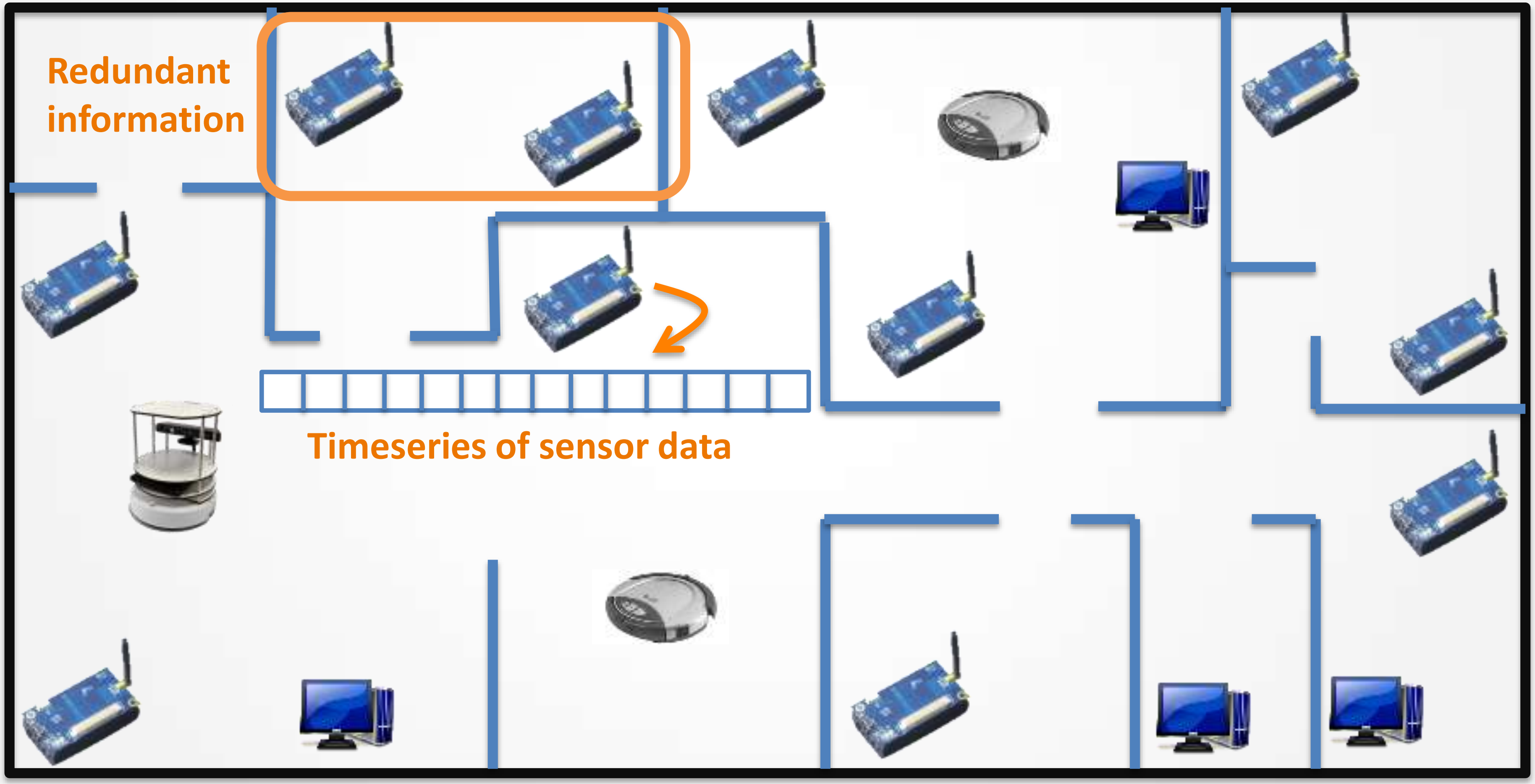
CIML

15th International Conference on Engineering Applications of Neural Networks
5 - 7 September 2014, Sofia, Bulgaria

SEVENTH FRAMEWORK PROGRAMME

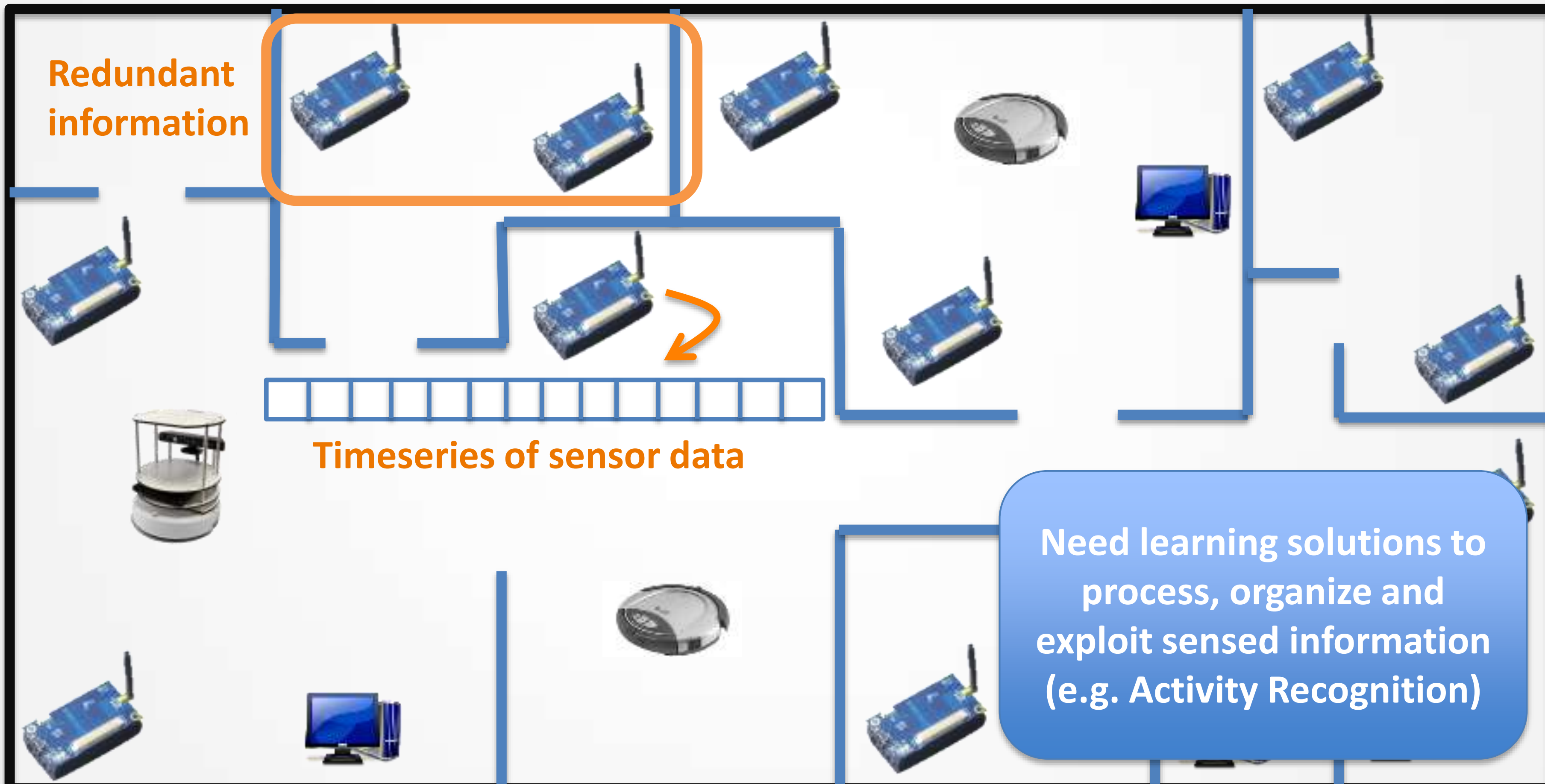
The Application Context

Pervasive computing



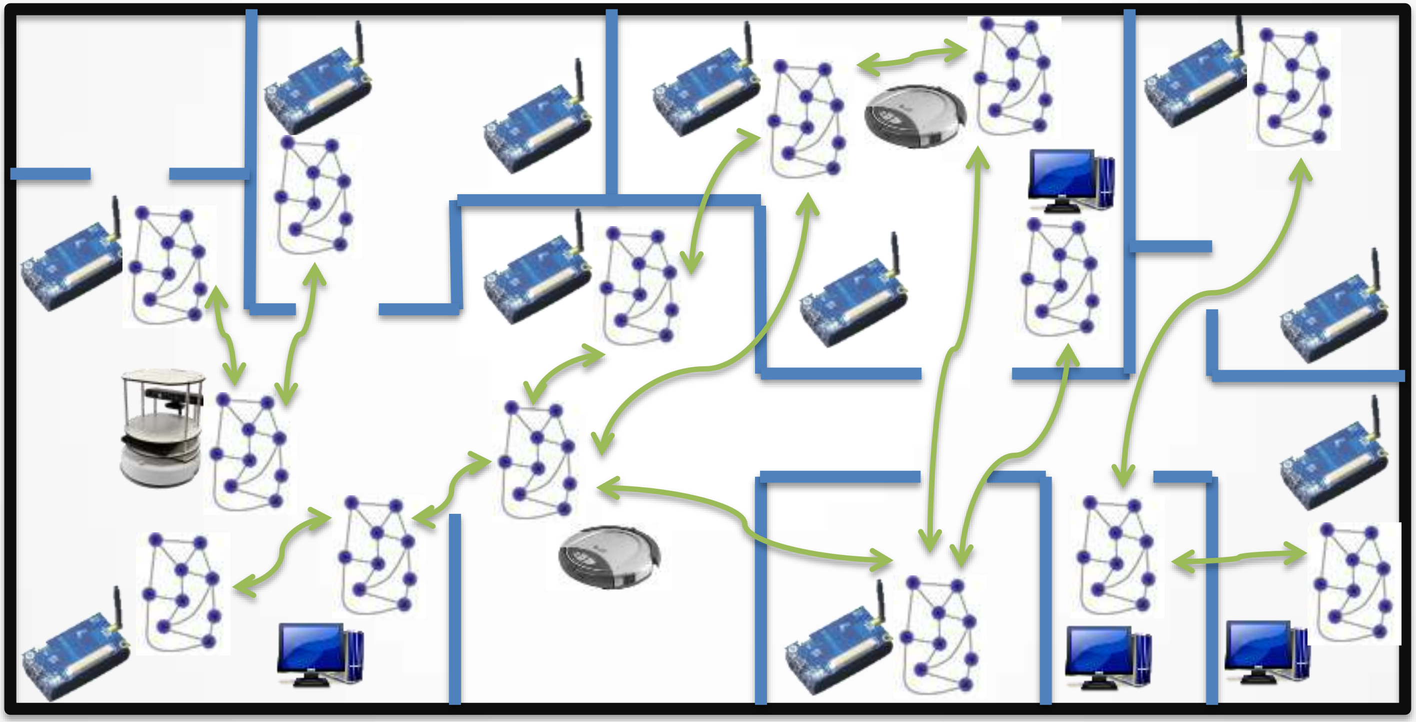
The Application Context

Pervasive computing



The RUBICON Approach

A distributed learning system comprising cooperating Echo State Networks embedded on the network devices



Incremental deployment of new learning tasks during system operation to account for changing environment conditions and new user habits

The Feature Selection Challenge

- Feature selection is required to
  - Reduce computational and communication cost for sensor information processing
  - Suppress redundant/irrelevant information depleting predictive performance
- Application specific requirements
  - Timeseries data
  - Heterogeneous sensor information
  - Automatized and computationally efficient

Lack of significant results in the context of open-ended discovery from real-world sensor-rich environments

Iterative Cross-Correlation Filter (ICF)

- Forward selection-elimination procedure to filter redundant features
- Redundancy is measured by normalized cross-correlation

$$\overline{\phi}^{*}_{x^1x^2} = \max_{\tau}\left|\overline{\phi}_{x^1x^2}(\tau)\right|$$

- Based on the following intuition
  1. Select variables not correlated with any other features
  2. Variables correlated with all currently selected variables are candidates for elimination.
  3. If left with mutually correlated variables, maintain those less correlated with selected features

ICF Algorithm

I. Compute the DxD feature redundancy matrix **R**
   - $R_{ij} = 1$, if features i and j are redundant
   - $R_{ij} = 0$, otherwise

$$\mathbf{R} = \begin{matrix} 0 & 0 & 1 & 1 \\ 0 & 0 & 0 & 0 \\ 1 & 0 & 0 & 0 \\ 1 & 0 & 0 & 0 \end{matrix}$$

II. While unselected/undeleted features exist in R do
   a. Rule 0 – Select completely uncorrelated features
   b. Rule 1 – Delete completely correlated features
   c. Rule 2 – If **R** contains only ones select less correlated feature
   d. Rule 3 – If stuck, pick one feature for selection and one for deletion

ICF Algorithm

I. Compute the DxD feature redundancy matrix **R**
   - $R_{ij} = 1$, if features i and j are redundant
   - $R_{ij} = 0$, otherwise

$$\mathbf{R} = \begin{matrix} 0 & 0 & 1 & 1 \\ 0 & 0 & 0 & 0 \\ 1 & 0 & 0 & 0 \\ 1 & 0 & 0 & 0 \end{matrix}$$

II. While unselected/undeleted features exist in R do
   a. Rule 0 – Select completely uncorrelated features
   b. Rule 1 – Delete completely co
   c. Rule 2 – If **R** contains only or
      feature
   d. Rule 3 – If stuck, pick one fea
      one for deletion

$$O(\underbrace{N}_{\text{samples}} \cdot (\underbrace{D}_{\text{features}}^2 \cdot T^{\underbrace{max}_{\text{length}}}) + D)$$

ICF Algorithm – Rule 0

Before entering the while loop let

- $F = \{1, \ldots, D\}$, the set of unassigned features
- $SF = \{\}$, the set of selected features
- $DF = \{\}$, the set of deleted features

If any row R_i is completely uncorrelated with the others in **R**

$$\mathbf{R} = \begin{array}{cccc} 1 & 2 & 3 & 4 \\ 0 & \boxed{0} & 1 & 1 \\ 0 & \boxed{0} & 0 & 0 \\ 1 & \boxed{0} & 0 & 0 \\ 1 & \boxed{0} & 0 & 0 \end{array}$$

ICF Algorithm – Rule 0

Before entering the while loop let

- F = {1, …, D}, the set of unassigned features
- SF = {}, the set of selected features
- DF = {}, the set of deleted features

If any row R_i is completely uncorrelated with the others in **R**

I. Set $SF = SF \cup \{i\}$ and $F = F \,/\, \{i\}$

$$\mathbf{R} = \begin{array}{cccc} 1 & 2 & 3 & 4 \\ 0 & \boxed{0} & 1 & 1 \\ 0 & 0 & 0 & 0 \\ 1 & 0 & 0 & 0 \\ 1 & 0 & 0 & 0 \end{array}$$

$F = \{1,3,4\}$, $SF = \{2\}$, $DF = \{\}$

ICF Algorithm – Rule 0

Before entering the while loop let

- $F = \{1, \ldots, D\}$, the set of unassigned features
- $SF = \{\}$, the set of selected features
- $DF = \{\}$, the set of deleted features

If any row R_i is completely uncorrelated with the others in **R**

I. Set $SF = SF \cup \{i\}$ and $F = F / \{i\}$

II. Remove i-th row/col from **R**

$$\mathbf{R} = \begin{array}{ccc} 1 & 3 & 4 \\ 0 & 1 & 1 \\ 1 & 0 & 0 \\ 1 & 0 & 0 \end{array}$$

$F = \{1,3,4\}$, $SF = \{2\}$, $DF = \{\}$

ICF Algorithm – Rule 1

If any row R_i is completely correlated with others in **R** and there is at least 1 non completely correlated feature

I. Set $DF = DF \cup \{i\}$ and $F = F / \{i\}$

$$\mathbf{R} = \begin{array}{ccc} 1 & 3 & 4 \\ \boxed{\begin{matrix} 0 \\ 1 \\ 1 \end{matrix}} & \begin{matrix} 1 \\ 0 \\ 0 \end{matrix} & \begin{matrix} 1 \\ 0 \\ 0 \end{matrix} \end{array}$$

$F = \{3,4\}$, $SF = \{2\}$, $DF = \{1\}$

ICF Algorithm – Rule 1

If any row R_i is completely correlated with others in **R** and there is at least 1 non completely correlated feature

I. Set $DF = DF \cup \{i\}$ and $F = F \,/\, \{i\}$

II. Remove i-th row/col from **R**

$$\mathbf{R} = \begin{array}{ccc} 1 & 3 & 4 \\ \boxed{\begin{array}{c} 0 \\ 1 \\ 1 \end{array}} & \begin{array}{c} 1 \\ 0 \\ 0 \end{array} & \begin{array}{c} 1 \\ 0 \\ 0 \end{array} \end{array} \Rightarrow \mathbf{R} = \begin{array}{cc} 3 & 4 \\ 0 & 0 \\ 0 & 0 \end{array}$$

F = {3,4} , *SF* = {2}, *DF* = {1}

ICF Algorithm – Rule 1

If any row R_i is completely correlated with others in **R** and there is at least 1 non completely correlated feature

I. Set *DF* = *DF* ∪ {i} and *F* = *F* / {i}

II. Remove i-th row/col from **R**

$$\mathbf{R} = \begin{array}{ccc} 1 & 3 & 4 \\ 0 & 1 & 1 \\ 1 & 0 & 0 \\ 1 & 0 & 0 \end{array} \Rightarrow \mathbf{R} = \begin{array}{cc} 3 & 4 \\ 0 & 0 \\ 0 & 0 \end{array} \Rightarrow \begin{array}{l} F = \{\} \\ SF = \{2,3,4\}, \\ DF = \{1\} \end{array}$$

RULE 0 will fire again at next iteration

F = {3,4} , *SF* = {2}, *DF* = {1}

ICF Algorithm – Rule 2

If **R** contains only ones off-diagonal

I. Pick feature i in *F* that is less correlated with those already in *SF*

$$\mu_{ij} = \frac{\sum_{n=1}^{N} \overline{\phi}^{*}_{x_i^n x_j^n}}{N}$$

$$\mathbf{R} = \begin{array}{cc} 3 & 4 \\ 0 & 1 \\ 1 & 0 \end{array}$$

F = {3,4}

SF = {2}, DF = {1}

ICF Algorithm – Rule 2

If **R** contains only ones off-diagonal

I. Pick feature i in *F* that is less correlated with those already in *SF*

$$\mu_{ij} = \frac{\sum_{n=1}^{N} \overline{\phi}^{*}_{x_i^n x_j^n}}{N}$$

$$\mathbf{R} = \begin{array}{cc} 3 & 4 \\ \boxed{\begin{matrix} 0 \\ 1 \end{matrix}} & \boxed{\begin{matrix} 1 \\ 0 \end{matrix}} \end{array} \qquad \boxed{\mu_{32}} < \boxed{\mu_{42}}$$

F = {3,4}

SF = {2}, *DF* = {1}

F = {}

SF = {2,3},

DF = {1,4}

ICF Algorithm – Rule 2

If **R** contains only ones off-diagonal

I. Pick feature i in *F* that is less correlated with those already in *SF*

II. Set *SF* = *SF* ∪ {i} and *F* = *F* / {i}

III. Set *DF* = *DF* ∪ *F* and *F* = {}

IV. Terminate while loop

$$\mu_{ij} = \frac{\sum_{n=1}^{N} \overline{\Phi}^{*}_{x_i^n x_j^n}}{N}$$

$$\mathbf{R} = \begin{array}{cc} 3 & 4 \\ \hline 0 & 1 \\ 1 & 0 \end{array} \Rightarrow \boxed{\mu_{32}} < \boxed{\mu_{42}} \Rightarrow$$

F = {}
SF = {2,3},
DF = {1,4}

F = {3,4}
SF = {2}, *DF* = {1}

ICF Algorithm – Rule 3

If neither RULE 1 nor RULE 2 fire,

I. Pick feature i in *F* that is minimally correlated with features in *SF*

$$\mathbf{R} = \begin{array}{cccc} 1 & 3 & 4 & 5 \\ \begin{matrix} 0 \\ 1 \\ 0 \\ 1 \end{matrix} & \begin{matrix} 1 \\ 0 \\ 0 \\ 0 \end{matrix} & \boxed{\begin{matrix} 0 \\ 0 \\ 0 \\ 1 \end{matrix}} & \begin{matrix} 1 \\ 0 \\ 1 \\ 0 \end{matrix} \end{array}$$

F = {1,3,4,5}
SF = {2}, *DF* = {}

$$\boxed{\mu_{42}} < \mu_{32}$$

ICF Algorithm – Rule 3

If neither RULE 1 nor RULE 2 fire,

I. Pick feature i in *F* that is minimally correlated with features in *SF*

II. Pick feature j from those correlated to i that is also the maximally correlated with selected features in *SF*

$$\mathbf{R} = \begin{array}{cccc} 1 & 3 & 4 & 5 \\ 0 & 1 & 0 & 1 \\ 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 \\ 1 & 0 & 1 & 0 \end{array}$$

F = {1,3,4,5}

SF = {2}, *DF* = {}

$$\boxed{\mu_{42}} < \mu_{32}$$

ICF Algorithm – Rule 3

If neither RULE 1 nor RULE 2 fire,

I. Pick feature i in *F* that is minimally correlated with features in *SF*

II. Pick feature j from those correlated to i that is also the maximally correlated with selected features in *SF*

III. Add i to *SF* and j to *DF*

IV. Set *F* = F / {i,j}

$$R = \begin{array}{cccc} 1 & 3 & 4 & 5 \\ 0 & 1 & 0 & 1 \\ 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 \\ 1 & 0 & \boxed{1} & 0 \end{array}$$

F = {1,3,4,5}

SF = {2}, *DF* = {}

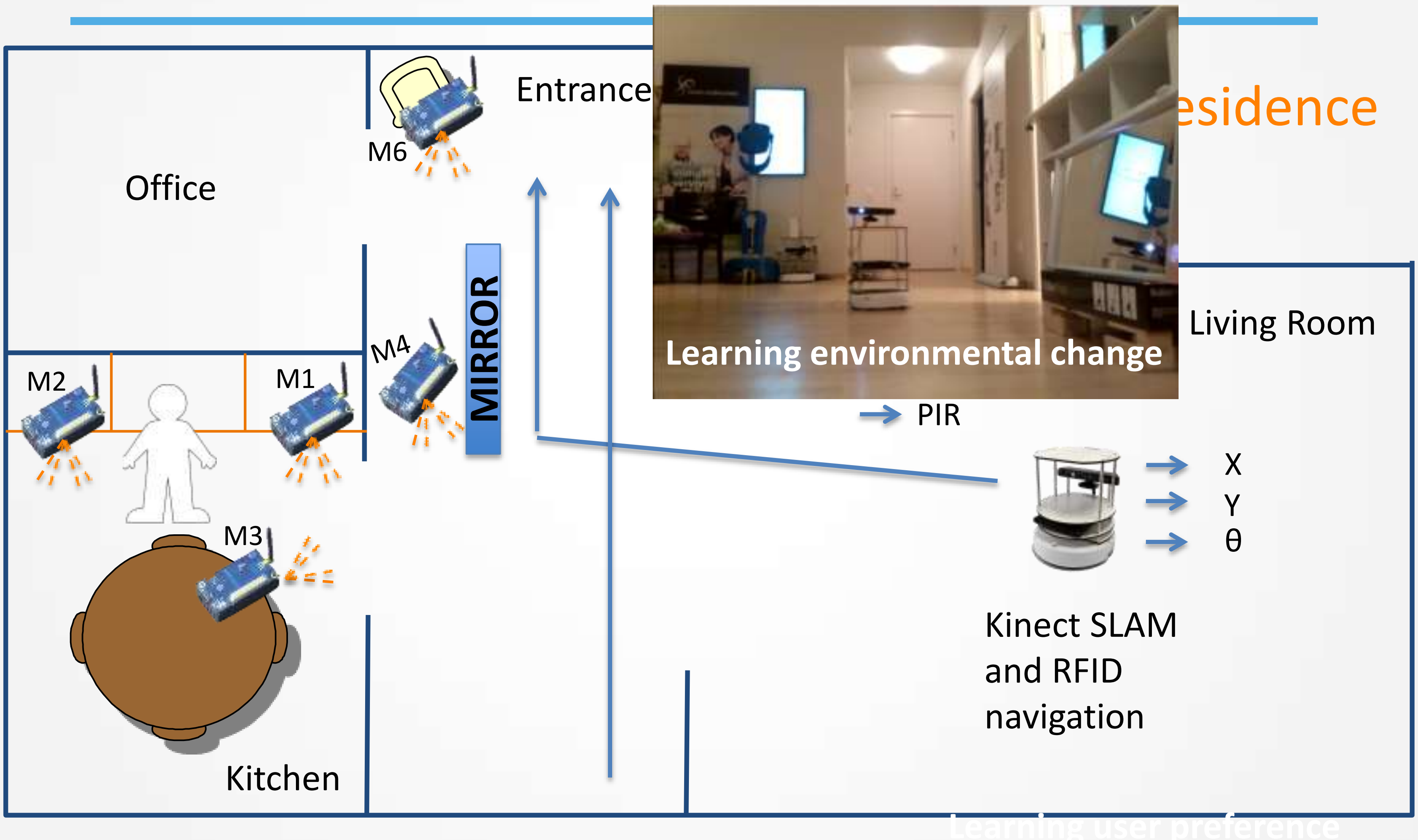
F = {1,3}

SF = {2,4}

DF = {5}

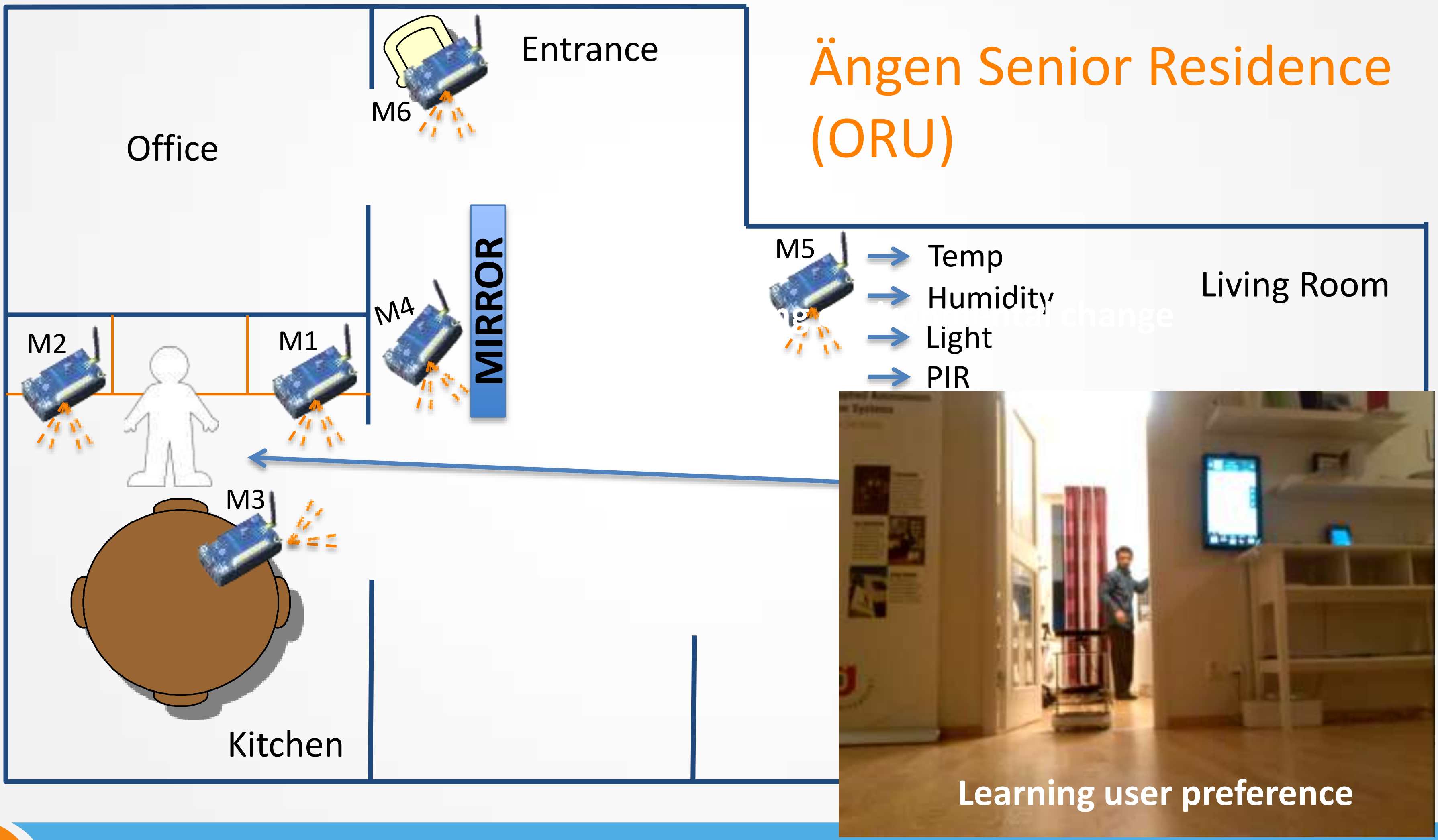
$$\boxed{\mu_{42}} < \mu_{32}$$

Experimental Scenario



Experimental Scenario

Ängen Senior Residence (ORU)



Datasets & Experimental Setup

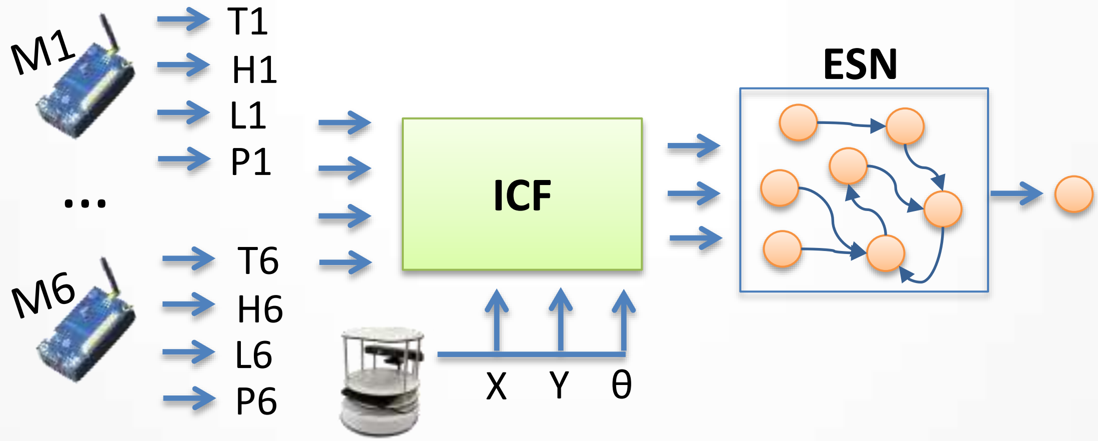
Entrance Dataset – 87 sequences, 2Hz sampling, 127 average timeseries length

Kitchen Dataset – 104 sequences, 2Hz sampling, 197 average timeseries length

Input : 24 sensors + 3 robot localization features

Target: preference weights in [0,1]

Test feature selection for different input features



Evaluate using oracular knowledge on which features are relevant for each task

Experimental comparison with state-of-the-art CleVer algorithm (needs human supervision!)

Entrance Task – Selected Features

Configuration	ICF	CleVer-IT	CleVer -OPT
M3 + (X,Y,θ)	X, Y, P3	L3, P3, Y	L3, P3, T3, Y
M3+M6+(X,Y,θ)	X, Y, P3	L3, P6, X,θ	L3, P3, P6, X,θ
M4-M6 + (X,Y,θ)	Y, P4, P6, L4	P4, L6, T6, θ	P4, L6, T6, θ
M3-M6 + (X,Y,θ)	X, Y, P3, P6	L3, P3, T5, θ	L3, T3, P4, P6, θ
M1-M6 + (X,Y,θ)	X, Y, P1, P2, P3, P6	L1, P2, T2, L3, L6, θ	P2, L3, P4, L5, θ

L = Light
P = PIR
T = Temperature

ICF can identify relevant features even when many uninformative features are present

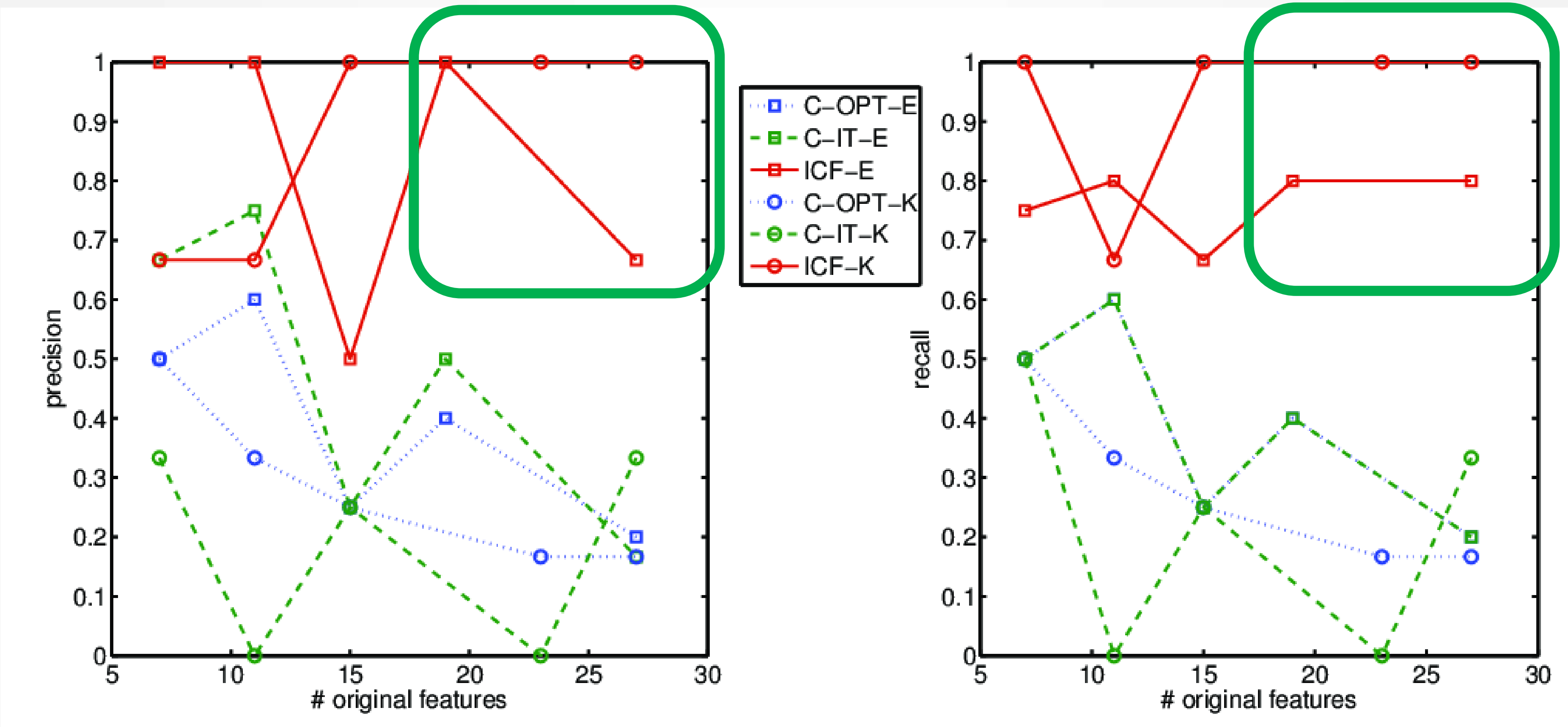
Kitchen Task – Selected Features

Configuration	ICF	CleVer-IT	CleVer -OPT
M3 + (X,Y,θ)	L3, P3, X	L3, X, Y	X, Y
M1+M3+(X,Y,θ)	L1, P3, X	L3, T3, Y	L3, X, Y
M1-M3 + (X,Y,θ)	P1, P2, P3, X	L1, L2, H2, X	L1, L2, H2, X, Y
M1-M5 + (X,Y,θ)	P1, P2, P3, P4, P5, X	L1, L2, H2, L3, T3, L4	L1, P2, L3, T3, L4, H5
M1-M6 + (X,Y,θ)	P1, P2, P3, P4, P5, X	T1, L2, P2, H2, L3, P4	T1, L2, H2, L3, P4, P6

L = Light
P = PIR
T = Temperature
H = Humidity

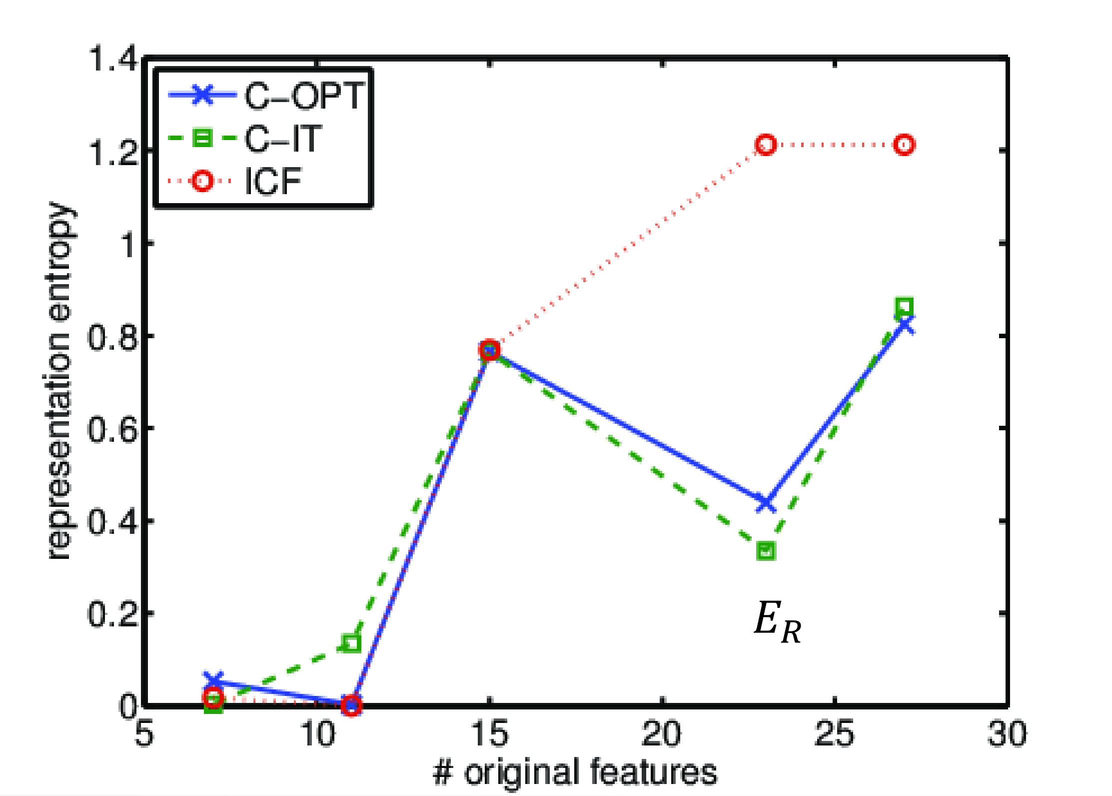
Selected features can drastically change in different CleVer executions

Precision-Recall Analysis



ICF proportion of FP and FN does not grow with the size of the search space

Kitchen Task – Representation Entropy



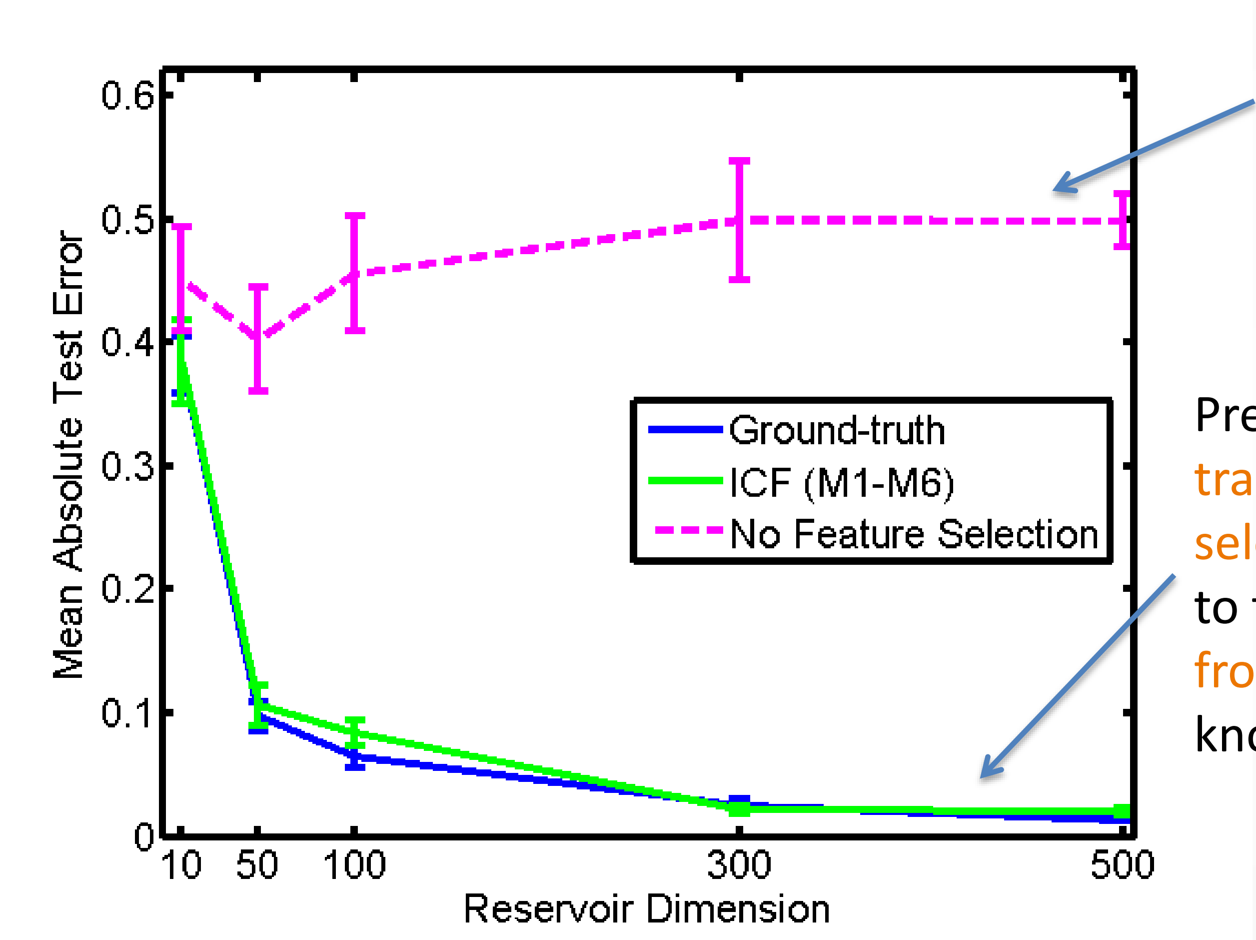
Representation entropy measures the amount of redundancy present in the selected feature subsets

$$E_R = -\sum_{i=1}^{K} \bar{\lambda}_i \log \bar{\lambda}_i$$

$\bar{\lambda}_i$ normalized eigenvalue associated to the i-th selected feature

E_R is minimal when all information is concentrated on a single feature

Kitchen Task – Predictive Performance



Redundant information hampers predictive performance

Predictive accuracy of ESNs trained on the feature selected by ICF is equivalent to that of ESNs using inputs from ground-truth knowledge

Conclusion & Future Works

- An efficient feature filter algorithm tailored to real-time pervasive computing applications
  - Noisy, often slowly changing, heterogeneous sensor timeseries
  - Provide unsupervised identification of non-redundant timeseries
  - Yield stable initialization-independent feature subsets
- Coming soon..
  - Two-phase feature selection mechanism
  - ICF filter to perform redundancy reduction
  - ESNnigma supervised wrapper to optimize ESN predictive performance

ICF and CleVer Matlab code soon available
www.di.unipi.it/~bacciu/icf

FP7 RUBICON (grant n. 269914, 2011-2014)