Data Mining and Neural Networks in Stata

2nd Italian Stata Users Group Meeting

Milano, 10 October 2005

Mario Lucchini e Maurizo Pisati
Università di Milano-Bicocca
mario.lucchini@unimib.it
maurizio.pisati@unimib.it

Data mining: definition

- DM is the process of exploration and modeling of large quantities of data in order to discover useful rules and relations in empirical data
- DM is the analysis of large observational data sets (huge databases, data warehouse, distributed information systems) to find unsuspected relationships and to summarize the data in ways that are useful and understandable for the data owner, the decision maker
- So DM can generate novel unsuspected interpretations of the data (serendipity)

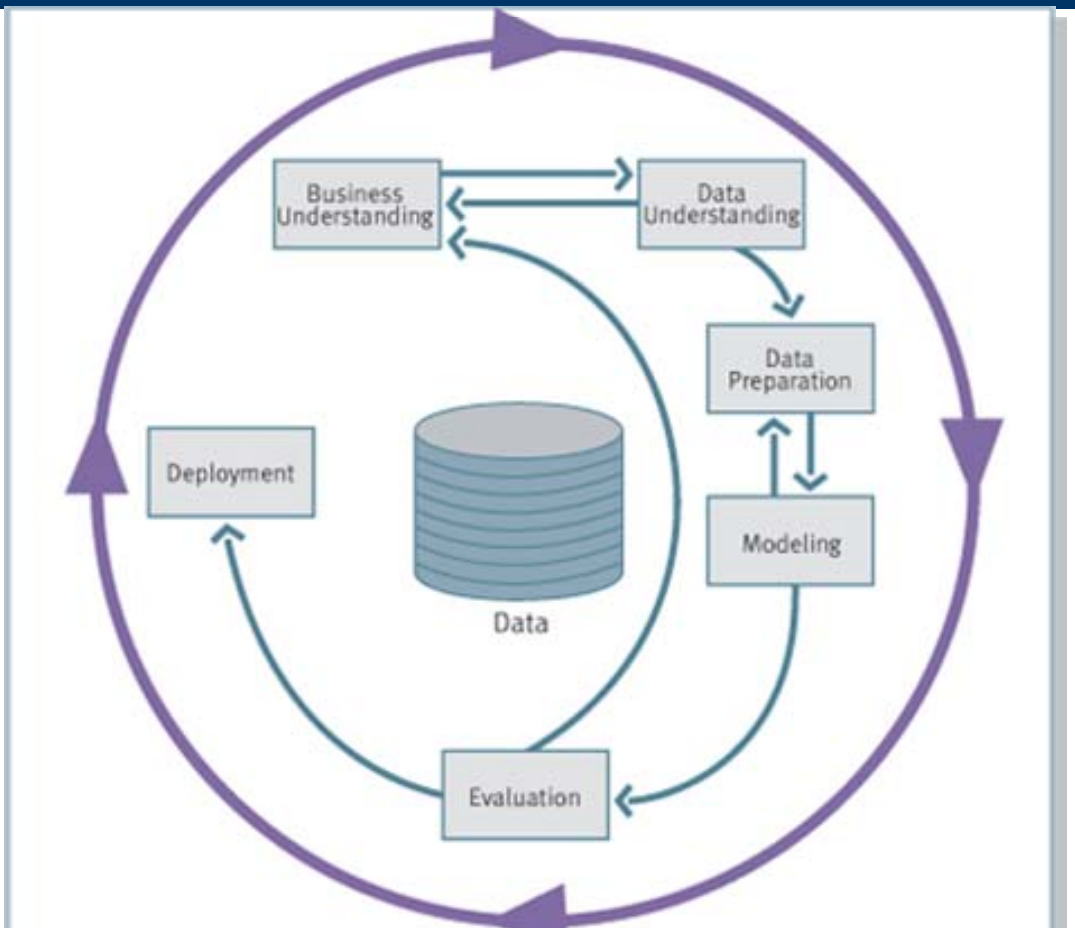
Data mining: applications

- Business can learn from their transaction data about the behavior of their customers and therefore can improve their business by exploiting this knowledge
- Scientists can obtain from observational data new insights on research questions

Data mining vs. statistics

- Statistics:
  - is more theory-based
  - is more model-driven and focused on testing hypotheses
  - top-down approach (transparent models)
- Data mining
  - is more heuristic and data-driven
  - focused on the process of knowledge discovery, including data cleaning, learning, and integration and visualization of results
  - bottom-up approach (black-box models)
- Distinctions are fuzzy

From raw data to knowledge



Data integration
-data cleaning
-data warehouse

Data selection
-datamart

Model specification

Model estimation

Model evaluation

Knowledge

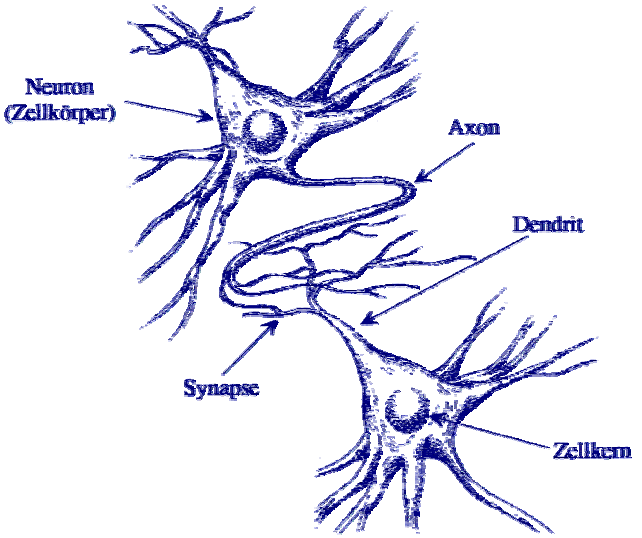
Data mining tasks

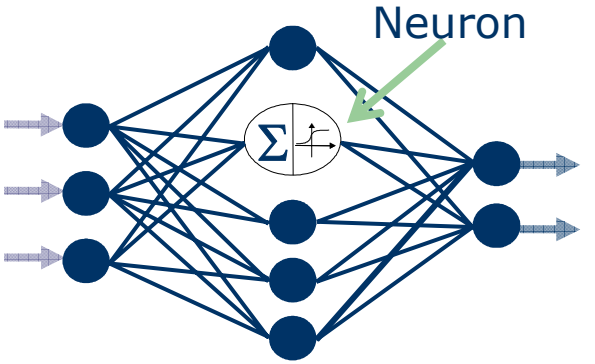
- Description (statistical summaries)
- Supervised classification and prediction (e.g., GLM, ANNs)
- Unsupervised classification and prediction (e.g., K-means clustering, SOMs, decision trees)
- Association rule mining (e.g., multidimensional scaling, factor analysis, basket analysis, correspondence analysis, conjoint analysis)
- Optimization (genetic algorithms)
- Visualization

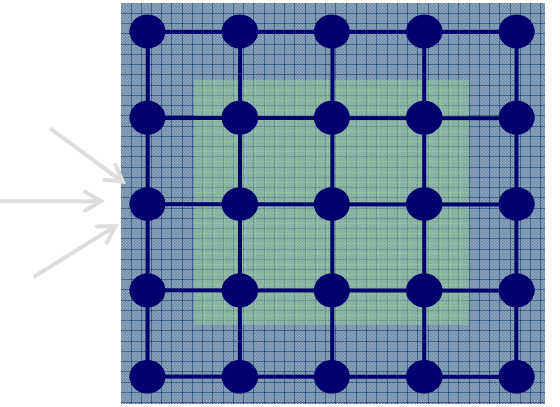
Artificial Neural Networks(ANNs): theoretical background

- ANNs represent new statistical models based on the anatomy and physiology of the nervous system
- ANNs can be considered an extreme simplification of biological nervous systems
- ANNs attempt to mimic the fault-tolerance and plasticity of learning of biological neural systems by imitating the structure of the brain
- ANN technology is being used to solve a wide variety of complex problems

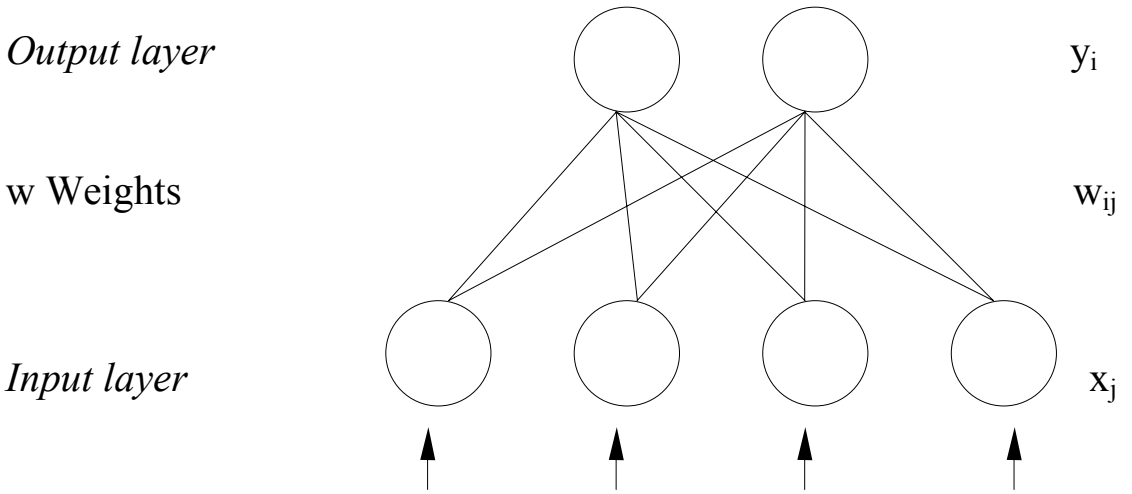
Artificial Neural Networks(ANNs): theoretical background



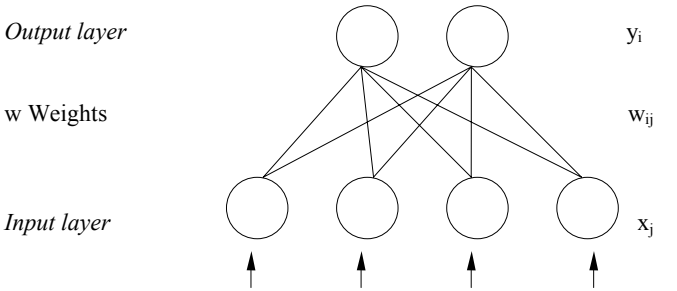




Formal characteristics of ANNs: architecture and functioning



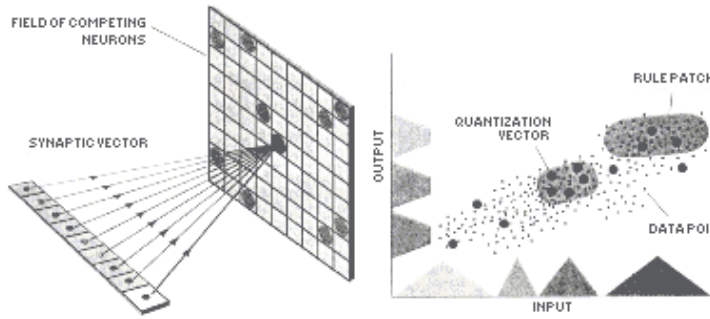
ANNs architecture



- The fundamental constituent elements of the neural networks are *units*, also called *processing elements, neurons* or *nodes*
- The units within a neural system are connected one to the other according to a scheme which can be defined as a *pattern of connectivity*
- The pattern of connectivity is shown in the W matrix of the synaptic weights
- The synapses of a neural network are the parameters which will be estimated in the course of the training process

ANNs architecture

- The behavior of an ANN depends on the way the processing elements (neurons) are connected (architecture) and on the strength of the weights
- The weights are adjusted according to a specified learning rule



ANNs architecture

- ANNs are adaptive systems able to learn prototype-based rules (fuzzy logic rules) from empirical data
- DIRO (Data In Rules Out) principle: ANNs are powerful tools for recognizing patterns, classifying data and making predictions
- ANNs don’t require explicit model or limiting assumptions of normality or linearity

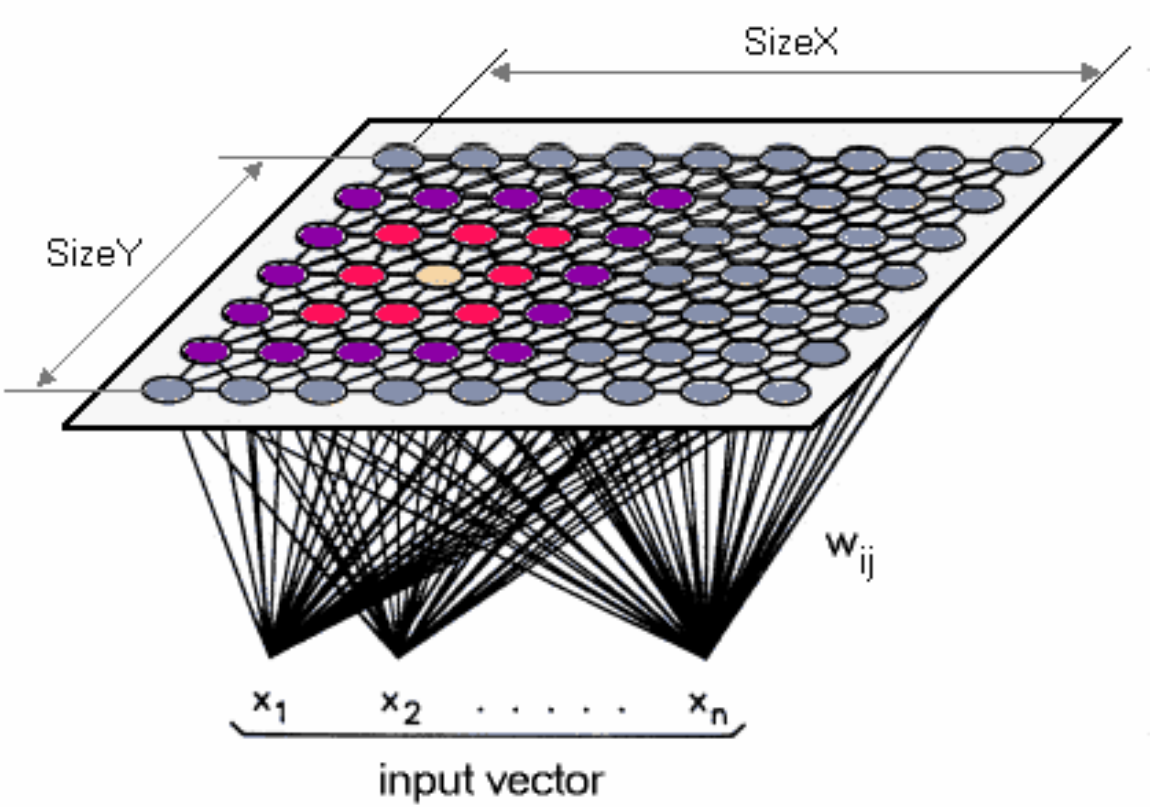
Supervised & unsupervised ANNs

- **Supervised** ANNs (feed-forward networks) are very similar to regression models. They have one-way connections from input to output layers, i.e., from independent to dependent variables. They are used for prediction, pattern recognition, and nonlinear function fitting
- **Unsupervised** ANNs are trained to find relationships and to detect regularities in the input data without any prior classification scheme

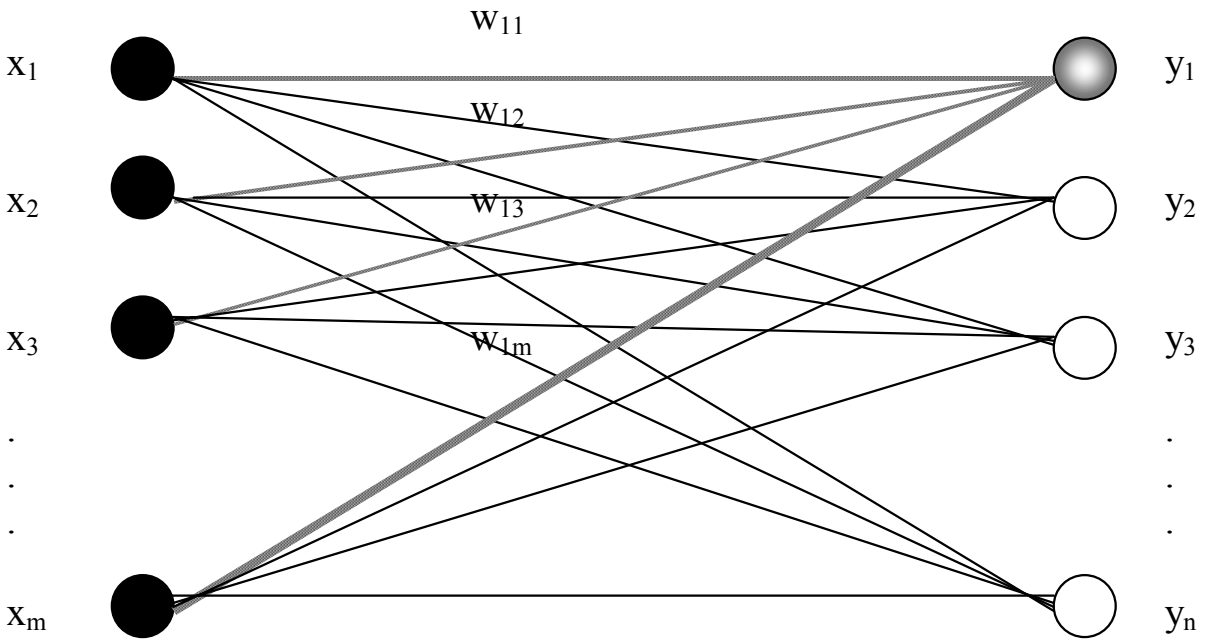
Self-Organizing Map

- The neurons of a SOM learn to recognize groups of similar input vectors in a topological way so that neurons physically close recognize similar input vector (Kohonen 1987)
- SOM learn to classify input vectors according their distances in the multidimensional input space
- At the end of the training a SOM is able to learn the topology of its input space

SOM architecture



SOM architecture



$$i^* = \max\left(\sum_j w_{ij} x_i\right)$$

$$i^* = \min \left\| X - W_i \right\|$$

The winner has output =1

SOM algorithm

1. Similarity matching

$$i^* = \max\left(\sum_j w_{ij} x_i\right) \quad \text{or} \quad i^* = \min \|X - W_i\|$$

2. Updating

$$\Delta W_{ij} = \eta \Lambda(i, i^*)(x_j - w_{ij})$$

where $\Lambda(i, i^*) = \exp\left(-\frac{r^2}{2\sigma^2}\right)$

`ksom`

- As a first step toward a full-blown Stata Neural Network Toolbox, in this talk we present the prototype of a new package called **`ksom`**
- This package is intended to implement Kohonen's Self-Organizing Maps

`ksom`

- **`ksom`** is intended to be articulated into five subcommands:
  - **`ksom preprocess`**: prepares data for the analysis
  - **`ksom initialize`**: defines SOM architecture
  - **`ksom training`**: trains SOM using proper input data
  - **`ksom visualize`**: displays results for SOM interpretation
  - **`ksom project`**: projects passive variables onto SOM

ksom

- **ksom** is written in standard Stata language + Mata language
- Working on **ksom** helped us appreciate Mata as a very powerful and fast language for implementing quantitative data analysis techniques
- We also (re)discovered Stata graphical capabilities: they're really outstanding!

ksom

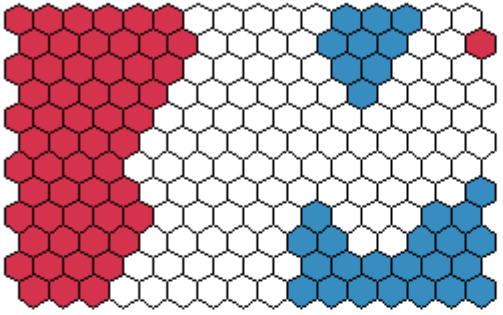
- **ksom** is still at development stage
- Currently, **ksom** is not publicly available

Example

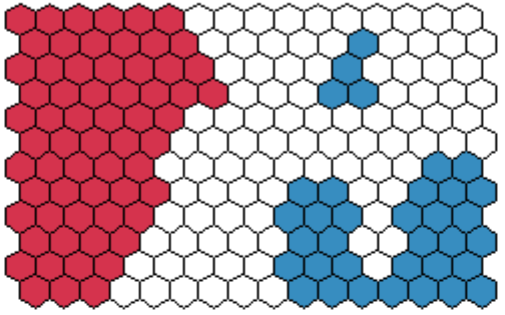
- Mapping of social exclusion in 12 EC countries
- Cases: 3.500 obs per country
- Variables (components): 6 pseudo-metric and 15 binary indicators of social exclusion
- SOM architecture: 16x12 rectangular map, hexagonal lattice
- Training epochs: 50

Component deviation from sample mean

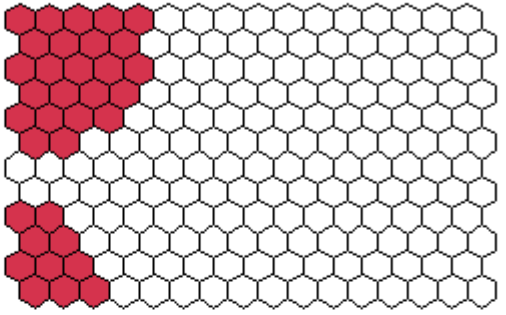
Satisfaction with finances



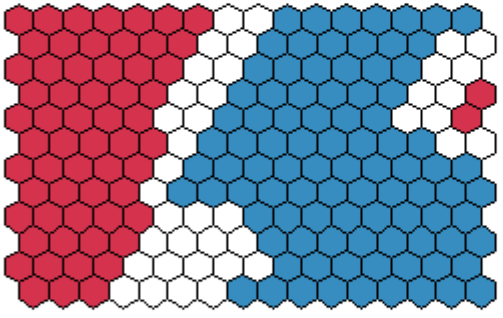
Difficulty to make ends meet



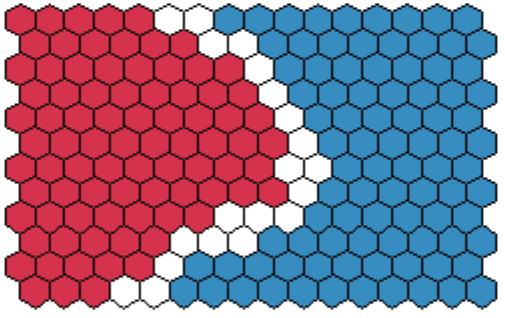
Can't keep home adequately warm



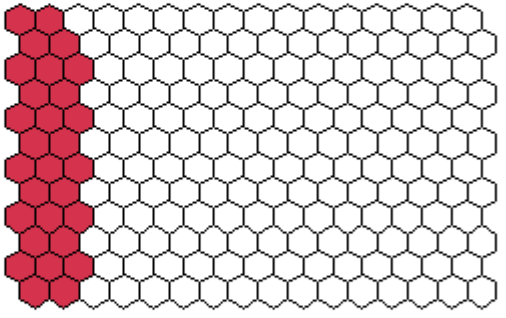
Can't afford 1 week annual holiday



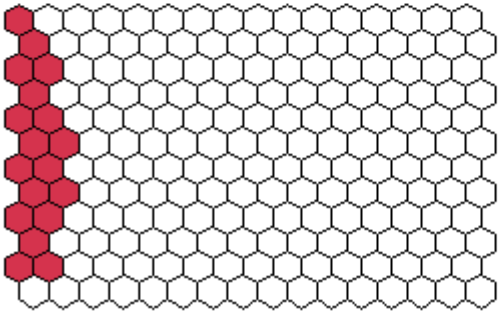
Can't afford replace worn-out furniture



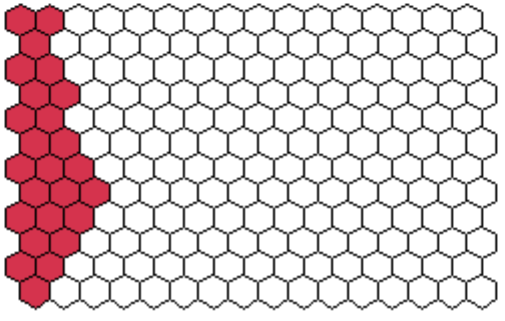
Can't afford new clothes



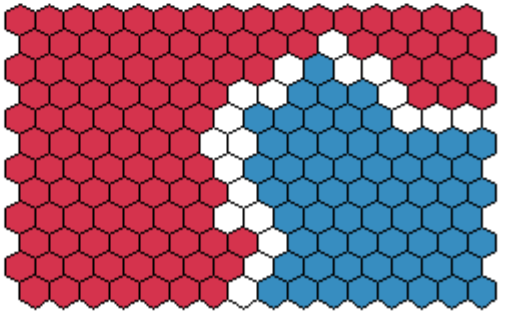
Can't afford eat meat every 2nd day



Can't afford have guests over for dinner once a month

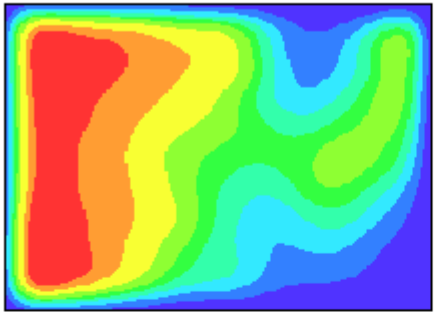


No money left to save

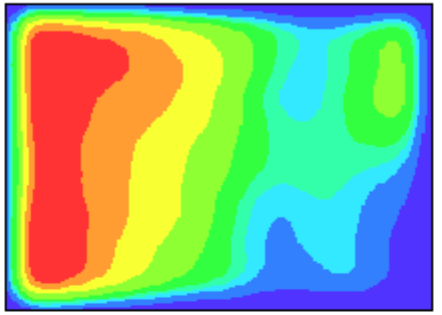


Component planes

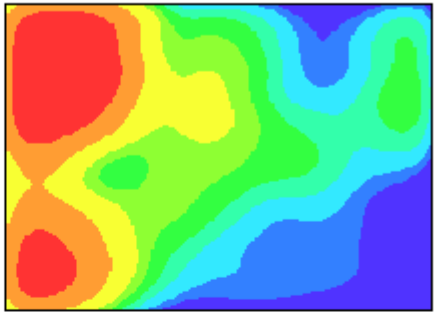
Satisfaction with finances



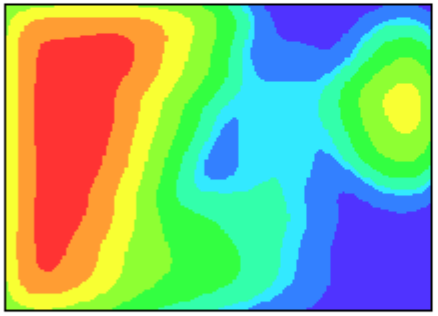
Difficulty to make ends meet



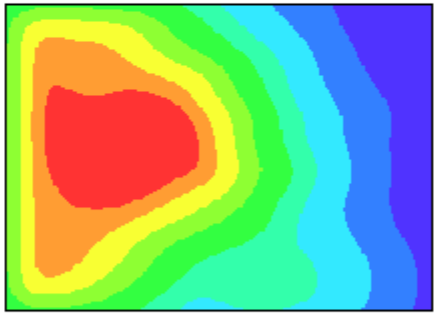
Can't keep home adequately warm



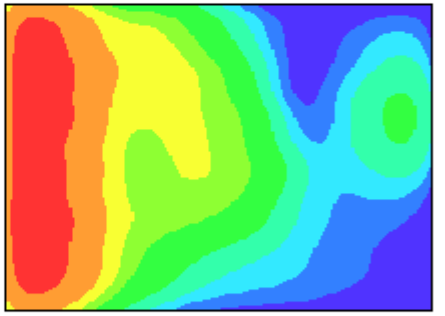
Can't afford 1 week annual holiday



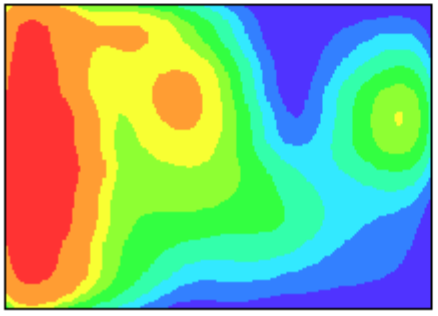
Can't afford replace worn-out furniture



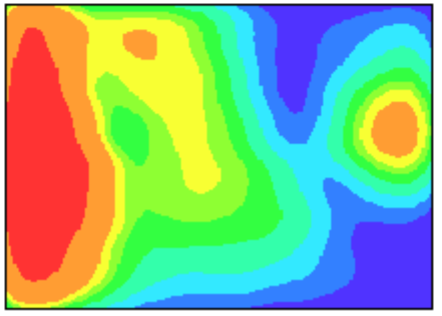
Can't afford new clothes



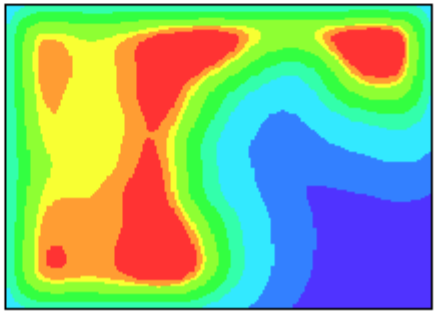
Can't afford eat meat every 2nd day



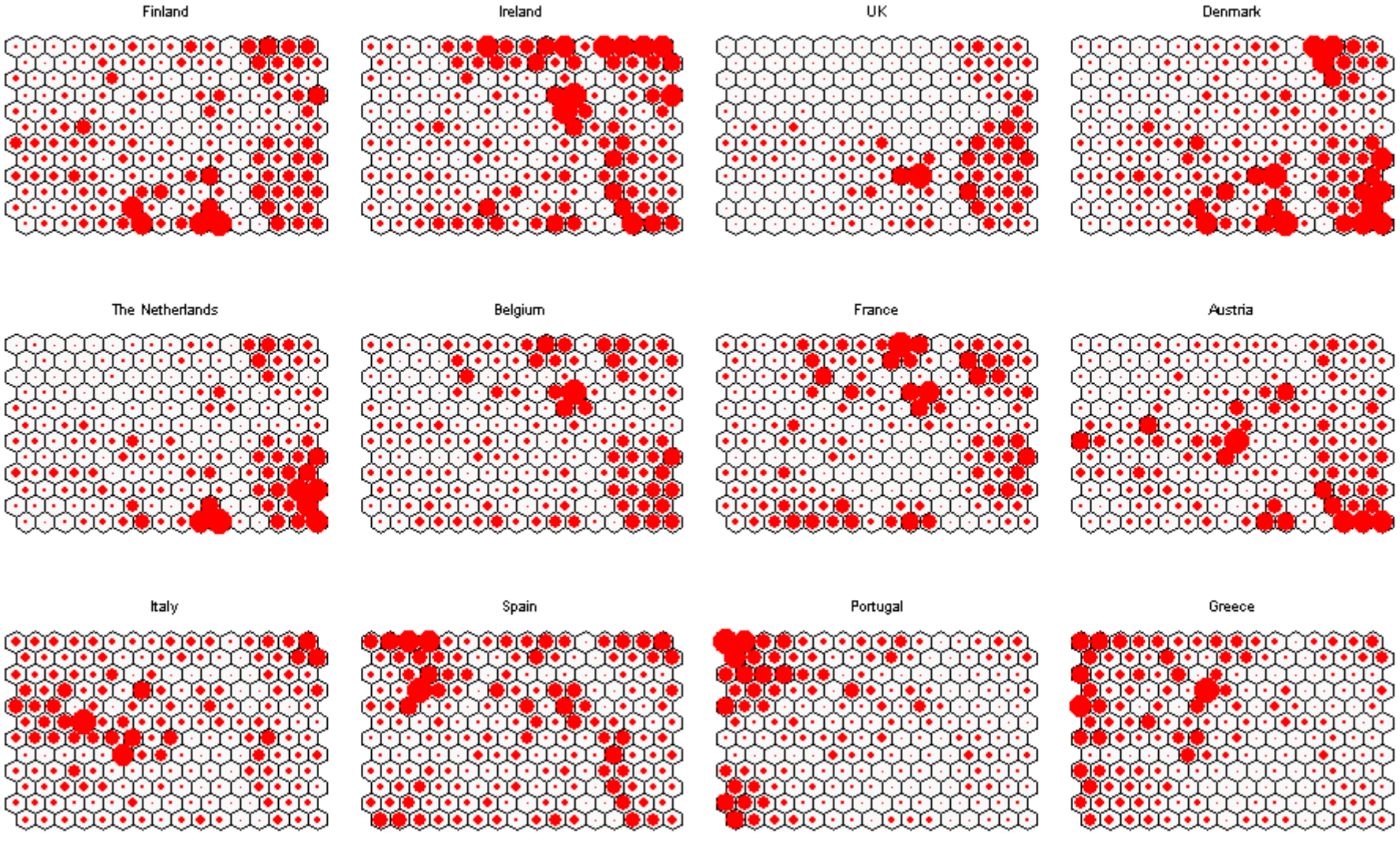
Can't afford have guests over for dinner once a month



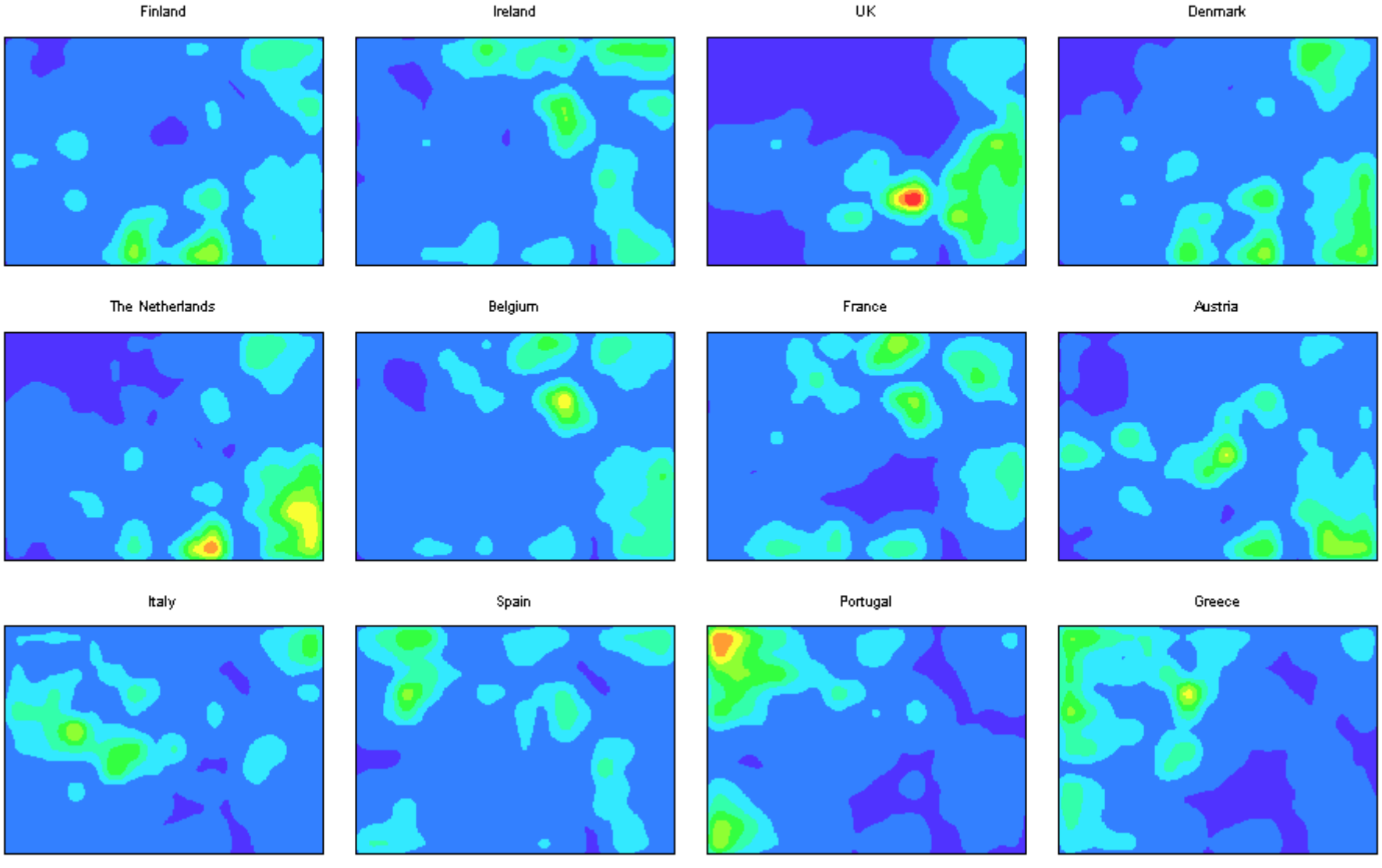
No money left to save



Pr(u | Country)


Finland
Ireland
UK
Denmark
The Netherlands
Belgium
France
Austria
Italy
Spain
Portugal
Greece

Pr(u | Country)


Finland
Ireland
UK
Denmark
The Netherlands
Belgium
France
Austria
Italy
Spain
Portugal
Greece