Data Mining with Multilayer Perceptrons (and other models): Intensive Care and Meat Quality applications

Paulo Cortez

pcortez@dsi.uminho.pt

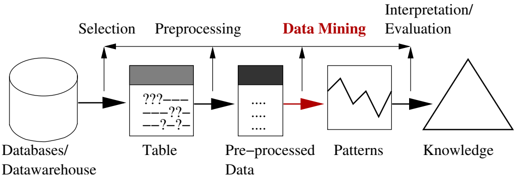
http://www.dsi.uminho.pt/~pcortez

Department of Information Systems
University of Minho - Campus de Azurém
4800-058 Guimarães - Portugal

Motivation

- With the advances in in Information and Communications Technologies, it is easy to collect, store, process and share data;
- There has been an **ever-increasing load of data** in organizations: massive datasets are commonplace and **stored data tends to double every 9 months**;
- All this data, often with high complexity, holds **valuable information**;
- Human experts are limited and may overlook relevant details;
- Moreover, classical statistical analysis breaks down when such vast and/or complex data are present;

- A better alternative is to use **automated discovery tools** to analyze the raw data and extract high-level information for the decision maker [Hand et al., 2001];



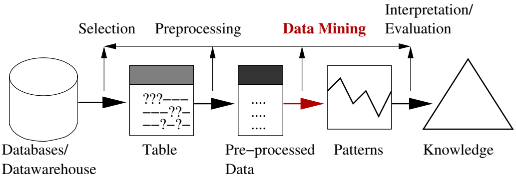
Knowledge Discovery in Databases (KDD)

"the overall process of discovering useful knowledge from data".

Data Mining (DM)

"application of algorithms for extracting patterns (models) from data".
This is a particular step of the KDD process (yet "Data Mining" is a more catchy term than KDD).

Knowledge Discovery from Databases and Data Mining [Fayyad et al., 1996]



Knowledge Discovery in Databases (KDD)

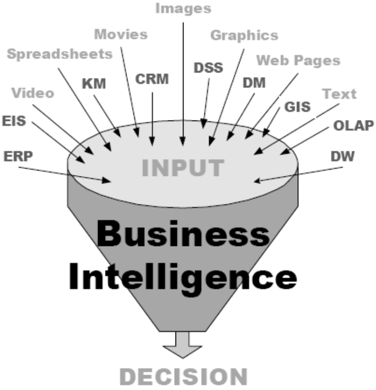
"the overall process of discovering useful knowledge from data".

Data Mining (DM)

"application of algorithms for extracting patterns (models) from data".
This is a particular step of the KDD process (yet "Data Mining" is a more catchy term than KDD).

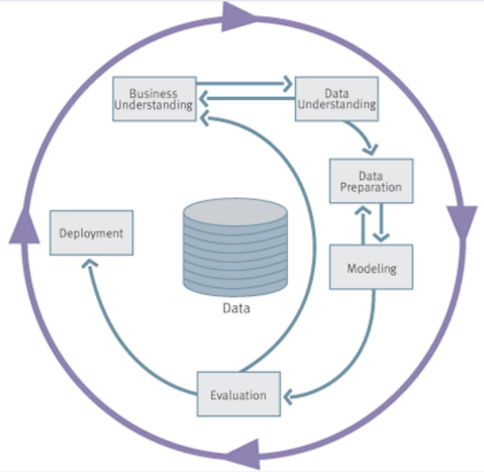
Business Intelligence and Data Mining [E. Turban and King, 2007]

- BI: "Umbrella term that includes architectures, tools, databases, applications and methodologies."
- The process of BI is to transform data into information, then to decisions and finally actions.



DM Methodologies: CRISP-DM (http://www.crisp-dm.org/):

- Tool-neural process, developed to increase the success of DM projects.
- Backed by Daimler-Chrysler, SPSS and NCR.
- Consists of six iterative and interactive phases:



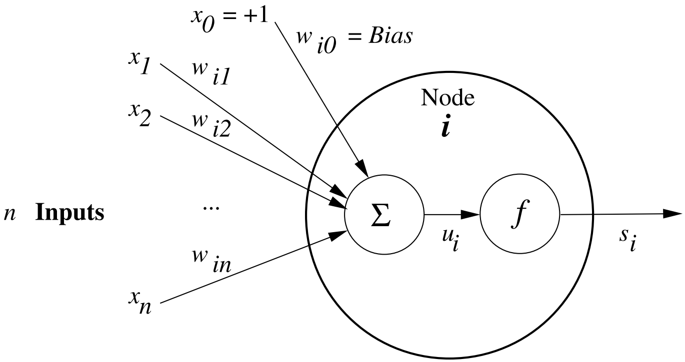
DM goals [Fayyad et al., 1996]

- **Classification** – labeling a data item into one of several predefined classes (e.g. diagnosing a disease according to patient's symptoms);
- **Regression** – mapping a set of attributes into a real-value variable (e.g. stock market prediction);
- **Clustering** – searching for natural groupings of objects based on similarity measures (e.g. segmenting clients of a database marketing study); and
- **Link analysis** – identifying useful associations in transactional data (e.g. "64% of the shoppers who bought milk also purchased bread").

DM methods

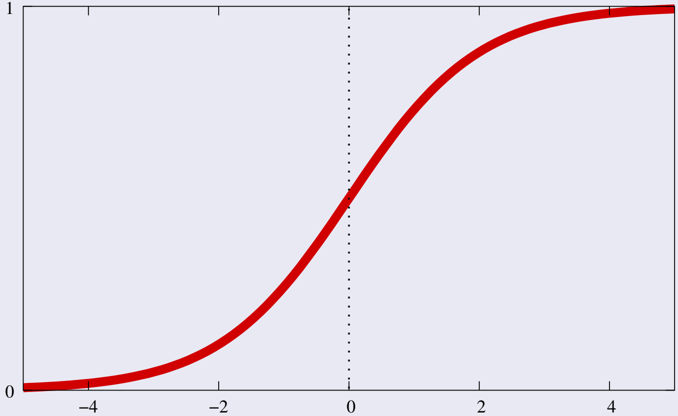
- Several methods available, with the distinction being based on two issues: model representation and search method used.
- Each own with its advantages and disadvantages: performance, computational effort and scalability, easy of use, easy to extract knowledge from, ...
- Some examples:
  - **Classification**: Decision Tree, Random Forest, Classification Rules, Linear Discriminant Analysis, Naive Bayes, Logistic Regression, MLP, RBF, SVM, ...
  - **Regression**: Regression Tree, Random Forest, Multiple Regression, MLP, RBF, SVM, ...
  - **Clustering**: K-means, EM, Single linkage, Ward's hierarchical method, Kohonen SOM, ...

- Feedforward neural network where each node **outputs** an activation function applied over the weighted sum of its **inputs**: $s_i = f(w_{i,0} + \sum_{j \in I} w_{i,j} \times s_j)$



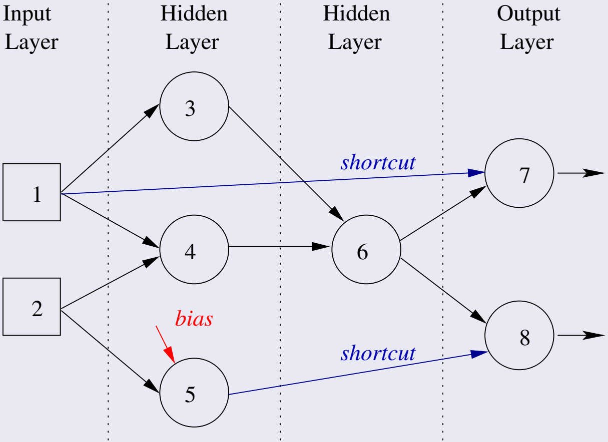
Activation functions

- Linear: $y = x$;
- Tanh: $y = tanh(x)$;
- **Logistic** or Sigmoid (most used): $y = \frac{1}{1+e^{-x}}$;



Architecture/Topology

- Only **feedforward** connections exist;
- Nodes are organized in **layers**;



Why Data Mining with MLPs? [Sarle, 2005]

- **Popularity** - the most used Neural Network, with several off-the-shelf packages available;
- **Universal Approximators** - general-purpose models, with a huge number of applications (e.g. classification, regression, forecasting, control or reinforcement learning);
- **Nonlinearity** - when compared to other data mining techniques (e.g. decision tree) MLPs often present a higher predictive accuracy;
- **Robustness** - good at ignoring irrelevant inputs and noise;
- **Explanatory Knowledge** - Difficult to explain when compared with other algorithms (e.g. decision trees), but it is possible to extract rules from trained MLPs.

Other methods can/should be also used (RBF, SVM, ...)!

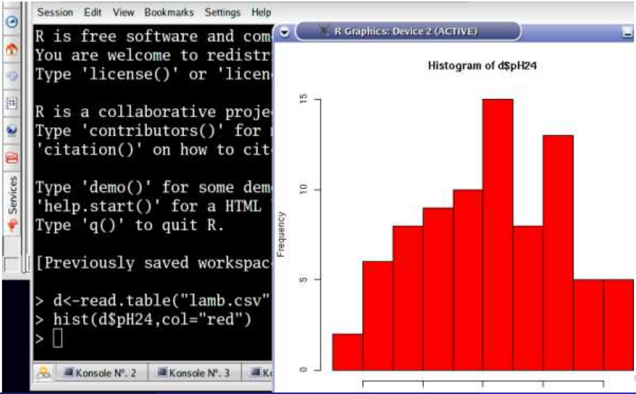
Commercial Software

- SPSS Clementine (74, 53 alone or with SPSS)
- Salford CART, MARS, TreeNet, RF (72, 34 alone)
- SPSS (68, 38 alone or with Clementine)
- Excel (61, 1 alone)
- SAS (55, 6 alone or with SAS EM)
- KXEN (32, 25 alone)
- SAS Enterprise Miner (24, 6 alone or with SAS)
- MATLAB (22,1 alone)
- SQL Server (20, 2 alone)
- Other commercial tools (12)
- Angoss (8)
- Your own code (50, 3 alone)

Free/Open Source Data Mining Software

- RapidMiner (72, 49 alone)
- R (39, 4 alone)
- Weka (36, 4 alone)

- Free open source and high-level matrix programming language;
- Provides a powerful suite of tools for statistical and graphical analysis;
- The RMiner library [Cortez, ress] facilitates the use of NN and SVM in data mining;
- Used in the 2 case studies presented (Intensive Care and Meat Quality);



Supervised Learning – input/output mapping (e.g. **classification** or **regression**):

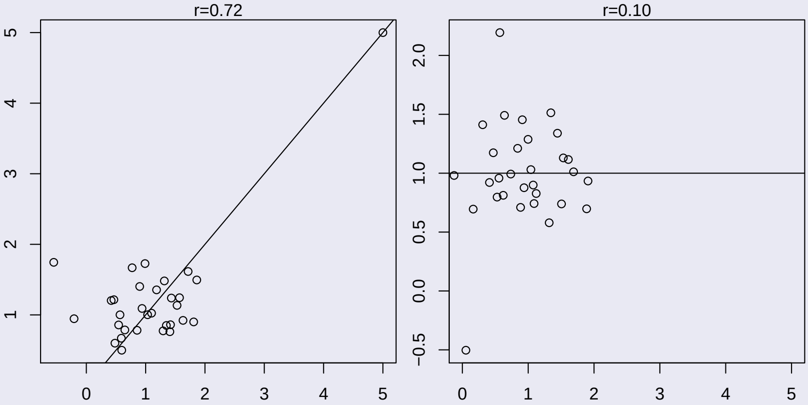
- **Data Collection** - learning samples must be representative, hundred/thousand of examples are required;
- **Preprocessing** - data transformation, dealing with missing data, outliers, ...;
- **Feature Selection** - what are the relevant inputs?
- **Modeling** – network design, training and performance assessment;
- **Prediction** – feed the fitted model with new data and interpret the output;
- **Explanatory Knowledge** – input importance (e.g. by sensitivity analysis) and extraction of rules;

Handling Missing Data ('?', 'NA', ...) [Brown and Kros, 2003]:

- Use complete data only (delete cases or variables);
- **Data Imputation**, substitute by:
  - Value given by an expert (**case substitution**);
  - Mean, median or mode;
  - Value from another database (**cold deck**);
  - Value of most similar example (**hot deck**);
  - Value estimated by a regression model (e.g. linear regression);
  - Combination of previous methods (**multiple imputation**);

Outliers

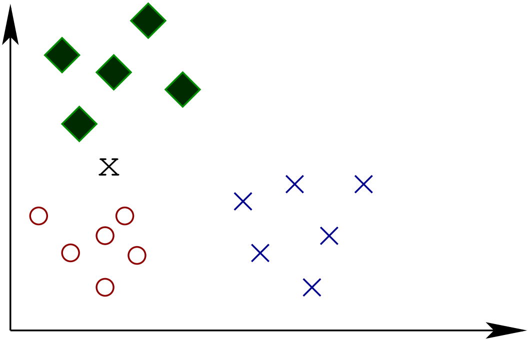
- Due to errors in data collection or rare events;
- Not related with the target variable, they prejudice the learning;
- Solution: use of experts, data visualization, statistical analysis, ...



Non numerical variable remapping [Pyle, 1999]

- Only numeric data can be fed into MLP, RBF, SVM, ...;
- **Binary** attributes can be coded into 2 values (e.g. {-1, 1} or {0, 1});
- **Ordered** attributes can be encoded by preserving the order (e.g. {low $\rightarrow$ -1, medium $\rightarrow$ 0, high $\rightarrow$ 1});
- **Nominal** (non-ordered with 3 or more classes) attributes:
  - **1-of-C** or 1-of-(C-1) remapping – use one binary variable per class (generic);
  - Other remappings – requires domain knowledge (e.g. a **state** can be coded into 2 variables, the horizontal and vertical position in a 2D map);

Example of 1-of-C remapping



- Attribute **color** = {Red, Blue, Green};
- With the linear mapping {Red → -1 , Blue → 0, Green → 1} it is impossible to describe **X**, which is half green and half red;
- With the **1-of-C** mapping { Red → 1 0 0, Blue → 0 1 0, Green → 0 0 1 }, **X** could be represented by: 0.5 0 0.5;

Rescaling/Normalization [Sarle, 2005]

- Several methods (MLP, SVM, ...) will improve learning if all **Inputs** are rescaled into the same range with a 0 mean:
  - $y = \frac{x - \overline{x}}{s}$ (**standardization** with mean 0 and standard deviation 1)
- **Outputs** limited to the [0,1] range if logistic function is used ([-1,1] if tanh).
  - $y = \frac{(x - min)}{max - min}$ (**linear scaling** with range $[0, 1]$)

Feature Selection

- Selection of the subset of relevant features. Why?
  - To reduce storage and measurement requirements;
  - To facilitate data visualization/comprehension;
  - Non relevant features/attributes will **increase the model complexity** and **worst performances** may be achieved.

Feature Selection methods [Witten and Frank, 2005]:

- A priori knowledge (e.g. the use of experts);
- **Filter** and **Wrapper** algorithms;
- Correlation analysis (only measures linear effects);
- Trial-and-error blind search (e.g. test some subsets and select the subset with the best performance);
- Hill-climbing search (e.g. forward and backward selection);
- Beam search (e.g. genetic algorithms);

Confusion matrix [Kohavi and Provost, 1998]

- Matches the **predicted** and **actual** values;
- The 2×2 *confusion matrix*:

$\downarrow$ **actual** $\backslash$ **predicted** $\rightarrow$	**negative**	**positive**
negative	**TN**	*FP*
positive	*FN*	**TP**

- Three accuracy measures can be defined:
  - the **Accuracy** $= \frac{TN+TP}{TN+FP+FN+TP} \times 100$ (%) (use if FP/FN costs are equal);
  - the **Sensitivity** (*Type II Error*) $= \frac{TP}{FN+TP} \times 100$ (%) ;
  - the **Specificity** (*Type I Error*) ; $= \frac{TN}{TN+FP} \times 100$ (%)

Classification Metrics

Confusion matrix [Kohavi and Provost, 1998]

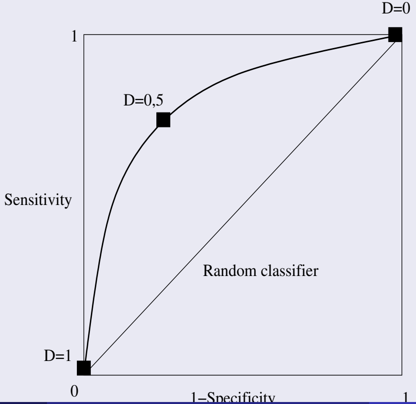
- Matches the **predicted** and **actual** values;
- The 2×2 *confusion matrix*:

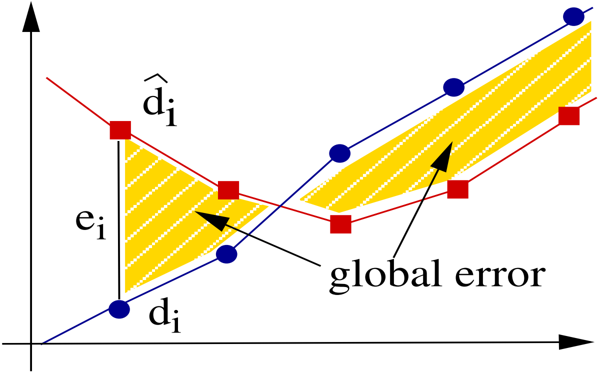
$\downarrow$ **actual** $\setminus$ **predicted** $\rightarrow$	**negative**	**positive**
negative	**TN**	*FP*
positive	*FN*	**TP**

- Three accuracy measures can be defined:
  - the **Accuracy** $= \frac{TN+TP}{TN+FP+FN+TP} \times 100\,(\%)$ (use if FP/FN costs are equal);
  - the **Sensitivity** (*Type II Error*) $= \frac{TP}{FN+TP} \times 100\,(\%)$;
  - the **Specificity** (*Type I Error*) ; $= \frac{TN}{TN+FP} \times 100\,(\%)$

Receiver Operating Characteristic (ROC) [Fawcett, 2003]

- Shows the behavior of a 2 class classifier ($y \in [0,1]$) when varying a decision parameter $D \in [0,1]$ (e.g. True if $y > 0.5$);
- The curve plots 1−Specificity (x−axis) vs the Sensitivity;
- Global performance measured by the **Area Under the Curve (AUC)**: $AUC = \int_0^1 ROC\,dD$ (the perfect AUC value is 1.0);





- The **error** e is given by: $e = d - \widehat{d}$ where d denotes the desired value and the $\widehat{d}$ estimated value (given by the model);

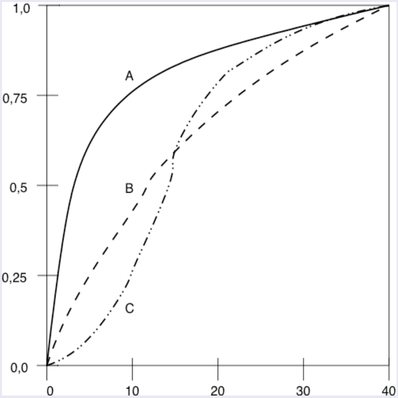
Given a dataset with the function pairs $x_1 \rightarrow d_1, \cdots, x_N \rightarrow d_N$, we can compute:

Error metrics

- **Mean Absolute Deviation (MAD)**: $MAD = \frac{\sum_{i=1}^{N} |e_i|}{N}$
- **Sum Squared Error (SSE)**: $SSE = \sum_{i=1}^{N} e_i^2$
- **Mean Squared Error (MSE)**: $MSE = \frac{SSE}{N}$
- **Root Mean Squared Error (RMSE)**: $RMSE = \sqrt{MSE}$
- **Relative MAD** (RMAD, scale independent): $RMAD = MAD/MAD_{\text{baseline}} \times 100$ (%), where baseline often denotes the average predictor.
- **Relative Root Mean Squared** (RRMSE, scale independent): $RRMSE = RMSE/RMSE_{\text{baseline}} \times 100$ (%)
- ...

Regression Error Characteristic (REC) curves [Bi and Bennett, 2003]

- Used to compare regression models;
- The curve plots the error tolerance (x-axis), given in terms of the absolute or squared deviation, versus the percentage of points predicted within the tolerance (y-axis);



Validation method: how to estimate the performance? [Flexer, 1996]

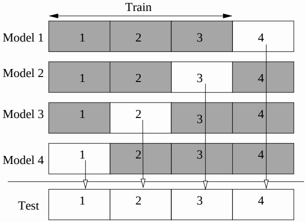
Holdout

Split the data into two exclusive sets, using random sampling:

- **training**: used to fit the model (2/3);
- **test**: used to measure the performance (1/3).

K-fold, works as above but uses rotation:

- data is split into K exclusive folds of equal size (10-fold most used);



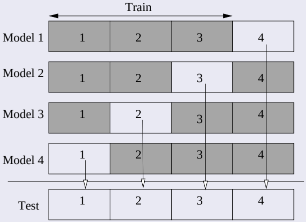
Holdout

Split the data into two exclusive sets, using random sampling:

- **training**: used to fit the model (2/3);
- **test**: used to measure the performance (1/3).

K-fold, works as above but uses rotation:

- data is split into K exclusive folds of equal size (10-fold most used);



MLP Training Algorithm

Gradient-descent [Riedmiller, 1994]:

- **Backpropagation (BP)** - most used, yet may be slow;
- Other algorithms: **Backpropagation with Momentum**; **QuickProp**; **RPROP**; **BGFS**, **Levenberg-Marquardt**, ...

Evolutionary Computation [Rocha et al., 2007]

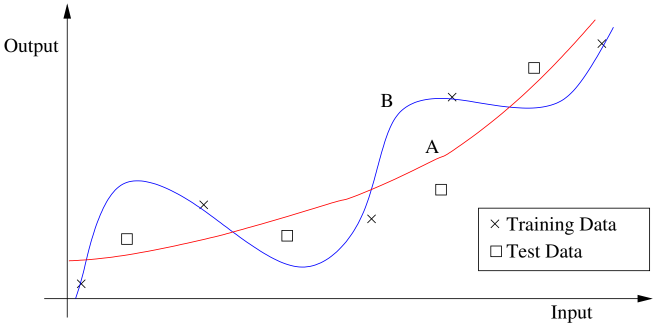
- May overcome local minima problems;
- Can be applied when no gradient information is available (reinforcement learning);

- The MLP weights are randomly initialized within small ranges (e.g. [-0.7;0.7]);
- Each training may converge to a different (local) minima;

Solutions

- Use of **multiple** trainings, selecting the *MLP* with lowest error;
- Use of **multiple** trainings, computing the average error of the MLPs;
- Use of **ensembles**, where the final output is given as the average of the MLPs;

Overfitting [Sarle, 2005][Hastie et al., 2001]



- If possible use **large datasets**: $N \gg \#model - parameters$;
- **Model Selection**: apply several models and then choose the best model;
- **Regularization**: use learning penalties or restrictions (weight decay);

Linear learning when:

- there are no hidden layers; or
- only linear activation functions are used.

Nonlinear learning:

- Any continuous function mapping can be learned with one hidden layer;
- Complex discontinuous functions can be learned with more hidden layers;

Typical design: One hidden layer of H hidden nodes.

Some MLP common design rules

Output nodes:

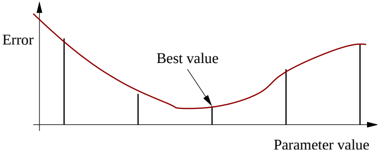
Often, it is better to perform **one** classification/regression task per network; i.e., use **C/1** output node(s).

Activation Functions:

- Hidden Nodes: use the **logistic**;
- Output Nodes: **logistic** if outputs bounded; else use the **linear** function;

Grid-Search Hyperparameter Tuning

- Simple approach, where one (or more) parameters are scanned through a given range.
- Range example for MLP hidden nodes: $H \in \{0,2,4,...,20\}$.
- Variants: two-level greedy grid-search – search at the first level, after finding the best value, a second pass is taken, using a smaller range and step;



Other approaches:

- **Hill-climbing**: one solution is tested at a given time
- **Beam search**: with population of solutions (e.g. **Evolutionary Computation**).

Explanatory Knowledge (MLP, RBF, SVM, ...)

In DM, besides obtaining a high predictive performance, it is also important to provide **explanatory knowledge**: what has the model learned?

Measuring Input Importance [Kewley et al., 2000]

- Use of sensitivity analysis, measured as the variance (V_a) produced in the output (y) when the input attribute (a) is moved through its entire range:

$$\begin{array}{rcl} V_a & = & \sum_{i=1}^{L} (y_i - \overline{y})/(L-1) \\ R_a & = & V_a / \sum_{j=1}^{A} V_j \end{array} \quad (1)$$

- A denotes the number of input attributes and R_a the relative importance of the a attribute;
- The y_i output is obtained by holding all input variables at their average values; the exception is x_a, which varies through its range with L levels;

Extraction of rules from fitted models (MLP, SVM, ...) [Tickle et al., 1998]

- **Pedagogical** techniques extract the direct relationships between the inputs and outputs of the model;
- By using a black-box point of view, less computation is required and a simpler set of rules may be achieved.
- An example will be shown in case study I.

MLPs vs Support Vector Machines (SVMs)

SVMs present **theoretical advantages** (e.g. absence of local minima) over MLPs and several comparative studies have reported **better predictive performances**!

Yet:

- SVM algorithms (may) require more computational effort for large datasets;
- Under reasonable assumptions, MLPs require the search of one parameter (hidden nodes or the decay) while SVMs require two or more (C, γ, ϵ, ...);
- MLPs can be applied in **real-time**, control & reinforcement or dynamic/changing environments;

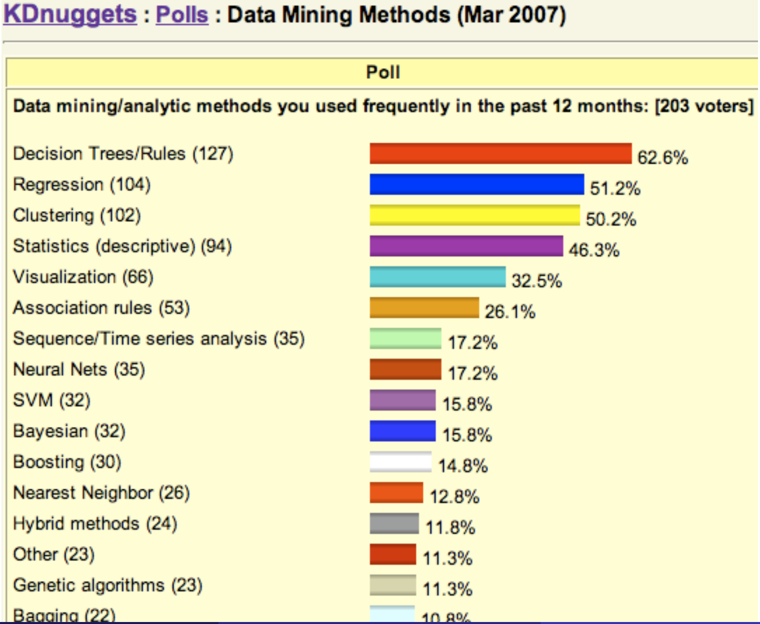
MLPs vs Support Vector Machines (SVMs)

SVMs present **theoretical advantages** (e.g. absence of local minima) over MLPs and several comparative studies have reported **better predictive performances**!

Yet:

- SVM algorithms (may) require more computational effort for large datasets;
- Under reasonable assumptions, MLPs require the search of one parameter (hidden nodes or the decay) while SVMs require two or more (C, γ, ϵ, ...);
- MLPs can be applied in **real-time**, control & reinforcement or dynamic/changing environments;

The most used DM models?



Case Study I: Intensive Care Medicine (Classification) [Silva et al., 2008]

More details at:
Á. Silva, **P. Cortez**, M.F. Santos, L. Gomes and J. Neves.
Rating Organ Failure via Adverse Events using Data Mining in the Intensive Care Unit, In **Artificial Intelligence in Medicine**, Elsevier, 43 (3): 179–193, 2008. ISSN:0933-3657.

Intensive Care Units (ICU)

- In the last decades, a worldwide expansion occurred in the number of **Intensive Care Units (ICUs)**;
- Scoring the severity of illness has become a daily practice, with several metrics available (e.g. SAPS II, SOFA);
- These scores have been used to improve the quality of intensive care and guide local planning of resources;
- Most of these scores are static (i.e. use data collected only on the first day);
- More recently, dynamic (or daily updated) scores have been designed, such as the **sequential organ failure assessment (SOFA)**;

Motivation (II)

SOFA score

- Six organ systems (respiratory, coagulation, hepatic, cardiovascular, neurological and renal) are scored from 0 to 4, according to the degree of failure;
- Expert-driven score: a panel of experts selected a set of variables and rules based on their personal opinions;
- Widely used in European ICUs;

Issues not yet solved:

- It is not clear how many daily times some variables (e.g. platelets, bilirubin) should be measured;
- No risk (i.e. probability) is provided for the outcome of interest (i.e. organ failure);

Motivation (II)

Bedside Monitoring Data

- Universal and routinely registered during patient ICU stay;
- The relationships within these biometrics are complex, nonlinear and not fully understood;
- Monitoring analysis is not standardized and mainly relies on the physicians knowledge and experience;
- The laboratory data usually depend on previous physiological impairments, thus using only biometric data should allow a more adequate evaluation and early therapeutic intervention
- Yet, an high amount of data available (several biometrics with too much detail), generating alarms that need to be interpreted;
- In previous work [Silva et al., 2006], it has been shown that **adverse events** of four biometrics have an impact on the mortality outcome of ICU patients;

Aim

- The main goal is to explore the impact of the adverse events, during the last 24h, on the current day organ risk condition (i.e. normal, dysfunction or failure)
- As a secondary goal, two DM techniques (i.e. Logistic Regression and NN) are evaluated and compared.

Data Collection

- A **EURICUS II** derived database was adopted, with records taken from 9 EU countries and 42 ICUs, during 10 months, from 1998 until 1999;
- Data manually collected by the nursing staff (every hour);
- The registered data was submitted to a double check, using both local (i.e. ICU) and central levels (i.e. Health Services Research Unit of the Groningen University Hospital, the Netherlands).
- The latter unit was used to gather the full database.

Preprocessing

- After a consult with ICU specialists, the patients with age lower than 18, burned or bypass surgery were discarded;
- Also, the last day of stay data entries were discarded, since the SOFA score is only defined for a 24h time frame and several of these patients were discharged earlier;
- Final database with 25215 daily records taken from 4425 patients.

Protocol for the out of range physiologic measurements

	BP	**SpO_2**	**HR**	**UR**
Normal Range	$90 - 180$mmHg	$\geq 90\%$	$60 - 120$bpm	≥ 30ml/h
Event[a]	$\geq$ 10min.	$\geq$ 10min.	$\geq$ 10min.	$\geq$ 1h
Event[b]	$\geq$ 10min. in 30min.	$\geq$ 10min. in 30min.	$\geq$ 10min. in 30min.	–
Critical Event[a]	$\geq$ 1h	$\geq$ 1h	$\geq$ 1h	$\geq$ 2h
Critical Event[b]	$\geq$ 1h in 2h	$\geq$ 1h in 2h	$\geq$ 1h in 2h	–
Critical Event[c]	$<$ 60mmHg	$<$ 80%	$<$ 30bpm $\vee >$ 180bpm	$\leq$ 10ml/h

BP - blood pressure, HR - heart rate, SpO_2 - pulse oximeter oxygen saturation, UR - urine output.

a Defined when continuously out of range.

b Defined when intermittently out of range.

c Defined anytime.

The intensive care variables

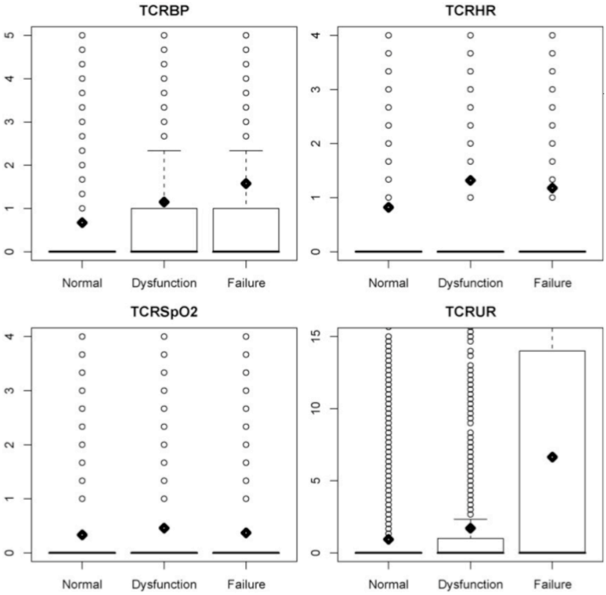
Attribute	Description	Min	Max	Mean[a]
admtype	admission type	Categorical[b]		
admfrom	admission origin	Categorical[c]		
SAPS II	SAPS II score	0	118	40.9±16.4
age	age of the patient	18	100	62.5±18.2
NBP	daily number of blood pressure events	0	24	0.8±1.9
NHR	daily number of heart rate events	0	24	0.6±2.3
$NSpO_2$	daily number of oxygen events	0	24	0.4±1.8
NUR	daily number of urine events	0	24	1.0±3.0
NCRBP	daily number of critical blood pressure events	0	10	0.3±0.7
NCRHR	daily number of critical heart rate events	0	10	0.2±0.6
$NCRSpO_2$	daily number of critical oxygen events	0	6	0.1±0.4
NCRUR	daily number of critical urine events	0	7	0.4±0.8
TCRBP	time of critical blood pressure events (% of 24h)	0	24.7	0.8±2.7
TCRHR	time of critical heart rate events (% of 24h)	0	24.7	1.0±3.4
$TCRSpO_2$	time of critical oxygen events (% of 24h)	0	24.7	0.4±2.1
TCRUR	time of critical urine events (% of 24h)	0	24.7	1.6±4.5

a mean and sample standard deviation.

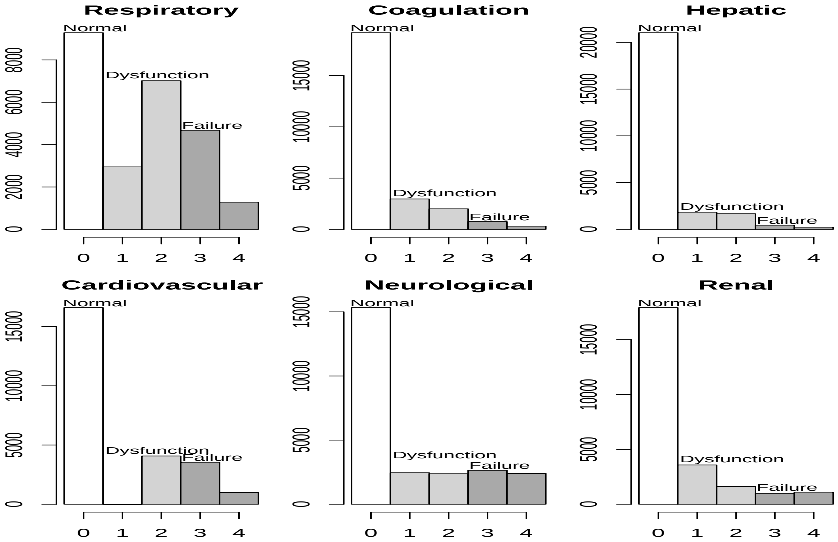
b 1 - unscheduled surgery, 2 - scheduled surgery, 3 - medical.

c 1 - operating theatre, 2 - recovery room, 3 - emergency room, 4 - general ward, 5 - other ICU, 6 - other hospital, 7 - other sources.

Boxplots of Critical Events per Renal Condition



Organ condition prevalence (histograms)



Discrimination: AUC of ROC

- Multi-class problem: one ROC per class and then compute a global AUC value weighted by the class prevalence;

Calibration: Brier Score

- The ROC measures the **discrimination** power, but in medicine it is also important to have a good **calibration**: the predictions should be close to the true probabilities of the event;
- Calibration will be measured using the Brier score;
- The Brier Score (also known as MSE) for a two-class scenario is: $Brier(c_j) = \frac{1}{N}\sum_{i=1}^{N}(p_j^i - \widehat{p}_j^i)^2$
- Inspired in the multi-class AUC metric, the global Brier score is defined as: $Brier_{Global} = \sum_{c_i \in C} Brier(c_i) \cdot prev(c_i)$

Multinomial Logistic Regression (MLR)

- The logistic regression is the most popular model within ICU physicians;
- The MLR is the extension to multi-class tasks:

$$\begin{array}{rcl} \widehat{p}_j & = & \frac{exp(\eta_j \mathbf{x})}{\sum_{k=1}^{\#C} exp(\eta_k \mathbf{x})} \\ \eta_j(\mathbf{x}) & = & \sum_{i=1}^{I} \beta_{j,i} x_i \end{array} \quad (2)$$

  where $\beta_{j,0}, \ldots, \beta_{j,I}$ denotes the parameters of the model, and $x_1, \ldots, x_I$ the dependent variables;
- This model requires that $\eta_k(\mathbf{x}) \equiv 0$ for one $c_k \in C$ (the baseline group) and this assures that $\sum_{j=1}^{\#C} \widehat{p}_j = 1$;

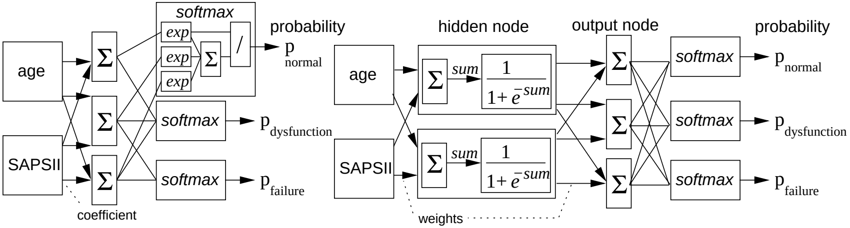
Neural Network (NN)

- Fully connected MLPs with **bias** connections, **one hidden layer** of H nodes and **logistic** activation functions;
- Linear function used at the $\#C$ output nodes;
- The final probability is given by:

$$\begin{aligned} \widehat{p}_j &= \frac{exp(y_j)}{\sum_{k=1}^{\#C} exp(y_k)} && \text{(softmax function)} \\ y_i &= w_{i,0} + \sum_{m=I+1}^{I+H} f(\sum_{n=1}^{I} x_n w_{m,n} + w_{m,0}) w_{i,n} \end{aligned} \quad (3)$$

where y_i is the output of the network for the node i; $f = \frac{1}{1+exp(-x)}$ is the logistic function; I represents the number of input neurons; $w_{d,s}$ the weight of the connection between nodes s and d; and $w_{d,0}$ is the bias.

MLR vs NN



softmax
exp
exp
exp
Σ
/
probability
p normal
age
SAPSII
Σ
Σ
Σ
softmax
p dysfunction
softmax
p failure
coefficient
hidden node
output node
probability
age
SAPSII
Σ
sum
1
1+ e −sum
Σ
sum
1
1+ e −sum
weights
Σ
Σ
Σ
softmax
p normal
softmax
p dysfunction
softmax
p failure

Feature and Model selection

- A backward feature selection based on the sensitivity analysis will be used;
- H will be fixed to the median of the grid range during the feature selection phase;
- After feature selection, the number of hidden nodes (H) is be tuned using a simple grid search $H \in \{2, 4, 6, 8, 10\}$;
- For both feature and H searches, the training data is randomly split into training (66.6%) and validation (33.3%) sets.
- The model with the lowest validation error is selected and the final model is retrained with all available data.

Experiments

- The **R** (statistical tool, open source) environment and **RMiner** library (**nnet** and **kernlab** packages) was used in all experiments [Cortez, ress];
- Training with the BGFS algorithm (quasi-newton method), set to maximize the likelihood;
- Continuous inputs were scaled into a zero mean and one standard deviation range; the nominal inputs were encoded into *1-of-*$(C-1)$ binary variables. **Admtype** example: $1 \rightarrow (0\,0)$; $2 \rightarrow (1\,0)$; and $3 \rightarrow (0\,1)$.
- To compare the learning models, 20 runs of a **5-fold cross-validation** [Kohavi, 1995] were executed (in a total of 20×5 simulations).
- Paired statistical comparison using the Mann-Whitney non-parametric test at the 95% confidence level;

Discrimination Results (values of AUC>70% are in bold)

	Normal		Dysfunction		Failure		Global	
Organ	MLR	NN	MLR	NN	MLR	NN	MLR	NN
respiratory	67.2	69.5	59.2	61.0	65.6	68.9	63.6	66.0
coagulation	63.6	65.5	60.1	62.0	**72.6**	**73.9**	63.3	65.1
hepatic	64.7	66.7	62.5	64.2	**72.6**	**76.0**	64.6	66.6
cardiovascular	67.9	**71.2**	63.8	65.6	67.3	**71.0**	67.1	**70.2**
neurological	**70.0**	**72.1**	58.8	61.2	**74.7**	**76.7**	68.8	**70.9**
renal	69.4	**70.7**	66.0	66.8	**73.5**	**76.1**	69.1	**70.4**
Average	67.1	69.3	61.7	63.5	**71.0**	**73.8**	66.1	68.2

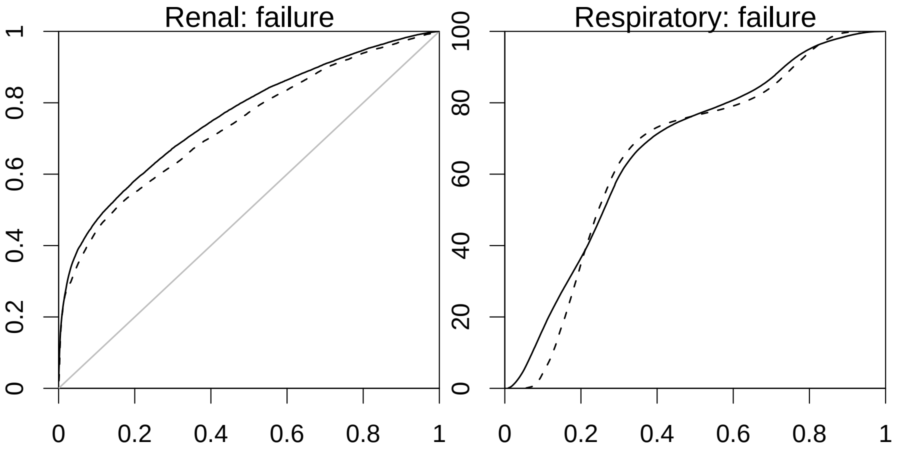
- In all cases, the NN/MLR differences are significant.
- The median number of H is 8 for all organs (except neurological where H=10);
- The feature selection discarded an average of two attributes;

Calibration Results (Brier score values)

Organ	Normal MLR	Normal NN	Dysfunction MLR	Dysfunction NN	Failure MLR	Failure NN	Global MLR	Global NN
respiratory	0.213	**0.204**	0.233	**0.230**	0.171	**0.166**	0.211	**0.205**
coagulation	0.173	**0.171**	0.155	**0.154**	0.038	0.038	0.134	**0.133**
hepatic	0.132	**0.130**	0.116	0.116	0.026	**0.025**	0.101	**0.100**
cardiovascular	0.205	**0.197**	0.132	**0.130**	0.138	**0.133**	0.160	**0.155**
neurological	0.208	**0.202**	0.153	**0.151**	0.136	**0.132**	0.169	**0.165**
renal	0.182	**0.179**	0.155	**0.155**	0.065	**0.063**	0.144	**0.142**
Average	0.185	**0.181**	0.157	**0.156**	0.096	**0.093**	0.153	**0.150**

Values in bold denote statistical significance when compared with MLR.

Results: ROC (renal failure) and REC (respiratory failure)

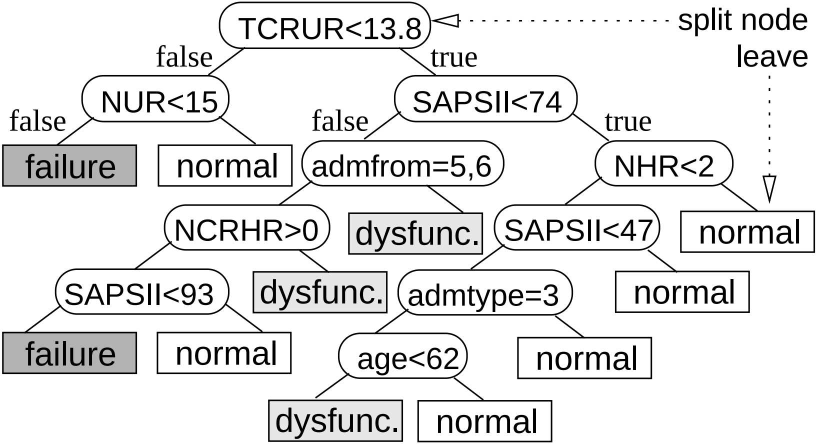


Input Relevance (NN)

Organ	admtype	admfrom	SAPS II	age	BP*	HR*	SpO_2*	UR*
respiratory	16.8	7.8	15.1	10.0	19.9	8.1	17.1	5.2
coagulation	30.9	10.8	12.7	7.0	7.5	2.6	18.1	10.4
hepatic	23.1	7.8	12.1	10.8	9.1	5.1	17.0	15.0
cardiovascular	14.1	17.3	16.5	12.8	9.8	9.6	13.4	6.5
neurological	31.2	10.2	15.6	7.5	17.3	3.5	10.4	4.3
renal	2.3	13.6	26.6	9.9	5.1	6.4	19.8	16.3
Average	19.7	11.3	16.4	9.7	11.4	5.9	16.0	9.6

* – All attributes related to the variable where summed (number of events, critical events and the time).

Knowledge extraction (Decision Tree example for the renal organ)



Conclusions (I)

Primary goal

- A data-driven analysis was performed on a large ICU database, with an emphasis on the use of daily adverse events, taken from four commonly monitored biometrics;
- The obtained results show that adverse events are important intermediate outcomes;
- It is possible to use DM methods to get knowledge from easy obtainable data, thus opening room for the development of intelligent clinical alarm monitoring.
- **Future work**: test this approach in a real environment with an on-line learning (pilot project **INTCare**, Hospital S. António).

Conclusions (II)

Second goal

- To reduce the bias towards a given model, we adopted the default suggestions of the R tool (the only exception H, set using a simple grid search);
- The default settings are more likely to be used by common (non expert) users, thus this seems a reasonable assumption for a fair comparison.
- With the same inputs, the NNs outperform the Logistic Regression;

Case Study II: Lamb Meat Quality (Regression) [Cortez et al., 2006]

More details at:

P. Cortez, M. Portelinha, S. Rodrigues, V. Cadavez and A. Teixeira. Lamb Meat Quality Assessment by Support Vector Machines. In Neural Processing Letters, Springer, 24 (1): 41-51, 2006. ISSN:1370-4621.

Motivation

Meat Quality

- The success of meat industry relies on the ability to deliver specialties that satisfy the consumer's taste;
- **Tenderness** is the most important factor that influences meat quality (although there are other factors such as juiciness);
- The ideal method for measuring tenderness such be accurate, fast, automated and non invasive;
- Two major approaches have been proposed to measure tenderness:
  - **Instrumental**: objective test based on a device (WBS);
  - **Sensory Analysis**: subjective test based on a taste panel (STP);
- Both approaches are invasive, expensive and time demanding, requiring laboratory work.

Motivation

Meat Quality Modeling

- An alternative is to use **carcass measurements** (e.g. pH and color), which are cheap, non invasive and can be collected 24h after slaughtering;
- The classic Animal Science approach uses **Multiple Regression** where meat features are the independent (input) variables and the output dependent target is the WBS/STP;
- Yet, these linear models will fail is nonlinearity is present;
- A better option may be the use of **Neural Networks (NN)** or **Support Vector Machines (SVM)**, flexible models with noise tolerance and nonlinear mapping capabilities, increasingly used in **Data Mining** tasks;
- The measure of input importance, also relevant within this domain, can be addressed by a **Sensitivity Analysis** procedure.

Data Collection

- This study considered lamb animals from the Trás-os-Montes northeast region of Portugal (collected from November/2002 until November/2003);
- Each entry denotes readings from a slaughtered animal;
- The dataset is quite small with 81 examples;
- In addiction, 2 (10) examples were discarded due to the presence of missing values in the WBS (STP) variables;
- The attributes were registered at the slaughterhouse and in laboratory;
- Due to their visual nature, color attributes (**a***, **b***, **dE**, **dL** and **dB***) have a high impact in consumer's perception.

Lamb Meat Data

Dataset Main Attributes

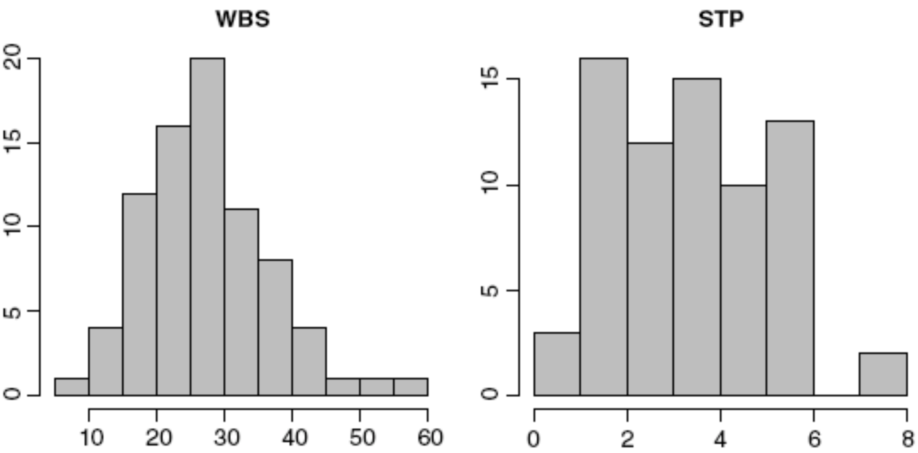
Attribute	Description	Domain
Breed	Breed type	$\{1, 2\}^a$
Sex	Lamb sex	$\{1, 2\}^b$
HCW	Hot carcass weight (kg)	$[4.1, 14.8]$
STF2	Sternal fat thickness	$[6.0, 27.8]$
C	Subcutaneous fat depth	$[0.3, 5.1]$
pH1	pH 1 hour after slaughtering	$[5.5, 6.8]$
pH24	pH 24 hours after slaughtering	$[5.5, 5.9]$
a*	Color red index	$[11.5, 22.2]$
b*	Color yellow index	$[6.5, 12.5]$
dE	Total color difference	$[46.5, 60.9]$
dL	Luminosity differential	$[-56, -39]$
dB*	Yellow differential	$[15.3, 22.5]$
WBS	Warner-Bratzler Shear force	$[9.5, 57.0]$
STP	Sensory Taste Panel	$[0.7, 7.1]$

[a] 1 – *Bragançana*, 2 – *Mirandesa*; [b] 1 – *Male*, 2 – *Female*

Output variables (WBS and STP)

- The **Warner-Bratzler Shear (WBS)** force is the major index for measuring meat tenderness (obtained in laboratory, 72 hours after slaughter);
- The **Sensory Taste Panel (STP)** measures the average rankings of 12 individuals, under a blind taste proof;
- In both cases (WBS and STP), low values suggest tender meat (high values indicate toughness).

Output histograms (WBS and STP)



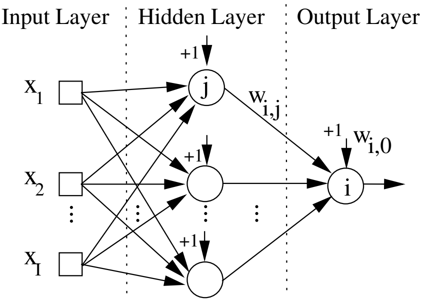
Learning Models

Regressors

- Each task (WBS and STP) is modelled separately (one model per task);
- The **Multiple Regression (MR)** model is easy to interpret and has been widely used in regression applications;
- **Neural Networks (NNs)** will be based on the **Multilayer Perceptron (MLP)**, with one hidden layer with H hidden nodes (sigmoid activation functions) and 1 output linear node;
- **Support Vector Machine (SVM)** with the gaussian kernel and -insensitive loss function;

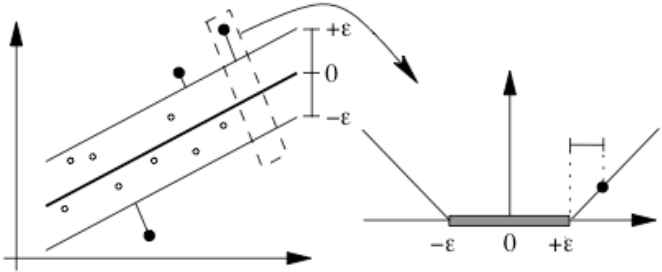
NN Setup

- Initial weights are randomly set within the range $[-0.7, +0.7]$;
- $R = 3$ different runs of 10 training epochs are applyed and the **NN** with the lowest error is selected;
- A fixed number of hidden nodes ($H = 12$) is used;
- Model complexity is set by changing the weight decay ($\lambda in[0, 1]$);



Support Vector Machine (SVM) Setup

- Performance affected by 3 parameters: C, ϵ and γ;
- To reduce the search space, C and ϵ values were set using the heuristics proposed in [Cherkassy and Ma, 2004]: $C = 3\sigma_y$, if $\overline{y} = 0$, $\widehat{\sigma} = 1.5/N \times \sum_{i=1}^{N}(y_i - \widehat{y}_i)^2$ and $\epsilon = \widehat{\sigma}/\sqrt{N}$. σ_y denotes the standard deviation of the output (y) and $\widehat{y}$ is the value predicted by the 3-nearest neighbor algorithm.
- Model complexity is set by changing the γ value;



Model Selection

Hyperparameters (λ and γ) tuned by a two level grid-search;

- First level will search the best value (λ_1 or γ_1) within the ranges $\lambda \in \{0.00, 0.01, \ldots, 0.20\}$ or $\gamma \in \{2^{-15}, 2^{-13}, ..., 2^3\}$;
- Second level proceeds with a fine tune within the range $\lambda_2 \in \{\lambda_1 - 0.005, \ldots, \lambda_1 - 0.001, \lambda_1 + 0.001, \ldots, \lambda_1 + 0.004\} \wedge \lambda_2 \geq 0$ or $\gamma_2 \in \{2^{s_1 - 1.75}, \ldots, 2^{s_1 - 0.25}, 2^{s_1 + 0.25}, \ldots, 2^{s_1 + 1.25}\} \wedge \gamma_2 \geq 0$.
- Prediction accuracy (*MAD*) in the grid-search is estimated by adopting a 10-fold cross-validation over the training data;
- After obtaining the best parameter, the final model is retrained using the whole training data.

Feature Selection (FS)

- **Backward selection** iterative approach, starting with 12 inputs and stopping when half of the features are discarded;
- **Sensitivity Analysis** is used to delete the least relevant attribute at a given iteration;

Experiments

- The R (statistical tool, open source) environment and RMiner library (nnet and kernlab packages) was used in all experiments [Cortez, ress];
- Training with the BGFS (NN) and SMO (SVM) algorithms, set to minimize the squared error;
- 30 runs of a leave-one-out (N-fold) procedure;
- Results shown in terms of mean and t-student 95% confidence intervals;
- Regression metrics: *MAD* and *RMAD*;

Regression Results

Task	Model	Inputs	Time	*MAD*	*RMAD*
WBS	*MR*	12	53	6.22±0.00	91.42±0.00
	NN	12	69869	6.17±0.09	90.56±1.27
	SVM*	12	28202	5.73±0.04	84.16±0.52
	FSNN	6	72698	6.12±0.06	89.94±0.81
	FSSVM$^{\dagger\diamond}$	6	60554	**5.60**±0.02	**82.18**±0.33
STP	*MR*	12	46	1.24±0.00	90.31±0.00
	NN	12	60512	1.35±0.02	98.21±1.19
	SVM*	12	24536	1.22±0.01	88.48±0.83
	FSNN†	6	63345	1.25±0.02	90.91±1.16
	FSSVM$^\diamond$	6	52952	**1.21**±0.01	**88.28**±0.40

$\star$ - Statistically significant (p-value<0.05) under pairwise comparisons with the previous *MR* and *NN* models

$\dagger$ - Statistically significant under a pairwise comparison with the same model without the *FS* procedure

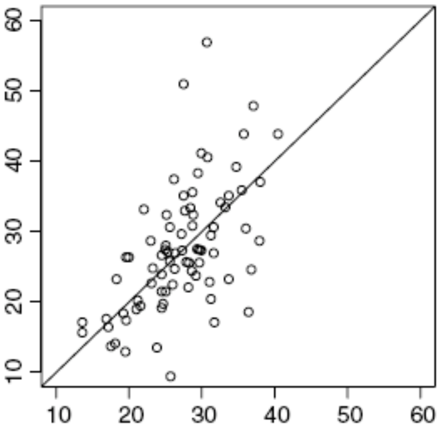
$\diamond$ - Statistically significant under a pairwise comparison with *FSNN*

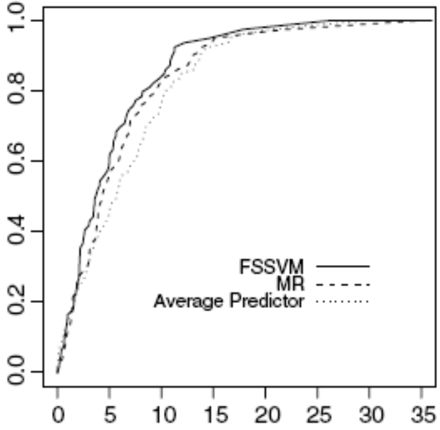
Results: Input Importance

Task	Model	Attribute								
		Bre.	HCW	STF2	pH1	pH24	a*	dE	dL	dB*
WBS	*FSNN*	0.4	7.4	5.2	0.3	1.3	**58.4**	20.2	2.9	3.6
	FSSVM	0.3	–	25.4	0.4	7.1	**32.4**	–	19.2	14.9
STP	*FSNN*	**35.3**	2.7	4.6	12.9	–	25.1	17.5	0.3	0.3
	FSSVM	**41.3**	7.8	0.7	16.0	–	26.3	–	0.3	6.9

- The differences obtained between the two tasks may be explained by psychological factors;
- The **Breed** importance increase in the **STP** contradicts the animal science theory;
- These results were discussed with the experts, which later discovered that the *Mirandesa* lambs were considered less stringy and more odor intense (due to animal stress?).

Results: scatter plot and REC for WBS





Conclusions

- The **FSSVM** algorithm outperformed other data mining methods;
- The proposed approach is much **simpler** (requiring only 6 inputs), **cheaper** than the **WBS** or **STP** procedures, and can be computed just 24 hours after slaughter;
- The drawback is the obtained accuracy, which is still high when compared with the simple constant average predictor.
- It should be stressed that the tested datasets are **very small**;
- Furthermore, modeling sensory preferences is a very difficult regression task;
- To our knowledge, this is the **first time** lamb meat tenderness is approached by neural regression models and further exploratory research needs to be performed.

Business Value

- The predictive models can be used to predict tender, moderate or tough meat;
- Different prices can be assigned to different meat quality: from premium meat (for restaurants) to minced meat (more cheap);

Future Work

- Apply this approach in a real environment, enriching the datasets by gathering more meat samples;
- Develop automatic tools for decision support and gather feedback from real users;

Bi, J. and Bennett, K. (2003).
Regression Error Characteristic curves.
In Fawcett, T. and Mishra, N., editors, *Proceedings of 20th Int. Conf. on Machine Learning (ICML)*, Washington DC, USA, AAAI Press.

Bishop, C. (1995).
Neural Networks for Pattern Recognition.
Oxford University Press.

Brown, M. and Kros, J. (2003).
Data mining and the impact of missing data.
Industrial Management & Data Systems, 103(8):611–621.

Cherkassy, V. and Ma, Y. (2004).
Practical Selection of SVM Parameters and Noise Estimation for SVM Regression.
Neural Networks, 17(1):113–126.

Cortez, P. (*InPress*).
RMiner: Data Mining with Neural Networks and Support Vector Machines using R.

In R. Rajesh (Ed.), *Introduction to Advanced Scientific Softwares and Toolboxes*.

Cortez, P., Portelinha, M., Rodrigues, S., Cadavez, V., and Teixeira, A. (2006).
Lamb Meat Quality Assessment by Support Vector Machines.
Neural Processing Letters, 24(1):41–51.

E. Turban, R. Sharda, J. A. and King, D. (2007).
Business Intelligence - A Managerial Approach.
Pearson Prentice-Hall, New Jersey, USA.

Fawcett, T. (2003).
Roc graphs: Notes and practical considerations for data mining researchers.
Technical Report HPL-2003-4, HP Laboratories Palo Alto.

Fayyad, U., Piatetsky-Shapiro, G., and Smyth, P. (1996).
Advances in Knowledge Discovery and Data Mining.
MIT Press.

Flexer, A. (1996).

Statistical evaluation of neural networks experiments: Minimum requirements and current practice.
In *Proceedings of the 13th European Meeting on Cybernetics and Systems Research*, volume 2, pages 1005–1008, Vienna, Austria.

Hand, D., Mannila, H., and Smyth, P. (2001).
Principles of Data Mining.
MIT Press, Cambridge, MA.

Hastie, T., Tibshirani, R., and Friedman, J. (2001).
The Elements of Statistical Learning: Data Mining, Inference, and Prediction.
Springer-Verlag, NY, USA.

Kewley, R., Embrechts, M., and Breneman, C. (2000).
Data Strip Mining for the Virtual Design of Pharmaceuticals with Neural Networks.
IEEE Trans Neural Networks, 11(3):668–679.

Kohavi, R. (1995).

A Study of Cross-Validation and Bootstrap for Accuracy Estimation and Model Selection.
In *Proceedings of the International Joint Conference on Artificial Intelligence (IJCAI)*, Volume 2, Montreal, Quebec, Canada, Morgan Kaufmann.

Kohavi, R. and Provost, F. (1998).
Glossary of Terms.
Machine Learning, 30(2/3):271–274.

Pyle, D. (1999).
Data Preparation for Data Mining.
Morgan Kaufmann, S. Francisco CA, USA.

Riedmiller, M. (1994).
Supervised Learning in Multilayer Perceptrons - from Backpropagation to Adaptive Learning Techniques.
Computer Standards and Interfaces, 16.

Rocha, M., Cortez, P., and Neves, J. (2007).
Evolution of Neural Networks for Classification and Regression.

Neurocomputing, 70(16-18):2809–2816.

Sarle, W. (2005).
Neural Network Frequently Asked Questions.
Available from ftp://ftp.sas.com/pub/neural/FAQ.html.

Silva, A., Cortez, P., Santos, M. F., Gomes, L., and Neves, J. (2006).
Mortality assessment in intensive care units via adverse events using artificial neural networks.
Artif Intell Med, 36:223–234.

Silva, A., Cortez, P., Santos, M. F., Gomes, L., and Neves, J. (2008).
Rating organ failure via adverse events using data mining in the intensive care unit.
Artificial Intelligence Medicine, 43(3):179–193.

Tickle, A., Andrews, R., Golea, M., and Diederich, J. (1998).
The Truth Will Come to Light: Directions and Challenges in Extracting the Knowledge Embedded Within Trained Artificial Neural Networks.
IEEE Transactions on Neural Networks, 9(6):1057–1068.

Witten, I. and Frank, E. (2005).
Data Mining: Practical Machine Learning Tools and Techniques with Java Implementations.
Morgan Kaufmann, San Francisco, CA.