Annales UMCS Informatica AI X, 1 (2010) 15-27
DOI: 10.2478/v10065-010-0032-x

Annales UMCS
Informatica
Lublin-Polonia
Sectio AI
http://www.annales.umcs.lublin.pl/

An Application of Expectation-Maximization for Model Verification

Barbara Łukawska*, Grzegorz Łukawski†, Krzysztof Sapiecha‡

Department of Computer Science, Kielce University of Technology, Al. Tysiaclecia Panstwa Polskiego 7, 25-314 Kielce, Poland.

Abstract – A description which summarizes entire and usually big set of data is called its model. The problem investigated in the paper consists in verification of models of data coming from a simulation experiment of selecting candidates for operators of mobile robot (more strictly building reliable predictive model of the data). The models are validated using train-and-test method and verified with the help of the EM (expectation-maximization) algorithm which was originally designed for solving clustering problems with missing data. Actually, the selecting is a clustering problem because the candidates are assigned to 'chosen', 'accepted' or 'rejected' subgroups. For such a case the missing data is the category (the subgroup) for which a candidate should be assigned on the basis of his activity measured during the simulation experiment. The paper explains the procedure of model verification. It also shows experimental results and draws conclusions.

1 Introduction

Nowadays, in the age of terabyte disk drives and the Internet, large sets of data are nothing unusual. The key to make use of such data is to pull out some useful information. The process of pulling the information is called data exploration. Its results, relationships and summaries are called models and patterns.

*b.lukawska@tu.kielce.pl
†g.lukawski@tu.kielce.pl
‡pesapiec@cyf-kr.edu.pl

A model is a global description which summarizes an entire, and usually large, set of data. It is 'an abstract representation of the real world processes' [1, 2]. A pattern is a local data characteristic reflecting its deviation from the general sample. A model of an object is built neither on the imagination nor the reality (that is not known). It is built on the basis of data, usually through identifying their relationships. The existence of the model does not implicate any causality (changing one of the variables describing the object does not modify another variable in a predictable way).

There are two main types of models: descriptive and predictive. Descriptive models summarize or condense data. A descriptive model makes it possible to identify real structure of data loaded with errors. Commonly, the density function models are used in this case. Predictive models help to draw conclusions about the whole population of objects described by a set of variables, or about their probable future values. One of the variables is expressed as a function of other variables. The problem of finding a predictive model becomes a classification problem, if the result variable is a categorical one. Classification predictive models divide the whole space of objects into separable decision areas (one area for each class of the objects).

The problem investigated in the paper is a that of model verification, and more strictly building a reliable predictive model. The problem of finding the best candidates for the mobile robot (mobot) operator will be used as an example [3–8]. A predictive model is required for ranking the candidates and choosing the best ones. The closer to the reality the model will be, the better decisions it will allow us to make (selection will be more reliable).

The simulation experiment, in which the candidates are trained, was used as a source of data [3–7]. The candidates were trained in virtual reality. Data about their behavior was collected. After the training the best candidates were chosen on the basis of these data.

The selection is a classification process which relies on assigning candidates into 'chosen', 'accepted' or 'rejected' subgroups of the operators. To this end, a predictive model was built using Weka [9] software. Weka is an open source software data exploration system. The four methods (naive Bayes [10], decision trees [11], decision tables [12] and linear regression [1]) were used for building the model. Unfortunately, four different variants of the model were obtained. The variants were then validated using the train-and-test method.

According to the above, all variants of the model should be verified. This is done with the help of the EM (expectation-maximization) algorithm [1, 2], which was designed for solving grouping (clustering) problems with missing data. Actually, the selection is a grouping problem because the candidates are assigned to 'chosen', 'accepted' or 'rejected' subgroups. For such a case the missing data is the category (the subgroup) for which a candidate should be assigned, on the basis of his activity measured during the simulation experiment. When the EM algorithm would be taken for solving this problem, we could get a model worked out from our experimental data.

The paper explains the procedure of model verification with the help of the EM algorithm. It presents experimental results and draws conclusions. The procedure can be expanded over derivation predictive models concerning similar problems. Section 2 contains a short introduction to modelling and data mining. In section 3 the experiment with training mobile robot operators is described. Motivation for the research is given in section 4. In section 5 the results of model verification with the help of the EM algorithm are given. Section 6 summarizes the research conclusions.

2 Modelling Data

There are two main types of data models: descriptive and predictive [1, 2]. Predictive models help to draw conclusions about the whole population of objects described by a set of variables, or about their probable future values. One of the variables is expressed as a function of other variables.

The predictive modelling has two forms: classification and regression. In classification the predictive variable is categorical one, in regression - quantitative.

Let Y denote a resulting variable and Xn predictive variables. Predictive models are following [1]:

- Linear model - represented by a linear function Y = aX + b. Linear modelling relies on approximation of a discrimination or regression function with a hyper surface (in the simplest case it has the form of a straight line). Simple optimization techniques may be used here, but the linear model often is not enough realistic.
- Addictive model - represented by a sum of components $Y = \sum a_i X_i + a_0$
- Multiplicative model - represented by a product of components $Y = \prod a_i X_i + a_0$
- Model with locally segmented structure - that contains different local relationships in different space areas (e.g. tree structures):
  - Partially linear model - where Y is locally a function of X;
  - Function of compound curves - where segments are low level polynomials.

All of the models are parametric. Hence, a result of modelling is a function, or some kind of the line (straight or curved) reflecting data relationships. Another type of the model is non-parametric, not reflecting data relationships explicitly, but determining the object value using the nearest neighbour values. Non-parametric models are as follows:

- Local neighborhood models - where Y is determined using average Y value from the nearest neighbours. Such models are of no use for summarizing data.
- Kernel models.

Classification predictive models divide the whole X space of objects into separable decision areas (one area for each class of objects).

Descriptive model characterizes all data or the process in which these data were generated. Descriptive models summarize or condense data. A descriptive model makes it possible to identify real structure of data burdened with errors. The following descriptive models could be distinguished:

- global data probability distribution models (density estimation),
- models dividing p-dimensional space into groups (cluster analysis, segmentation),
- models describing relationships between data (relationship modelling).

Usually, density function models are used in this case. There are parametric models, such as those defined by the position parameter (average value) and the scale - density function or the distribution. There are non-parametric models also, where distribution or the density are evaluated from the data.

Usually, density function models are used in this case. There are parametric models, such as those defined by the position parameter (average value) and the scale - density function or the distribution. There are non-parametric models also, where distribution or the density are evaluated from the data.

(1) Choosing the model form;
(2) Determining values of parameters using estimation (maximization or minimization of the ranking function, reflecting fitness of model and the data).

2.1 Data mining

In literature, different methods for data collection and analysis are proposed, including those of publishing results and acquiring benefits from the data mining project.

One of the data mining models is CRISP (Cross-Industry Standard Process for data mining), proposed in the middle 90-s by the European company consortium, as a public data mining standard [2]. In the CRISP model, the following project steps are proposed (Fig. 1).

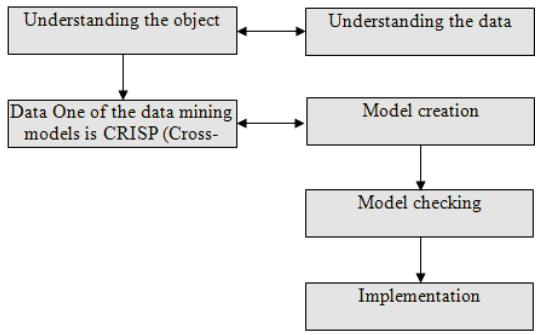

Fig. 1. CRISP data mining standard.

In Six Sigma, a different strategy is proposed. The Six Sigma is a well organized, data-based strategy for avoidance of defects and quality problems in all kinds of production, services, management and other kinds of business activity. Recently, Six Sigma has become more and more popular (due to many successful implementations) in the USA and around the world. Six Sigma recommends the following stages of data mining (so called DMAIC):

$$Define \rightarrow Measure \rightarrow Analyze \rightarrow Improve \rightarrow Control$$

Different methodology, similar to a certain degree to Six Sigma, is SEMMA, proposed by the SAS Institute. SEMMA is focused on technical side of data mining projects. This is as follows:

$$Sample \rightarrow Explore \rightarrow Modify \rightarrow Model \rightarrow Assess$$

The above methodologies are focused on using the data mining in an organization. They try to answer the questions: how to convert data into knowledge, how to involve proper persons (company owners, managers) in data mining, how to use and publish knowledge in a form that it could be easily used in the decision process.

Ranking functions are used to evaluate if the model fits the data set. As a ranking function there may be used one of these: likelihood, total square error (the sum of square differences between the real and predicted values), classification error factor, etc.

Maximum likelihood method [1, 2] is a general method for population parameters estimation with the help of values which maximizes the likelihood of a sample (L). Likelihood L consists of n observations $x_1, x_2, \ldots, x_n$. L is a function of combined probability $p(x_1, x_2, \ldots, x_n)$, where $x_1, x_2, \ldots, x_n$ are random discrete variables. If all of the variables are continuous, likelihood L consisting of n observations is a density function of total probability $f(x_1, x_2, \ldots, x_n)$. If L is a function of $\theta_1, \theta_2, \ldots, \theta_k$ and $L(\theta)$ is differentiable, maximum likelihood is the maximum of $L(\theta)$).

3 Mobile robots

Mobile robots (mobots) are often used for exploring areas dangerous or not accessible for human being, and also for monitoring, watching and guarding [3–8]. When such an area changes dynamically (is devastated by uncontrolled forces, for example) then a mobot must be remotely controlled by a human operator.

The operator should detect all objects in the area fast and precisely. The question arises how to select such a person from a group of candidates applying for this position. Due to the fact, that guarded properties and equipment used are valuable, guards should be possibly the best. To train and rank the operators a simulator was developed. The simulator is similar to a flight simulator. Mobot controlled by an operator moves

in virtual reality (in a simulated room) and takes photos. The simulator uses computer generated graphics instead of real images that would be taken by the mobot in a real room. The task for the operator is to find all scene changes in limited time [3–7].

A simulation experiment, in which the candidates are trained, is used as a source of data. The candidates are trained in virtual reality. After the training, the best candidates are chosen. The selection procedure consists of the following steps [7]:

- Basic operator activities are measured (the number of changes, errors, moves, photos, etc.).
- After the training, unreliable results and candidates are 'rejected' (too many errors, too many moves and photos, cheating, etc.).
- From the rest of the candidates ('accepted'), those with the largest number of discoveries are 'chosen'.

Predictive model, built on the basis of available data, allows for choosing the best candidates.

4 Problem statement

The data coming from the experiment should be worked out to get the best candidates. To this end, expert knowledge is used. The data are supplemented with a new 'decision' attribute coming from the expert. Now, the most desired result of data mining for the experiment is the predictive data model allowing us to classify candidates and estimate the decision value.

Data supplemented with expert decision was used to build a few predictive data models. Using different algorithms, we got different models, so the problem is to choose the best, the most reliable and the most universal one.

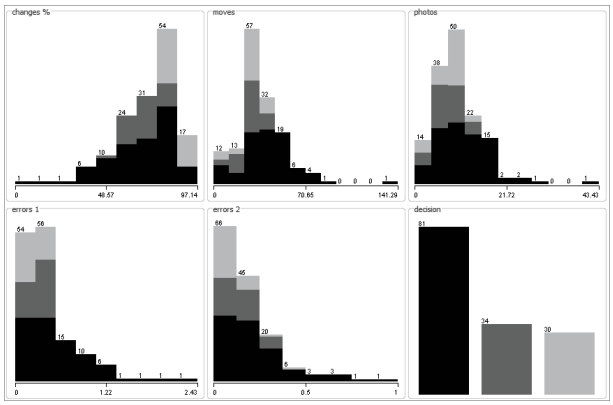

Fig. 2. Mobot operator data characterization (from Weka).

In Fig. 2 characterization of data coming from the experiment is shown. Black colour indicates the 'rejected' candidates, mid gray - 'accepted' and light gray - 'chosen'. Each

of the candidates has associated attributes like: a number of detected changes (%), a number of moves, a number of photos taken and a number of errors made. There are two kinds of errors - invalid discoveries (*errors 1*) and invalid moves (*errors 2*). To build and train a model, another attribute is added, a so called 'decision'. The 'decision' is a categorical variable, with the exception for linear regression, where it becomes a numerical (quantitative) three-state variable.

The whole experimental data consist of records for 145 candidates, where 67 records form a 'train' set, and the remaining 78 records form a 'test' set. The 'train' set was used for building the model, verified later with the 'test' set. Characteristics for the 'train' set and the 'test' set are very similar to the set characteristics (Fig. 2).

There are many software applications for data exploration, as Weka, Statistica, etc. To this end, a predictive model was built using the Weka [9] open source software. Four methods (Decision Tree, Decision Table, Naive Bayes Classifier, Linear Regression Model) were chosen for building the model. The results from Weka are the following (the letters denote the decision: (r)ejected, (a)ccepted and (c)hosen):

- **Decision tree**

  A model with the local segmented structure, the tree consists of decisions (tree nodes) and categories (tree leaves) [11]:

```
changes% <= 77.14
|   errors1 <= 0.43
|   |   moves <= 39.57
|   |   |   changes% <= 60
|   |   |   |   photos <= 6.71: r
|   |   |   |   photos > 6.71: a
|   |   |   changes% > 60: a
|   |   moves > 39.57: r
|   errors1 > 0.43: r
changes% > 77.14
|   photos <= 13.57: c
|   photos > 13.57: r
```

- **Decision table**

  A model consists of a set of 'if-then' rules, created with the help of rough set theory, 19 decision rules were generated for the data [12]:

```
changes%      moves          photos           errors1    errors2    decision
(78.57-inf)   (40.43-inf)    (13.715-inf)     (0.5-inf)  (0.36-inf)  r
...
(-inf-78.57]  (40.43-inf)    (-inf-13.715]    (-inf-0.5] (-inf-0.36] r
(78.57-inf)   (-inf-40.43]   (-inf-13.715]    (-inf-0.5] (-inf-0.36] c
(-inf-78.57]  (-inf-40.43]   (-inf-13.715]    (-inf-0.5] (-inf-0.36] a
```

- **Naive Bayes classifier**

  A simple probabilistic classifier based on applying the Bayes' theorem with strong (naive) independence assumptions [10]:

```
Naive Bayes Classifier
----------------------

Class a: Prior probability = 0.2
changes%: Normal Distribution. Mean = 67.6923 StandardDev =
    8.5663
WeightSum = 13 Precision = 4.7058823529411775
moves: Normal Distribution. Mean = 31.0885 StandardDev = 6.175
WeightSum = 13 Precision = 2.283333333333333
photos: ...
errors1: ...
errors2: ...

Class r: Prior probability = 0.57
changes%: Normal Distribution. ...
moves: Normal Distribution. ...
...

Class c: Prior probability = 0.23
...
```

- **Linear regression**

  The result is a function, computing value of the dependent variable, using the values of the independent variables [1][2]:

```
Linear Regression Model
-----------------------

decision = 0.0252*changes% - 0.0716*photos - 0.5379*errors -
    0.1644
```

  The analysis of the models obtained from Weka leads to the following conclusions:

  - for each of the models a set of required attributes was properly chosen;
  - the models differ considerably;
  - neither of the models is accurate.

  Therefore, all of the models should be verified.

5 Expectation-Maximization (EM) Algorithm

The models are verified with the help of the EM (expectation-maximization) algorithm, which was designed for solving grouping (clustering) problems with missing data [1,2]. Actually, the selection is a grouping problem because the candidates

are assigned to the 'chosen', 'accepted' or 'rejected' subgroups. For such a case the missing data is the category (the subgroup) for which a candidate should be assigned, on the basis of his activity measured during the experiment. When the EM algorithm would be taken for solving this problem, we could get a model worked out from our experimental data.

Table 1. Comparison of the predictive data models generated with Weka, built using the 'train' set and tested with the 'test' set (5 attributes were available).

	Decision Tree	Decision Table	Naive Bayes	Linear Regression
Attributes used	4	5	5	3
Time taken to build model	0.02	0.02	0.02	0.02
Correctly Classified Instances (%)	87.1795	88.4615	79.4872	0.6266 (Correlation coefficient)
Mean absolute error	0.0975	0.0849	0.1616	0.5205
Relative absolute error (%)	0.5205	21.411	40.7647	74.0148

5.1 EM background

The EM algorithm is used for finding the maximum likelihood estimates for parameters in a model. The model is built on the basis of unobserved variables (missing data). EM consists of two repeated steps:

- Expectation (E) step, where expectation of the likelihood is computed. The missing variables, as expected, are used in the computation.
- Maximization (M) step, where the expected likelihood of the parameters, found in E step, is maximized.

Parameters found in the M step are then used to perform another E step and so on. This terminates an optimum (which may be, unfortunately, a local optimum). In practice, the EM algorithm stops if values and parameters of missing variables do not change significantly between the two consecutive EM iterations.

Let $D = x(1), \ldots, x(n)$ denote a set of n observed data vectors, $H = z(1), \ldots, z(z)$ represents a set of missing data ($z(i)$ applies to $x(i)$). Logarithmic likelihood of the observed data may be expressed as [1]:

$$l(\theta) = \log p(D|\theta) = \log \sum_{H} p(D, H|\theta). \tag{1}$$

The observed $l(\theta)$ is described by a probabilistic model $p(D, H/\theta)$, where θ are unknown model parameters In this case, the values of both θ and H are not known. If $Q(H)$ is a probability distribution of missing data H, the logarithmic likelihood may be stated as:

$$l(\theta) = F(Q, \theta). \tag{2}$$

The function $F(Q, \theta)$ is the lower bound of likelihood function $l(\theta)$, which should be maximized. The EM algorithm maximizes F as the function of Q with the fixed θ(E step), and F as a function of θ with the fixed distribution $Q = p(H)$(M step).

In the case of computer implementation of the EM algorithm (as in Weka [9]), different distribution of the missing variable may be used for getting the best results, such as Gaussian, normal or Poisson distribution, or mixture of many distributions.

6 EM results for the mobot operator data

The experimental data evaluated with the help of Weka and its EM clusterer [9]. The 'train' and the 'test' data sets were concatenated to form a single set. The EM algorithm was run many times with different parameters, getting very similar results. The results for the set of data with all attributes, are given below:

```
Clustered Instances
0       22 ( 15%)
1       60 ( 41%)
2       25 ( 17%)
3        3 (  2%)
4       35 ( 24%)
Log likelihood: -10.38553
Class attribute: decision
Classes to Clusters:
        0  1  2  3  4  <-- assigned to cluster
        9 17 21  3 31 | r
        8 20  4  0  2 | a
        5 23  0  0  2 | c
Cluster 0 <-- a
Cluster 1 <-- c
Cluster 2 <-- No class
Cluster 3 <-- No class
Cluster 4 <-- r
Incorrectly clustered instances :       83.0      57.2414 %
```

Surprisingly, the decision values seem to be of no use, as the resulting clusters are not reflecting the decision attribute. One cluster for one decision was expected in the best case. But here no one but all clusters cover the (r)ejected decision. Moreover, over 57% of the data was classified incorrectly, so the EM algorithm seems to be inappropriate for our data or reversely.

7 Expectation-Maximization (EM) Algorithm

To verify the predictive data models, the data set was supplemented with decisions made by each of the models. The clustering results were compared with each data model. The comparison is similar to the expert knowledge verification. For all of the models the clustering does not reflect model decisions (Table 2).

Table 2. Results from the models compared to the EM clustering.

Predictive model	Incorrectly clustered instances	
	number of	%
Expert knowledge	83	57.2414
Decision Tree	81	55.8621
Decision Table	74	51.0345
Naive Bayes	79	54.4828
Linear Regression	83	57.2414

As a few of the predictive models do not use all attributes for computing the result, some attributes were also excluded from the data set for the EM algorithm. Further analysis led to the results shown in Table 3.

Table 3. EM algorithm for the data with excluded attribute results.

Excluded attribute(s)	Incorrectly clustered instances	
	number of	%
changes	87	60.0000
moves	88	60.6897
photos	75	51.7241
errors1	89	61.3793
errors2	84	57.9310
moves & errors1	80	55.1724
moves & errors2	76	52.4138

The results from the EM algorithm are much less dependable than from any predictive model. The question arises why the results are not reliable. The answer may be one of the following:

- Mobot experiment data is not 'clusterable', because of the distribution of the attributes and the threshold decisions forming the knowledge of the expert?
- A data model is not reliable if one (or many) of the variables confuses the result?
- The distributions of values are not normal. Weka claims that they are normal, but not all of the attributes are used in predictive data models?

8 EM results for selected data

To verify if the EM algorithm may give good results for different data, verification was repeated for the selected set of data - 65 from 145 total records (45%). The records were chosen to form 'clusters', and the EM clustering algorithm was repeated. The results are good as expected, data forms three distinct clusters, reflecting the decision attribute (expert knowledge based decision):

```
Clustered Instances
0       21 ( 32%)
1       24 ( 37%)
2       20 ( 31%)
Log likelihood: -8.8971
Class attribute: decision
Classes to Clusters:
0  1  2  <-- assigned to cluster
0 23  0 | a
21  1  1 | c
0  0 19 | r
Cluster 0 <-- c
Cluster 1 <-- a
Cluster 2 <-- r
Incorrectly clustered instances :       2.0        3.0769 %
```

The clusters are almost 100% accurate in such a case. Moreover, it would be easy to get 100% accuracy by removing these two confusing records from the selected data set.

9 Conclusions

The EM algorithm is a well known and reliable method for data clustering, especially for data similar to these of our experiment (such as medical, sociological data, etc.). Choosing the EM algorithm to verify our predictive data models (and the expert knowledge itself) was a new but very promising idea.

However, the EM algorithm gives good results only if records with the same decision form a cluster, which was proved for the selected set of data coming from the

experiment. Unfortunately, the whole data do not meet such a requirement. This means that the results from the EM algorithm cannot be used for any verification. All predictive data models get better classification results than the EM (compare Tables 1, 2 and 3).

The experiment studied in the paper, that is a model of training mobot operators, is a new problem for data mining. For medical purposes, the data models and their parameters are well known and thoroughly verified. So for such sort of data the EM algorithm gives good results.

References

[1] Hand D., Mannila H., Smyth P., Eksploracja danych (WNT, Warszawa, 2005): 207–252.

[2] StatSoft Electronic Textbook, http://www.statsoft.com/textbook/stathome.html (accessed 2009-02-17).

[3] Sapiecha K., Łukawska B., Paduch P., Experimental Data Driven Robot for Pattern Classification, Annales UMCS Informatica AI 3 (2005): 263–271.

[4] Łukawska B., Paduch P., Sapiecha K., An application of virtual reality for training and ranking operators of mobile robot, Annales UMCS Informatica AI 5 (2006): 393–399.

[5] Sapiecha K., Bedla M., Łukawska B., Paduch P., Computer-based system for training and selecting mobile robot operators – evolving software tools, Annales UMCS Informatica AI 7 (2007): 107–115.

[6] Bedla M., Łukawska B., Sapiecha K., Software architecture for a system of remote training mobot operator, Advanced Computer Systems and Networks: Design and Application. Proc. of the third International Conference ACSN-2007 (ACSN, Lwów, 2007): 104–106.

[7] Sapiecha K., Łukawska B., Bedla M., Computer-based system for training and ranking mobot operators – selection procedure, Annales UMCS Informatica AI 8 (2008): 107–118.

[8] Polski Wortal Robotyki, http://www.asimo.pl (accessed 2009-02-17).

[9] The University of Waikato, WEKA, http://www.cs.waikato.ac.nz/ml/weka (accessed 2009-02-17).

[10] John G. H., Langley J., Langley P., Estimating Continuous Distributions in Bayesian Classifiers, Proc. of the Eleventh Conference on Uncertainty in Artificial Intelligence (Morgan Kaufmann, San Mateo, 1995): 338–345.

[11] Quinlan R., C4.5: Programs for Machine Learning (Morgan Kaufmann Publishers, San Mateo, 1993): 1–26.

[12] Kohavi R., The Power of Decision Tables, Proc. European Conference on Machine Learning (1995): 174–189.