The effect of Missing Values using Genetic Programming on evolvable diagnosis.

James Cunha Werner (jamwer2000@hotmail.com)

Tatiana Kalganova (Tatiana.Kalganova@brunel.ac.uk)

Department of Electronic & Computer Engineering, Brunel University

Uxbridge, Middlesex, UB8 3PH

Abstract. Medical databases usually contain missing values due the policy of reducing stress and harm to the patient. In practice missing values has been a problem mainly due to the necessity to evaluate mathematical equations obtained by genetic programming. The solution to this problem is to use fill in methods to estimate the missing values. This paper analyses three fill in methods: (1) attribute means, (2) conditional means, and (3) random number generation. The methods are evaluated using sensitivity, specificity, and entropy to explain the exchange in knowledge of the results. The results are illustrated based on the breast cancer database. Conditional means produced the best fill in experimental results.

Keywords: genetic programming, missing values, disease diagnostic, fill in methods; entropy.

1. Introduction.

The origin of missing values in databases is not totally random. There is always a physical effect behind its occurrence that justifies its pattern.

A problem in an external sensor, such as broken cable, can originate records with missing values. This situation generates a time interval without values, while the problem is not fixed. The information behind this event is the sensor acquisition system is out of order.

The industrial laboratory many times uses composed samples: if one sample per hour is acquired, to evaluate the average value over 12 hours, the same amount from each sample can be composed in one sample and only one analysis is performed resulting in the average value, instead to do 12 analysis and then evaluate the average. When the average value leaves the expected range, then each analysis must be performed to establish if there is tendency, if the problem is in the sample acquisition, etc. In this case, a set of new information, more detailed during this time interval (each hour), will introduce missing values in the remaining of the dataset that were acquired each 12 hours.

Other source of missing values is the use of one instrument (such as Near Infrared Analyser) in different places in the process to monitor some special condition. In this case, when some situation occurs, there are values in the dataset. Missing values in this case mean that the situation is not occurring, because the system is not active. Medical data are in this type of missing values source. It is very harmful to the patient the samples collection process, so the strict necessary to support the diagnostic is done, and the experience of the physician guides the data acquisition.

Missing values can be created when the storage/recovery process is not totally controlled. In this case, it is difficult to establish the source of error because this type of problem is intermittent, and can present any type of patterns: absence in intervals, or randomly distributed.

Hence, there is a degree of information in missing values pattern. The absence of information is information too [1]. However, when working with mathematical models, where numbers must be replaced in variables position, missing values is a big problem.

The practical fulfilment of mathematical model evaluation using databases with missing values is implemented by fill in methods. It is a difficult task because in this process data bias or dispersion can be introduced. Several fill in techniques have been developed based in statistical concepts.

The ideal case for analysis is of database is when the data are complete. If a record has missing data for any one variable used in a particular analysis, omit that entire record from the analysis. This approach is implemented as the default method of handling incomplete data by many statistical procedures in commonly-used statistical software packages. The model obtained with these data is based in real data only, without any fill in technique which could introduce noise. The disadvantage is the amount of information lost when omits the entire record.

Cold deck imputation replaces missing values by a constant value from an external source, such as a value from a previous realization of the same survey. Some authors replace by the “normal” value [18]. The disadvantage of the method is that it does not specifies how to define normal values without introduce noise.

There are a number of fill in methods such as statistical values, Buck’s method [3], Hot deck imputation ([4] to [7]), Regression methods, Raw maximum likelihood methods [1], Expectation maximization (EM) [1], and Multiple imputation [1] that use the statistical properties of the data to fill the missing values. The disadvantage of these methods is the introduction of noise. This leads sometimes to misinterpreting the data.

Although the existing methods are capable to fill in the missing values no methods have been found that show how the information is affected by these methods. This can be crucial in medical databases. In order to overcame this problem we propose to introduce missing values and study 3 different fill in methods: (1) attribute means, (2) conditional means, and (3) random number generation.

Our research shows how to evaluate the introduction and removal of information from the database and its effect in the disease model by the different fill in methods through the use of sensitivity and specificity (ROC) [13]. We introduce entropy to evaluate the information in the database, and how it affect the predictions obtained by the model.

Genetic programming is applied to obtain the discriminate function of the disease using each filled database.

Section 2 describes the general method adopted in this paper. Section 3 introduces entropy and information, and section 4 the statistical methods and its entropic explanation. The used fill in methods are described in section 5. The discriminate function used for diagnosis (section 6) was obtained by genetic programming (section 7). The experiment uses breast cancer database (section 8). The paper concludes with a discussion of the effect of the fill in process in the amount of information available in the database.

2. The methodology

Fig. 1 shows the general methodology adopted in this paper.

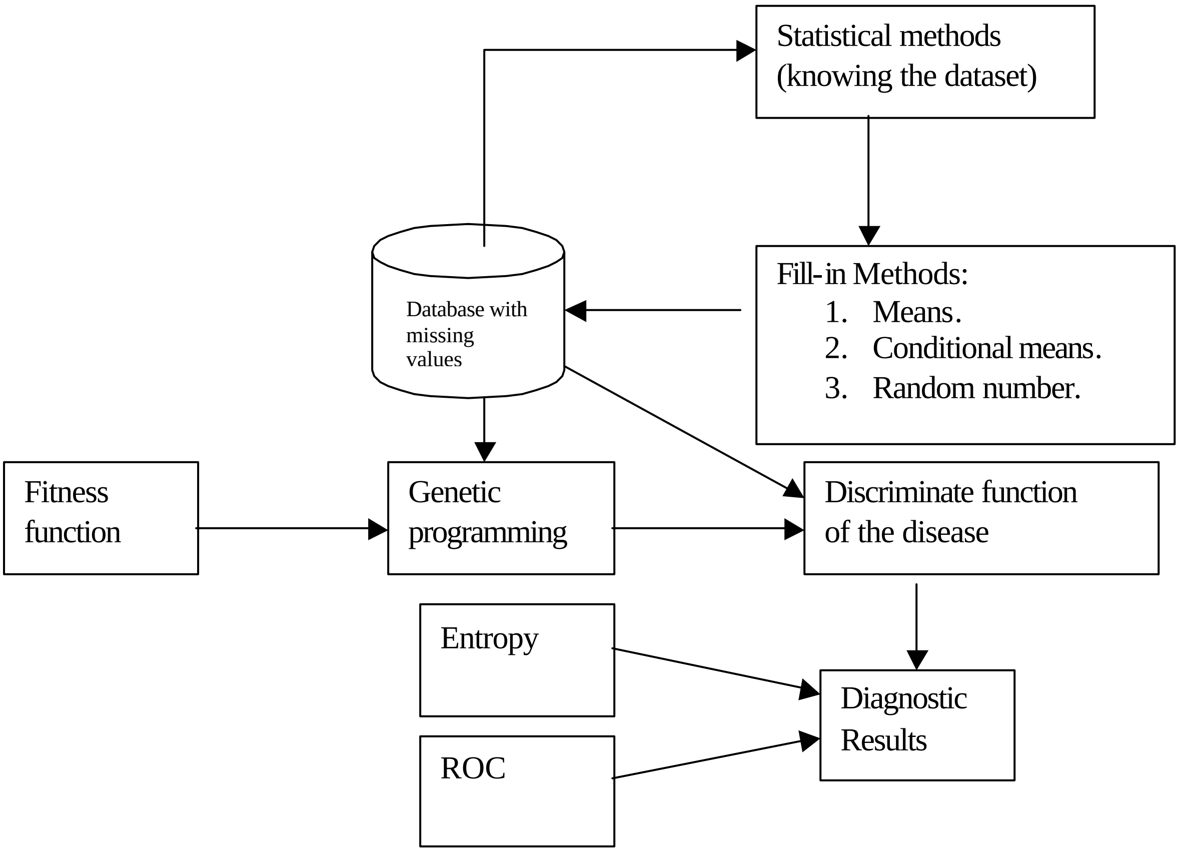

Fig. 1 The methodology adopted to study the effect of missing values fill in methods.

The attributes of the database is marked as missing by chance. The resulting database is analysed with statistical tools (such as means, standard deviations and correlation) to obtain the parameters for each fill in method. Genetic programming obtains the discriminate function. This function describes the diagnostic of the different patient data maximising the fitness function. The accuracy of the model applied to the database is studied using entropy and ROC methods. In the following sections we will describe in details the introduced approach.

3. Entropy and information

This section introduces the concept of entropy and its relation with information. The imputation and measure quality affect the value of entropy, and this is a crucial point to analyse the fill in methods used.
The Shannon information content of an outcome *x* is defined to be [2]

$$h(x) = \log_2 \frac{1}{P(x)} \qquad \textbf{(1)}$$

where *P(x)* is the probability distribution of *x* and *h(x)* is measured in bits. The entropy of an ensemble X is defined to be the average Shannon information content of an outcome:

$$H(X) \equiv \sum_{x \in A_x} P(x) \log \frac{1}{P(x)} \qquad \textbf{(2)}$$

Table 1 presents four different dispersion values from a hypothetic dataset. The first is measure variable always value 105, e.g., the probability distribution is 1.0 for 105 and 0 for any other value. The information value is obtained using equation 2 for the values of measure range with some probability of occurrence. The others datasets contains different values of dispersion. The effect of measures dispersion is evaluated in the "ensemble entropy" row using different dispersions (dataset 2 to 4). When the value is known with 100% certainty (dataset 1), the information new measures is zero. There is not new information possible to be acquired, and the system is completely known.

However, when disperse around a mean value (dataset 2 and 3), the ensemble entropy increases, meaning that information still missing, and new measures can improve the knowledge of the system, or the measure value is not predictable with 100% of certainty.

The format of the dispersion (if flat like dataset 4 or bell shaped like datasets 1 to 3) affects the entropy value too. It is intuitive that measures in dataset 3 are nearest value 105 than dataset 4, with can be any value between 95 and 115 with the same probability.

Table I Effect of dispersion in ensemble entropy for four hypothetic datasets. The information value is obtained with equation 1 and the ensemble entropy with equation 2.

	Dataset 1		**Dataset 2**		**Dataset 3**		**Dataset 4**	
Measure	**P(x)**	**Information**	**P(x)**	**Information**	**P(x)**	**Information**	**P(x)**	**Information**
95					0.10	0.33	0.2	0.46
100			0.3	0.520	0.22	0.48	0.2	0.46
105	1.0	0	0.4	0.528	0.35	0.53	0.2	0.46
110			0.3	0.520	0.22	0.48	0.2	0.46
115					0.11	0.35	0.2	0.46
Ensemble Entropy	0		1.56		2.17		2.32	

Concluding, ensemble entropy can measure the information available in the dataset, and how good is the method of deal with data of the database, such as fill in methods or model prediction.

4. *The statistical method: Knowing the dataset.*

Probabilities refer to frequencies of outcomes in random experiments. A set of data can be studied using average, standard deviation, covariance and correlation. The rules of probability ensure that if two people make the same assumptions and receive the same data them they will draw identical conclusions [2].

When the average is calculated, it is necessary to select datasets with the same characteristics. For example, there is 50% of chance to born a male offspring. This means that one in two children is male and one is female. However, physical condition (such as ionic acidity, time of the period, etc) can affect the predisposition to male or female offspring.

The use of statistics methods to fill in missing values can interfere in the entropy of the dataset. If an average value is introduced, the dispersion will be centred in the average value, changing the shape of the distribution. This can be seen, for example, in table 1. If

the dataset 3 is fill in with the average (105), the probability distribution will change for something like dataset 2 because more points will be in value 105 than originally.
The conclusion is that statistical fill in methods can introduce a change in the information due the bias it introduce in the data, and entropy is able to detect it.

5. Fill in methods used.

In order to obtain the missing values the following fill in methods were adopted:

- **Imputing unconditional means.** A particularly simple form of imputation is to estimate missing values by the mean of the recorded values. The average of the observed and imputed values is then clearly the estimate from available case analysis.
  A natural consequence is an underestimation of variance because the missing values are imputed at the centre of the distribution.
  The average of events sometimes can introduce erroneous values. If the average of Clump Thickness from Wisconsin Diagnostic Breast Cancer repository is evaluated, the result is 4.44. The average for benign tumour is 2.96 and malign is 7.22. The creation of a new cluster is meaningless.
- **Imputing conditional means.** For each different attribute, the mean for each possible condition (the diagnostic in medical data case) is obtained and replaces the missing values of the attribute. The disadvantage of this method is the previous knowledge of the condition for each record.
- **Random filling in missing values (random number).** If we ignore sampling variability of the estimates of μ and Σ based on complete cases, then the conditional means are the best point estimates of the missing values in the sense of minimising the expected squared error. The marginal distributions of the completed data are distorted by mean imputation.

These considerations suggest an alternative strategy where imputations are selected randomly from a distribution of plausible values, rather than from the centre of the distribution. To apply this method, we used the *randlib.c* software available in the Internet [17].

6. Discriminate function of the disease

In order to obtain precocious diagnostic, Werner & Kalganova [16] introduced a new approach in disease modelling. Given a database one model is evolved (called discriminate function). This model describes mathematically the diagnostic of the disease. The dynamic of the disease is taken into account in the model.
Discriminate function maps the original multi dimensional space (the clinical database) in a one-dimensional real number image (the disease diagnostic). The output space has a threshold with separate diagnostic classes. In our approach the origin was adopted as a threshold: positive values mean an ill patient and negative values mean a healthy patient.
A multiplicative weight (termed punishment) is introduced to give more priority to false negatives (the model result is patient healthy, but the true is the patient has the disease and the treatment will be delayed). It guarantees minimal false negatives, which costs accuracy in true negative values. This is a safe condition for the patient.
The advantage of the method is that the information is available in the model. It is possible to detect patterns and understand the disease mechanism through the analysis of the model. Genetic programming is used to obtain the discriminate function of the disease.
The problem of missing values introduces the necessity of a estimated numerical value to be replaced in the equation model. Missing values are very disruptive.

7. Genetic programming applied in evolvable discriminate function for diagnosis.

Genetic Programming (GP) is an optimization algorithm which mimics the evolution and improvement of life through reproduction. Each individual contributes with its own genetic information to the building of new ones (offspring) adapted to the environment with higher chances of surviving. This is the basis of genetic algorithms and programming ([8] - [11]).
The software we have developed is an adaptation of LilGP [12], where GP is structured in a pre-compiled library, with other artificial intelligence procedures, such as neural networks, fuzzy logic, adaptive algorithms, etc. Outputs are written in Excel XLS format direct from the program, to generate an accessible and functional Human-Computer Interface (HCI).

Chromosome representation. The chromosome represents the model of the problem solution using trees. A tree is a model representation that contains nodes and leaves.
Nodes are mathematical operators. We have used multiplication, addition, subtraction, and division. Leaves are terminals (the attributes of the dataset and numbers). The discriminate function in a GP context is a tree using operators (or so called Functions) and leaves (or so called Terminals). Let us consider the following discriminate function:

$$X_1+3.14 \cdot X_2+5.3 / X_3$$

In the tree representation it can be rewritten as following:

$$(+ X_1 (+ (\cdot\ 3.14\ X_2) (/ 5.3\ X_3)))$$

where X_1, X_2, and X_3 are the attributes of the clinical data, and multiplication($\cdot$), addition(+), subtraction (-), and division(/) are the operators. Replacing the values of the clinical data in the equation results in a number which should be positive (the patient is ill) or negative (the patient is healthy). This model is the representation of one discriminate function, and the effect of each clinical data in the diagnostic.

Genetic operators. Trees are manipulated through genetic operators. The crossover operator points a tree branch and exchanges it with another branch and obtains new trees. The mutation operator changes the branch with a random new branch. The length of the chromosome is variable.

Fitness function. Fitness function defines the quality of chromosome as a solution to the problem. It is a numerical positive value. The dataset is divided in two parts: one is for training and the second one is for testing (validation). The training dataset is used to obtain the model and the validation dataset is used to measure the accuracy of the model with data that were not used in training.
The fitness function evaluates how good the diagnostic model coded in chromosome is, over all training dataset using Receiver Operating Characteristics (ROC) [13].
The difference between the real value and the predicted one can be sorted in several classes, and the different classes of results can be represented in one table such as Table 2.

Table 2 The different classification of prediction.

		Prediction	
		Benign	**Malign**
Real	**Benign**	True negative	False positive
	Malign	False Negative	True positive

This representation will be used in the experimental results.
ROC criterion value is sliding in the output projection and the number of true negative (N$_{TN}$), true positive (N$_{TP}$), false negative (N$_{FN}$), and false positive (N$_{FP}$):

$$\boldsymbol{a} = \frac{N_{TP}}{N_{TP} + N_{FN}} \quad \boldsymbol{b} = \frac{N_{TN}}{N_{TN} + N_{FP}} \tag{3}$$

where á is the *Sensitivity*, and â is the *Specificity*. Sensitivity α is the probability that a test result will be positive when the disease is present (true positive rate, expressed as a percentage). *Specificity* $\boldsymbol{b}$ is the probability that a test result will be negative when the disease is not present (true negative rate, expressed as a percentage). These indexes show the performance of the model over the data, but it do not evaluate the amount of information that you exchange with different approaches. Entropy fills this gap.
The fitness function *F* used in the disease diagnostic defines the accuracy of the model, with a weight over false negatives predictions:

$$F = \frac{N_{ok}}{N_{ok} + N_{FP} + \boldsymbol{s} * N_{FN}} \tag{4}$$

where *ó* is the overprice for false negative (high risk condition), or punishment weight, N_{ok} it is the number of correct forecast, N_{FP} it is the number of false positives and N_{FN} it is the number of false negatives.
The consequence of this fitness function is the fail safe condition for the patient through the reduction of false negative diagnostic.

8. Experimental results.

The purpose of these experiments is to evaluate the effect of different fill in methods in the discriminate function, and evaluate the efficacy of ROC and ensemble entropy in the measurement of information.
Two experiments using the breast cancer database were carried on. Experiment 1 considers the specific features of the training stage and Experiment 2 identifies how testing stage effects missing values. The Wisconsin Diagnostic Breast Cancer ([14], [15]) contains 679 events (236 ill and 443 healthy records) without any missing values. The dataset contains the following attributes: Clump Thickness, Uniformity of Cell Size, Uniformity of Cell Shape, Marginal Adhesion, Single Epithelial Cell Size, Bare Nuclei, Bland Chromatin, Normal Nucleoli, Mitoses, Class (2 benign and 4 malignant). Each attribute is an integer between 1 and 10.
The database was initially filled with missing values. Several threshold value are defined (10%, 30% and 50%), and for each attribute value a random number is generated. If the generated random value is less than the threshold, a missing value is created.
Three methods where used to fill in the missing values: (1) attribute means, (2) conditional means, and (3) random number generation.
The filled dataset is used by genetic programming to obtain the discriminate function.
Two experiments are carried on: the first one analyses the effect of missing values in the training stage, and the second one analyses the effect in the test stage.
The parameters used in genetic programming are the same for both experiments. This has been done in order to do not introduce any perturbation in the entropy: probability of crossover is 60%, the probability of mutation is 20%, and population size is 200 individuals. We adopt a high value of the mutation probability to spread the population over all solution space.

Experiment 1: The study of fill in methods in the training stage.

The purpose of this experiment is to study the effect of the different fill in methods in the training process.

The complete dataset was first used to obtain the discriminate function (Table 3) and study the effect of *normalisation*. The representation in Table 3 is the same as in Table 2 for each different condition. The normalisation process determines the minimum and maximum value of the attributes, and converts this range linearly to the interval [0,1]. It is important to introduce normalisation, because some attributes range should be larger than others, introducing some type of bias. The normalisation solves the problem of range for random numbers in GP, that can be defined between [0,1].

The normalisation has been tested in the complete database and do not affect the solution.

Table 3. Experiment 1: Normalisation effect in training with complete dataset. α is the sensitivity and β is the specificity obtained by equation 3. H is the ensemble entropy obtained by equation 2.

Original data				**Normalised data**			
		Predict				**Predict**	
		Benign	**Malign**			**Benign**	**Malign**
Real	**Benign**	429	14	**Real**	**Benign**	411	32
	Malign	2	234		**Malign**	0	236
		α=0.9915; β=0.9683; H=1.08				α=1.000; β=0.9277; H=1.17	

The problem of missing values deals with generating a percentage of missing values in the original complete database (table 4). The amount of missing values is adopted for all attributes, except for the class attribute. Three fill in methods were studied.

The first replaces by the attribute *average*, whatever the diagnostic. There is not a degradation of the discriminate function, because the false negative still in the same value, but there is a clear increase in the false positive.

The *conditional means* introduces a very interesting behaviour. The number of false positives and negatives decrease when the amount of missing values increases. The explanation lies in the better definition of the boundary between ill/healthy cases.

When the missing value is replaced by the average of the diagnostic case, the points are concentrated in the region, and the dispersion effect introduced by noise and experimental measure is reduced, reducing the entropy. In this case, there is an increase of information in the data of the system.

The last method introduces a *random number* conditioned by variance and covariance of each attribute, and shows the increase of uncertainness of the boundaries between ill/healthy cases. In this case there is a reduction in the information in the data of the system.

The conclusion of this experiment is that the entropy is able to evaluate the amount of information changed in the fill in method. The conditional means method is the better solution for fill in missing values, and the sensitivity, specificity and entropy shows the same result.

Table 4. Experiment 1: Missing values effect in training with complete dataset. α is the sensitivity and β is the specificity obtained by equation 3. H is the ensemble entropy obtained by equation 2.

Fill in method			Missing Values Threshold					
			10%		**30%**		**50%**	
			Predict		**Predict**		**Predict**	
			Benign	**Malign**	**Benign**	**Malign**	**Benign**	**Malign**
1.Means	**Real**	**Benign**	426	17	389	54	364	79
		Malign	2	234	3	233	2	234
			α=0.9915; β=0.9616; H=1.10		α=0.9872;β=0.8781; H=1.31		α=0.9915; β=0.8216; H=1.39	
2.Conditional means	**Real**	**Benign**	428	15	436	7	440	3
		Malign	0	236	0	236	0	236
			α=1.00; β=0.9616; **H=1.07**		α=1.00; β=0.9841; **H=1.00**		α=1.00; β=0.9932; **H=0.96**	
3.Random number	**Real**	**Benign**	395	48	329	114	277	166
		Malign	2	234	2	234	12	224
			α=0.9915; β=0.8916; H=1.27		α=0.9915;β=0.7426; H=1.49		α=0.9491; β=0.6252; H=1.65	

Experiment 2: The study of fill in methods in the test of non trained data.

The purpose of this experiment is to study the effect of the fill in method in non trained data. The evaluation of the model accuracy to new data was studied by dividing the database in 10 parts. Cyclically, each time 9 parts were used to obtain the discriminate function and the remaining part (not used in training) were used to test the model. After 10 changes, the final result is the test result of all database. The algorithm can be described as:

```
Stage 1: divide the database into $n$ parts.
Stage 2:  for $j$=1 to $n$
            training genetic programming with all pieces except piece $j$.
            evaluate the solution obtained by GP with piece $j$, and write the solution into file
Stage 3: use file with the 10 solution tests to evaluate the sensitivity, specificity, and entropy of the solution obtained by genetic programming
```

This approach is very interesting because at end there are 10 discriminate functions, one for each piece of database, and new data can be evaluated by the 10 models and the probability of ill or health can be obtained.

Table 5 shows the effect of *normalisation* in test results. There is no effect of the normalisation, as shown in Table 3.

Table 5. Experiment 2: Normalisation effect in test with rotate test dataset. α is the sensitivity and β is the specificity obtained by equation 3. H is the ensemble entropy obtained by equation 2.

Original data				Normalised data			
		Predict				**Predict**	
		Benign	**Malign**			**Benign**	**Malign**
Real	**Benign**	411	32	Real	**Benign**	414	29
	Malign	4	232		**Malign**	4	232
		α=0.9830; β=0.9277;H=1.21				α=0.9830; β=9345; H=1.20	

Table 6. Experiment 2: Missing values effect in test with rotate test dataset. α is the sensitivity and β is the specificity obtained by equation 3. H is the ensemble entropy obtained by equation 2.

Fill in method			**Missing Values**					
			10%		**30%**		**50%**	
			Predict		**Predict**		**Predict**	
			Benign	**Malign**	**Benign**	**Malign**	**Benign**	**Malign**
1.Means	**Real**	**Benign**	418	25	408	35	369	74
		Malign	7	229	8	228	12	224
			α=0.9703; β=0.9435; H=1.20		α=0.9661; β=0.9209; H=1.26		α=0.9491; β=0.8329; H=1.45	
2.Conditional means	**Real**	**Benign**	424	19	436	7	442	1
		Malign	1	235	1	235	0	236
			α=0.9957; β=0.9571; H=1.11		α=0.9957; β=0.9841; H=1.02		α=1.0; β=0.9977; H=0.94	
3.Random number	**Real**	**Benign**	395	48	358	85	241	202
		Malign	3	233	8	228	26	210
			α=0.9872; β=0.8916; H=1.28		α=0.9661; β=0.8081; H=1.45		α=0.9576; β=0.5440 H=1.75	

The same methods described in Experiment 1 are repeated in Experiment 2.

The different behaviour of fill in methods shown in Table 6 shows the same behaviour as in Table 4. Again, the conditional means had the better performance, but in this case for non trained data. Note that the entropy remains the same (as shown Table 4).

Fig. 2 shows the dependence of sensitivity and specificity with missing values percentage for each one of the fill in methods. For each case the sensitivity is not affected by the methods, which shows the effectiveness of the algorithm to find a model that contains the minimal false negative.

However, the specificity is affected by the method as a result of the increase of false positives.

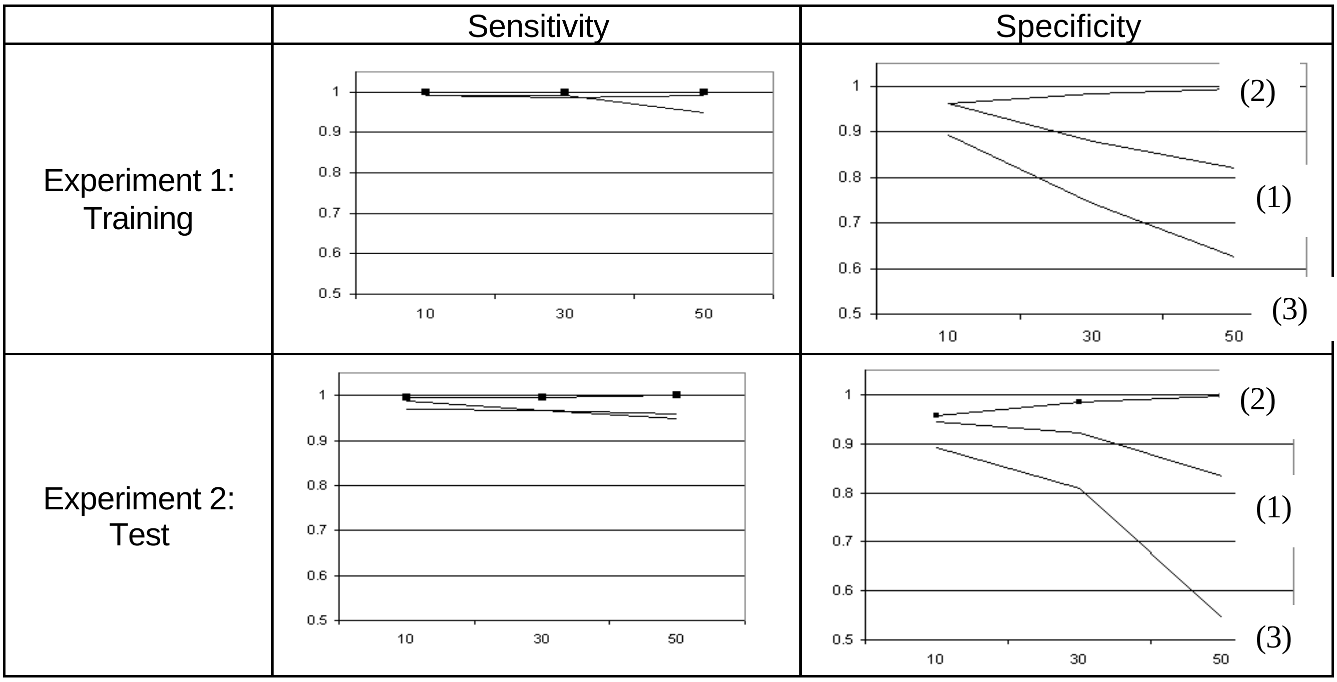

Fig. 2 Sensitivity and specificity as function of Missing Values percentage for (1) average, (2) conditional means, and (3) random number fill in methods for training and test stages of genetic programming.

Summary and conclusions.

This paper studied the effect of different fill in methods for missing values in evolvable discriminate function. The conditional means fill in method shows a better performance due to its reduction in the diffusiveness of the ill/healthy regions.

The average and random numbers in fill in methods show an increasing in the uncertainness of the regions, and as consequence a decrease in the accuracy.

There is a discussion point about use the diagnostic information to fill in the missing values. The use of the diagnostic to select the attribute average can improve the model accuracy, but it will not be used to new data because in this case the diagnostic is not available. In this case, the complete clinical analysis set will need to be done or the average or random method must be used. But in this case, the numbers will be replaced in the best available model, obtained with the filled database using conditional means method.

The recommendation from the results is to use the conditional means to fill in missing data, and use the database division to obtain a certain number of models that can evaluate the accuracy of the diagnostic for new values.

References.

[1] Little,R.J.A.; Rubin,D.B.; Statistical analysis with missing data; John Wiley & Sons, 1987.

[2] MacKay, D.J.C.; Information theory, inference and learning algorithms; http://www.inference.phy.cam.ac.uk/itprnn/book.pdf , 2003.

[3] Buck,S.F.; A method of estimation of missing values in multivariate data suitable for use with an electronic computer; J. Roy. Statist. Soci. B22, 302-306,1960.

[4] Ernest,L.R.; Variance of the estimated mean for several imputation procedure; American statistical association 1980, Proceedings of the survey research methods section pp. 716-720, 1980.

[5] Kalton,G.; Kish,L.; Two efficient random imputation procedures; American statistical association 1981, Proceedings of the survey research methods section pp. 146-151, 1981.

[6] Ford,B.N.; An overview of hot deck procedures, in Incomplete data in sample surveys, vol II: Theory and annotated bibliography (W.G.Madow, I.Olkin, and D.B.Rubin, Eds) New York: Academic Press, 1983.

[7] David,M.H.; Little,R.J.A.; Samuhel,M.E., and Triest,R.K.; Alternative methods for CPS income imputation, J.Am.Statist.Associ., 81,29-41,1986.

[8] HOLLAND,J.H. “Adaptation in natural and artificial systems: an introductory analysis with applications to biology, control and artificial intelligence.” Cambridge: Cambridge press 1992.

[9] GOLDBERG,D.E. “Genetic Algorithms in Search, Optimisation, and Machine Learning.” Reading, Mass.: Addison-Whesley, 1989.

[10] CHAMBERS,L.; “The practical handbook of Genetic Algorithms” Chapman & Hall/CRC,2000.

[11] KOZA,J.R. “Genetic programming: On the programming of computers by means of natural selection.” Cambridge,Mass.: MIT Press, 1992.

[12] LilGP “Genetic Algorithms Research and Applications Group (GARAGe)”, Michigan State University; http://garage.cps.msu.edu/software/lil-gp/lilgp-index.html

[13] Bradley, A.P.; “The use of the area under the ROC curve in the evaluation of machine learning algorithms”; Pattern Recognition, 30(7):1145-1159, 1997.

[14] Wolberg,W.H.; Street,W.N. ; Mangasarian, O.L.; Wisconsin Database on Breast Cancer; University of Wisconsin http://www.ics.uci.edu/~mlearn/MLRepository.html

[15] Werner,J.C.; Fogarty,T.C.; “Severe diseases diagnostics using Genetic Programming.” Intelligent Data Analysis in medicine and pharmacology – IDAMAP2001; September 4th, 2001 London http://magix.fri.uni-lj.si/idamap2001/scientific.asp

[16] Werner,J.C.; Kalganova,T.; Disease modeling using Evolved Discriminate Function.; in Proceedings of the EuroGP 2003 conference. To be published.

[17] RandLib.c odin.mdacc.tmc.edu/anonftp

[18] Frize M, Ennett CM, Stevenson M, Trigg HCE.: ‘Clinical decision support systems for intensive care units: using artificial neural networks’, Medical Engineering and Physics, 23, pp:217:225, 2001.