HSIM: A Supervised Imputation Method for Hierarchical Classification Scenario

Leandro R. Galvão(✉) and Luiz H.C. Merschmann

Computer Science Department, Federal University of Ouro Preto, Ouro Preto, Brazil
leandrodvmg@gmail.com, luizhenrique@iceb.ufop.br

Abstract. The missing value imputation process can be defined as a preprocessing step that fills missing values of attributes in incomplete datasets. Nowadays, the problem of incomplete datasets in the hierarchical classification scenario must be solved using unsupervised missing value imputation methods due to the lack of supervised methods to deal with the hierarchical context. Thus, in this work, we propose and evaluate a supervised missing value imputation method for datasets used in hierarchical classification problems in which the classes are organized into tree structure. Experiments were performed on incomplete datasets to evaluate the effect of the proposed missing value imputation method on classification performance when using a global hierarchical classifier. The results showed that, using the proposed method for dealing with missing attribute values, it provided higher classifier predictive performance than other unsupervised missing value imputation methods.

Keywords: Missing attribute value imputation · Hierarchical classification · Data mining

1 Introduction

The technological advances in the last decades have allowed the production and storage of a huge amount of data related to different types of applications. The transformation of this data in useful, valid and understandable information is essential. However, this is not an easy task, requiring automated strategies to analyse the data [1]. Thus, the Knowledge Discovery from Data (KDD) process adopted for this purpose is basically composed by three main steps: data preprocessing, data mining and results validation.

The missing value imputation process can be defined as a preprocessing step that fills missing values of attributes in incomplete datasets [2]. Attribute's missing value can occur in a dataset for several reasons, such as filling failure, omission of data by respondents in survey questions or even a failure on the sensor responsible to collect the data. Missing values can make the manipulation of datasets more complex and reduce the efficiency of data mining algorithms [3].

Classification is a data mining task that aims to identify an instance's class through its characteristics [1]. Different types of classification problems can be

T. Calders et al. (Eds.): DS 2016, LNAI 9956, pp. 134–148, 2016.
DOI: 10.1007/978-3-319-46307-0_9

found in the literature, each one with its own complexity level [4]. In flat classification problems each instance is assigned to a class, in which classes do not have relationships to each other. Nevertheless, there are more complex classification problems, known as hierarchical classification problems, in which the classes are hierarchically organized.

Several application domains such as text categorization [5,6], protein function prediction [7,8], music genre classification [9,10], image classification [11,12] and emotional speech classification [13] can benefit from hierarchical classification techniques, since the classes to be predicted are naturally organized into class hierarchies. Despite of some works have ignored the class hierarchy and performed predictions considering only leaf node classes (flat classification approach), hierarchical classification methods are overall better than flat classification methods when solving hierarchical classification problems [4]. Therefore, hierarchical classification is a research topic that certainly deserves attention.

In literature, it is possible to find several works that deal with missing attribute values. Expectation Maximization and KNNImpute are examples of popular methods often used to handle missing attribute values [14–16]. Since they are unsupervised methods (ignore the target class values), they can be applied to datasets used in flat or hierarchical classification scenario. Thus, in hierarchical classification context, due to the lack of suitable supervised missing value imputation methods (able to take into account the class relationships in the target problem), the researchers are limited to use unsupervised missing value imputation techniques. [17–19] are examples of works that have used an unsupervised missing value imputation method for hierarchical classification context.

Therefore, in this work, we fill this gap by presenting and evaluating a supervised missing value imputation method for datasets for hierarchical classification task. Experiments performed on incomplete datasets using a global hierarchical classifier showed that the method proposed to deal with missing attribute values provided higher classifier predictive performance than other popular unsupervised missing value imputation methods.

2 Background

2.1 Hierarchical Classification

Several classification problems are available in literature wherein the huge majority of them deals with the flat classification scenario. In flat classification problems there is no relationship among the classes. However, there are more complex classification problems where the classes are hierarchical organized as a tree or DAG (Direct Acyclic Graph). This group of problems is known as hierarchical classification problems [4].

Hierarchical classification methods can be analysed according to different aspects. The first aspect is related to the hierarchical structure the method is able to deal with. The classes can be hierarchically organized in a tree or DAG (Direct Acyclic Graph) structure. The main difference between these structures

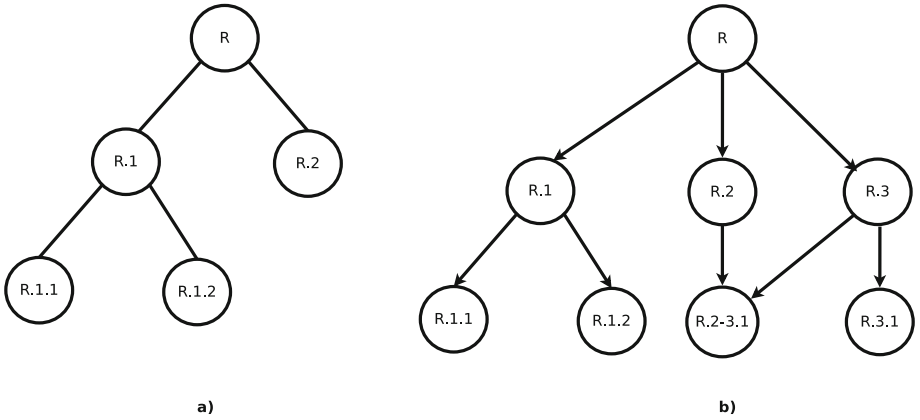

Fig. 1. (a) Class hierarchy structured as tree; (b) Class hierarchy structured as DAG.

is related to the number of parents of a class. The tree structure restricts each class to possess only one parent class. However, in the DAG structure a class (node) is allowed to have more than one parent class. Figure 1 presents a tree and a DAG example, where the nodes represent the classes and the edges indicate a relationship between them.

The second aspect is related to the prediction depth of the method. Thus, hierarchical classification problems can either be organized in mandatory leaf node prediction (MLNP) or non-mandatory leaf node prediction (NMLNP). In the mandatory leaf node prediction, the output of a classifier is always a class represented by a leaf node of the class hierarchy. For the non-mandatory leaf node prediction, the most specific class predicted by the classifier can be represented by a node at any level (internal or leaf) of the class hierarchy.

The third aspect refers to the number of different paths in the hierarchy a method can assign to a given instance. This aspect defines two different types of problems: single path of labels (SPL) and multiple path of labels (MPL). Single path problems restrict each instance to be assigned to at most one path of predicted labels whilst multiple path problems allow an instance to be assigned to multiple paths of predicted labels.

Finally, the fourth aspect concerns how the hierarchical structure is handled by the method. In [4], three approaches are listed: flat classification, which do not take into account the class hierarchy and performs predictions considering only the leaf node classes; local model approaches, when a group of flat classifiers are employed; and global model approaches, when a single classifier is built considering the class hierarchy as a whole.

Several works in literature consist of proposals to modify existing flat classifiers to cope with the entire class hierarchy in a single step and, therefore, creating global classifiers. Some examples of modifications of traditional flat classification algorithms are: HC4.5 [20] and HLC [21] (modified versions of the C4.5), Global Model Naive Bayes (modified version of the Naive Bayes) [22],

Clus-HMC (based on Predictive Cluster Trees) [23,24] and hAnt-Miner (adaptation of Ant-Miner algorithm) [25]. Given the relevance of global classifiers to the hierarchical classification scenario, one can see that it is important the development of preprocessing techniques to deal with the class hierarchy as a whole. Thus, in this work, we propose a supervised missing value imputation method for datasets used by global hierarchical classifiers.

2.2 Missing Value Imputation

Missing value imputation can be defined as the estimation of values based on the analysis of the known values of the attribute [26]. The missing values in data can be the result of failure during the data collection process, accidental removal of values or refusal to answer questions in surveys used to collect data. Whatever the reasons, missing attribute values can pose an obstacle for classification and other data mining tasks.

The missing value imputation methods can be categorized according to different criteria [2]. Methods that use the target class values during the imputation process are categorized as supervised, while methods that ignore the target class values are categorized as unsupervised. While univariate methods impute missing values on each attribute for each instance based on observed values for other instances on same attribute, multivariate methods impute missing values on each attribute for each instance based on observed values for same instance on other attributes.

The strategies used to impute missing attribute values vary from simple statistical algorithms like mean imputation to more complex approaches based on predictive models.

Mean imputation, one of the easiest ways to impute missing values, works by replacing each missing value with mean of observed values of that attribute for other instances. Despite its simplicity, this method changes the distribution of attribute, producing different kinds of bias.

Expectation Maximization [27] algorithm is a popular method to impute missing attribute values. It is an iterative refinement method that assumes the data are distributed based on a parametric model with unknown parameters. Basically, in each iteration, it estimates missing values and parameters model. After parameters model initialization, it has an Expectation step, where a missing attribute value for an instance i is substituted by its expected value computed from the estimates for parameters model and observed values for same instance i on other attributes. Then, in the Maximization step, parameters model are updated in order to maximize the complete data likelihood. These two steps are iteratively repeated until convergence is obtained. The Expectation Maximization method is categorized as unsupervised and multivariate.

KNNImpute [15] is another widely used method to impute missing attribute values. In short, it selects the k nearest instances (neighbours) from dataset instances with known value in the attribute to be imputed. Aiming at finding the nearest neighbours a distance measure (e.g., Euclidean distance) must be adopted. Then, the missing attribute value is replaced by a value calculated from

the k selected neighbours. Usually, mode (for categorical data) and mean (for continuous data) are used to compute the replacement value. The KNNImpute method is categorized as unsupervised and multivariate.

3 Proposed Method

The missing value imputation method proposed in this work will be referenced as *Hierarchical Supervised Imputation Method* (HSIM). The HSIM is a supervised method able to deal with datasets containing classes hierarchically organized in a tree structure.

The motivation behind the HSIM is to take into account the class hierarchy to impute missing attribute values. In short, whenever there are no known attribute values for the instances associated to a particular class, the main idea of the proposed method is replacing each missing value with mean (for continuous data) or mode (for categorical data) of observed values of that attribute for other instances associated to a class descendant or ascendant of the class of the instance containing the missing value.

An example is now presented to illustrate the operation of the proposed method. Consider the incomplete dataset shown in Table 1, where the instances 2, 7, 9 and 13 contain missing value for attribute F. For each of these instances, a different strategy is adopted by HSIM to impute missing attribute value.

Note that instance 2 is associated to the class R.2, that is also the class associated to other instances (3, 6 and 10) containing observed values of the attribute F. In this case, the missing value is substituted by the average of the

Table 1. Incomplete dataset

ID	F	Class attribute
1	0.15	R.1.1
2	?	R.2
3	3.41	R.2
4	4.12	R.2.1
5	0.22	R.1.2
6	3.5	R.2
7	?	R.1
8	0.34	R.1.2
9	?	R.3.1.1
10	4.1	R.2
11	1.6	R.3.1
12	1.55	R.3
13	?	R.4
14	1.71	R.3.1

three observed values (3.41, 3.5 and 4.1) of the instances associated to the same class of the instance containing the missing value (R.2). Thus, the attribute value of instance 2 is imputed as 3.67.

In the case of instance 7, where the missing value is associated to the class R.1, since there are no other instances associated to that class, the missing value is imputed with the average of the observed values (0.15, 0.22 and 0.34) of instances associated to descendant classes of the class associated to the instance containing the missing value. Thus, the attribute value of the instance 7 is imputed as 0.24.

The previous strategies are not applicable to the case of instance 9, as the missing value is associated to the class R.3.1.1, which is not associated to any other dataset instance and neither is an ascendant class of any class associated to an instance with observed value of the attribute F. Then, the missing value is replaced with mean of the observed values (1.6, 1.55 and 1.71) of instances associated to ascendant classes of the class associated to the instance containing the missing value. Thus, the attribute value of the instance 9 is imputed as 1.62.

Finally, instance 13 has a missing value associated to the class R.4, which is not associated to any other dataset instance and neither is an ascendant or descendant class of any class associated to an instance with observed value of the attribute F. Then, the missing value is replaced with mean of observed values of attribute F for all other instances. In this example, the attribute value of the instance 13 is imputed as 2.07.

Table 2 shows the complete dataset achieved after application of the HSIM on the incomplete dataset presented in Table 1. In this table, the imputed attribute values are highlighted in bold. The steps of the proposed method described in the Algorithm 1 are detailed next.

Algorithm 1 describes the steps of the proposed method. First, HSIM receives as input a dataset represented by an $M \times N$ matrix, where M is the number of instances and N is the total number of attributes (predictive attributes and class attribute). In addition, we consider that the last matrix column is the class attribute. By scanning the data matrix (lines 1 and 2), whenever a missing value for attribute j in instance i is found (line 3), four empty vectors (*sameClass*, *descendantClass*, *ascendantClass* and $differentClass$) are initialized (lines 4, 5, 6 and 7). Then, all known values for attribute j associated to the same class of the instance i (line 10) are stored in the vector *sameClass* (line 11). Similarly, all known values for attribute j associated to a class descendant of the class of the instance i (line 12) are stored in the vector *descendantClass* (line 13). In the same way, all known values for attribute j associated to a class ascendant of the class of the instance i (line 14) are stored in the vector *ascendantClass* (line 15). The remaining known values for attribute j associated to a class non-descendant and non-ascendant of the class of the instance i are stored in the vector $differentClass$ (line 17). After, if the vector *sameClass* has some element, the average or mode of the *sameClass* elements is used to impute the missing value (lines 23 and 25). Otherwise, if the vector *descendantClass* is not empty, the average or mode of the *descendantClass* elements is used to impute the missing value (lines 29 and 31).

Table 2. Complete dataset

ID	F	Class attribute
1	0.15	R.1.1
2	**3.67**	R.2
3	3.41	R.2
4	4.12	R.2.1
5	0.22	R.1.2
6	3.5	R.2
7	**0.24**	R.1
8	0.34	R.1.2
9	**1.62**	R.3.1.1
10	4.1	R.2
11	1.6	R.3.1
12	1.55	R.3
13	**2.07**	R.4
14	1.71	R.3.1

Alternatively, when *sameClass* and *descendantClass* vectors are empty, if the vector *ascendantClass* is not empty, the average or mode of the *ascendantClass* elements is used to impute the missing value (lines 35 and 37). Finally, whenever *sameClass*, *descendantClass* and *ascendantClass* vectors are empty, the average or mode of the $differentClass$ elements is used to impute the missing value (lines 41 and 43).

4 Computational Experiments

4.1 Datasets and Experimental Setup

Computational experiments were carried out on incomplete datasets to evaluate the effect of the proposed missing value imputation method on classification performance when using a global hierarchical classifier. Thus, the proposed method (HSIM) was compared against the following popular unsupervised missing value imputation methods: Mean Imputation (MI), Expectation Maximization (EM) and KNNImpute. Since there are no supervised missing value imputation methods proposed in literature for hierarchical classification context, we consider that it is fair to have a comparison of proposed method against unsupervised methods, given that they can be directly applied for datasets used in hierarchical classification scenario.

While KNNImpute algorithm was implemented in C++ programming language, for MI and EM, the experiments were executed using the WEKA [28]

Algorithm 1. HSIM

```
Input: DB matrix ; //Incomplete Dataset
Output: DB_full matrix ; //Complete Dataset
for j = 1; j < N − 1; j + + do
    for i = 1; i <= M; i + + do
        if DB[i][j] == "?" then
            sameClass ← ∅
            descendantClass ← ∅
            ascendantClass ← ∅
            differentClass ← ∅
            for k = 1; k <= M; k + + do
                if k ≠ i and DB[k][j] ≠ "?" then
                    if DB[k][N] == DB[i][N] then
                        sameClass.insert(DB[k][j]);
                    else if isDescendant(DB[k][N], DB[i][N]) then
                        descendantClass.insert(DB[k][j]);
                    else if isAscendant(DB[k][N], DB[i][N]) then
                        ascendantClass.insert(DB[k][j]);
                    else
                        differentClass.insert(DB[k][j]);
                    end if
                end if
            end for
            if size(sameClass) > 0 then
                if is_continuous(j) then
                    DB_full[i][j] = average(sameClass);
                else
                    DB_full[i][j] = mode(sameClass);
                end if
            else if size(descendantClass) > 0 then
                if is_continuous(j) then
                    DB_full[i][j] = average(descendantClass);
                else
                    DB_full[i][j] = mode(descendantClass);
                end if
            else if size(ascendantClass) > 0 then
                if is_continuous(j) then
                    DB_full[i][j] = average(ascendantClass);
                else
                    DB_full[i][j] = mode(ascendantClass);
                end if
            else
                if is_continuous(j) then
                    DB_full[i][j] = average(differentClass);
                else
                    DB_full[i][j] = mode(differentClass);
                end if
            end if
        else
            DB_full[i][j] = DB[i][j]
        end if
    end for
end for
```

implementations, named ReplaceMissingValues and EMImputation, respectively. For KNNImpute, the experiments were conducted by varying the parameter k (number of nearest instances considered to impute missing values) between 10 % and 50 % of the number of instances in the dataset, in increments of 10 %. Since the best results were obtained for $k = 10\,\%$, it was adopted to obtain the results presented here. For EM, WEKA's default parameters were used.

Experiments were conducted by running both the proposed and the baseline methods on 8 bioinformatics datasets related to gene functions of yeast. In these datasets, the predictor attributes include the following types of bioinformatics data: secondary structure, phenotype, homology, sequence statistics, and expression. In addition, the classes to be predicted are hierarchically organized in a tree structure. The datasets, initially presented in [20], were multi-label data. Since in this work we focus on single path label scenario, before running the missing value imputation algorithms, the datasets were converted into single label data by choosing one class for each instance. This process consisted of selecting, for each instance, the more frequent class in the original dataset. HSIM implementation and the single label datasets are available at https://github.com/leandrodvmg/HSIM. Table 3 provides the main characteristics of the datasets used in the experiments. This table shows, for each dataset, its number of predictive attributes, number of instances, number of classes at each hierarchy level (1st/2nd/3rd/4th/5th/6th levels), percentage of instances containing at least one missing attribute value and percentage of missing data in the $M \times P$ matrix, where M is the number of instances and P is the total number of predictive attributes.

After all datasets were processed by both the proposed and the baseline missing value imputation methods, the imputation quality was measured by running a global hierarchical classifier on these datasets. However, before running the classifier, as in [7,9,29], an unsupervised discretization algorithm based on equal-frequency binning (using 20 bins) was applied to continuous attributes.

Table 3. Characteristics of the datasets

Dataset	# Attributes	# Instances	# Classes per level	% Incomplete instances	% Missing values
	Categorical / Continuous				
CellCycle	0 / 77	3758	8/37/73/46/25/2	93.45	5.57
Church	1 / 26	3756	8/37/72/47/25/2	61.76	9.65
Eisen	0 / 79	2425	5/26/55/34/22/2	75.45	1.93
Expr	4 / 547	3780	8/37/73/46/26/2	100.00	8.90
Gasch1	0 / 173	3765	8/37/73/46/26/2	86.34	2.27
Gasch2	0 / 52	3780	8/37/73/46/26/2	61.15	3.55
Sequence	5 / 473	3920	8/37/73/46/26/2	0.66	0.01
SPO	3 / 77	3704	8/37/73/46/26/2	99.43	2.28

The Global-Model Naive Bayes (GMNB) [22], an extension of the flat classifier Naive Bayes to deal with hierarchical classification problems, was the global hierarchical classifier adopted in these experiments. It makes possible predictions at any level of the class hierarchy. In order to evaluate the predictive performance of the hierarchical classifier GMNB, we used the 10-fold cross validation method [1] and the hierarchical F-measure, an adaptation of the flat F-measure customized for hierarchical classification scenario. For each dataset, the same ten folds were used in the evaluation of the GMNB classifier.

4.2 Computational Results

As mentioned earlier, the objective of experiments was to compare the hierarchical classifier performances when running on datasets preprocessed using different missing value imputation methods. More specifically, the HSIM method was compared against each one of the baseline methods (Mean Imputation, Expectation Maximization and KNNImpute). Therefore, for each dataset, in order to determine if there is a statistically significant difference between the F-measures of the GMNB classifier when running on the dataset preprocessed by HSIM and by other baseline method, we have used the Wilcoxon's Signed-Rank Test (two-sided test) with Bonferroni adjustment on the results as we are making many-to-one comparisons [30]. This statistical test was applied with 95 % of confidence level.

Experimental results are shown in Table 4 for each dataset listed in the first column. This table shows, from the second to fifth column, the average hierarchical F-measure (hF) achieved by GMNB classifier (with standard deviation in parentheses) when running on each dataset preprocessed by the missing value imputation methods Mean Imputation (MI), Expectation Maximization (EM), KNNImpute (KNN) and HSIM, respectively. In bold we mark the best result achieved for each dataset. In addition, the ♦ symbol after an hF value indicates that the difference between that baseline method and HSIM holds statistical significance. Finally, the last row of the table summarizes the results of statistical test, i.e., for each baseline method, it is presented the number of times the HSIM outperformed the baseline method by providing a better GMNB classifier performance.

From results presented in Table 4 it is possible to observe that for most of datasets the GMNB classifier achieved higher predictive performance when the dataset was preprocessed using the proposed HSIM. In 6 out of 8 datasets, HSIM obtained significantly better results than MI and, in the remaining two datasets there was no statistically significant difference between the two methods. EM is outperformed by HSIM, with statistical significance, in 5 out of 8 datasets and, in the remaining datasets, the difference between the methods was not statistically significant. Finally, HSIM outperformed KNNImpute in 7 out of 8 datasets with statistical significance and, in the remaining dataset there was no statistically significant difference between the methods. It is also interesting to note that, in 5 out of 8 datasets, the HSIM outperformed all baseline methods by providing significantly better GMNB predictive performance. For only one

Table 4. Experimental Results

Dataset	MI + GMNB hF (std. error)	EM + GMNB hF (std. error)	KNN + GMNB hF (std. error)	HSIM + GMNB hF (std. error)
CellCycle	15.57 (2.00) ♦	15.91 (1.13) ♦	16.00 (1.70) ♦	**27.17 (2.17)**
Church	8.42 (1.20) ♦	8.31 (1.17) ♦	8.26 (0.95) ♦	**13.10 (1.41)**
Eisen	20.59 (2.23)	20.56 (1.66)	20.06 (1.36) ♦	**21.88 (1.68)**
Expr	19.62 (1.84) ♦	20.27 (1.27) ♦	20.20 (1.51) ♦	**45.64 (2.32)**
Gasch1	18.29 (1.30) ♦	18.47 (2.06) ♦	18.22 (1.61) ♦	**22.97 (1.86)**
Gasch2	15.37 (1.30) ♦	15.44 (1.35) ♦	15.44 (1.65) ♦	**19.55 (1.78)**
Sequence	18.73 (1.13)	**18.95 (1.48)**	18.76 (1.15)	18.75 (1.15)
SPO	13.37 (1.03) ♦	13.22 (1.13)	13.37 (1.03) ♦	**14.36 (0.76)**
HSIM wins	6	5	7	

dataset (Sequence) HSIM was statistically equivalent to all baseline imputation methods.

When analysing the results showed in Table 4 and the percentage of missing values in datasets presented in Table 3, it is interesting to note that HSIM improves the most over the other methods on datasets with large percentage of missing values. Besides, the unique dataset (Sequence) where HSIM does not outperform any baseline method has very small percentage (0.01 %) of missing values.

In order to contribute to understand the results presented in Table 4, in the graphs of Fig. 2, it is presented the distribution of missing value per attribute as well as information on the predictive power of the attributes for classification purpose. The hierarchical Symmetrical Uncertainty (SU_H), originally proposed in [31] to deal with feature selection for hierarchical classification problems, was adopted as measure of predictive power of each attribute. A higher SU_H indicates better predictive power. In these graphs, the bars represent the percentage of missing value of each attribute while the solid and dashed lines correspond to SU_H of the attributes without missing value imputation and with missing value imputation using HSIM, respectively.

From the graphs presented in Fig. 2, we can observe that for most of datasets (6 out of 8) the predictive power (according to SU_H) of several attributes improved after missing value imputation process using HSIM. These graphs show that the missing value imputation does not necessarily imply higher predictive power of attributes, since it depends on the quality of imputation. Nevertheless, even for datasets where only a few attributes had their predictive power increased after missing value imputation process (e.g., Church dataset), it is possible to verify the improvement of the predictive performance of the GMNB classifier.

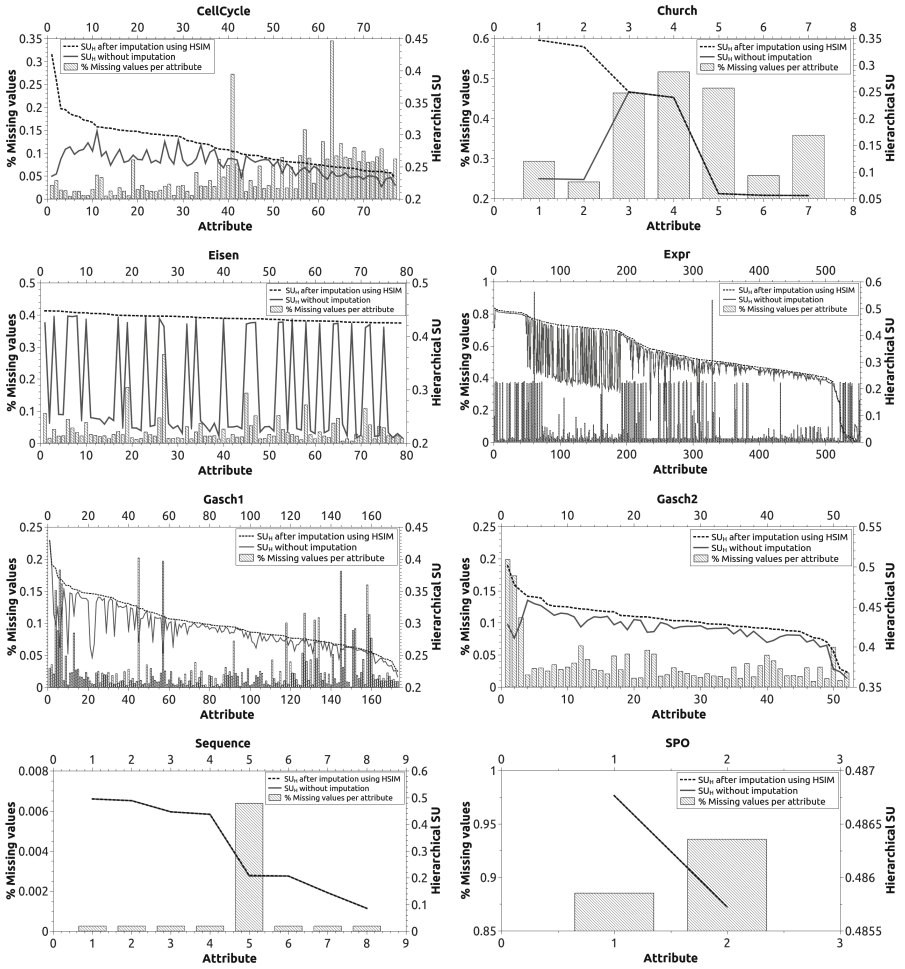

Fig. 2. Percentage of missing values and predictive power for each attribute with missing data.

5 Conclusion

In data mining applications, incomplete datasets is a very common situation. Since many classification algorithms are sensitive to missing attribute values, they can pose an obstacle for classification and other data mining tasks. Although several methods for substituting the missing values can be found in the literature, to the best of our knowledge, for hierarchical classification scenario, there are no supervised methods to deal with the hierarchical context. Therefore, in this work, we proposed and evaluated a supervised missing value imputation method for datasets used in the hierarchical classification problems.

The proposed method, named HSIM, takes into account the class relationships in the target problem to impute missing attribute values. The main idea of HSIM is replacing each missing value with mean or mode of observed values of that attribute for other instances associated to a class descendant or ascendant of the class of the instance containing the missing value. This procedure is adopted whenever there are no known attribute values for the instances associated to a particular class.

The evaluation of the proposed method was conducted on 8 bioinformatics datasets by comparing it against the following popular unsupervised missing value imputation methods: Mean Imputation, Expectation Maximization and KNNImpute. As the objective was to evaluate the effect of the proposed missing value imputation method on classification performance when using a global hierarchical classifier, the imputation quality was measured by running the global hierarchical classifier, known as Global-Model Naive Bayes, on datasets preprocessed by aforementioned imputation methods.

In our experiments, for most of datasets, the hierarchical classifier achieved the best predictive performance when the dataset was preprocessed using the proposed HSIM. Considering the Wilcoxon's Signed-Rank statistical test with Bonferroni correction, the HSIM outperformed all baseline imputation methods (by providing significantly better GMNB predictive performance) in 5 out of 8 datasets. In the remaining three datasets, HSIM reached results statistically equivalent or better than baseline methods. Therefore, we conclude that the proposed missing value imputation method has shown good performance in the hierarchical classification context.

As future work we intend to evaluate the performance of the method proposed in this work in other application domains, such as image classification, music genre classification and text categorization. We also intend to extend the HSIM to deal with hierarchical multi-label classification scenario.

Acknowledgements. This research was partially supported by CNPq, FAPEMIG, UFOP, and by individual grants from CAPES.

References

1. Han, J., Kamber, M.: Data Mining: Concepts and Techniques: Concepts and Techniques. Elsevier, Amsterdam (2011)
2. Little, R.J., Rubin, D.B.: Statistical Analysis with Missing Data. Probability and Statistics, vol. 1, 2nd edn. Wiley, New York (2002)
3. Schafer, J.L., Graham, J.W.: Missing data: our view of the state of the art. Psychol. Methods **7**(2), 147 (2002)
4. Silla Jr., C.N., Freitas, A.A.: A survey of hierarchical classification across different application domains. Data Min. Knowl. Disc. **22**(1–2), 31–72 (2011)
5. Qiu, X., Huang, X., Liu, Z., Zhou, J.: Hierarchical text classification with latent concepts. In: Proceedings of the 49th Annual Meeting of the Association for Computational Linguistics: Human Language Technologies: Short Papers, vol. 2, pp. 598–602. Association for Computational Linguistics (2011)

6. Dollah, R.B., Aono, M.: Classifying biomedical text abstracts based on hierarchical 'concept' structure. World Acad. Sci. Eng. Technol. Int. J. Comput. Electr. Autom. Control Inf. Eng. **5**(2), 178–183 (2011)
7. Campos Merschmann, L.H., Freitas, A.A.: An extended local hierarchical classifier for prediction of protein and gene functions. In: Bellatreche, L., Mohania, M.K. (eds.) DaWaK 2013. LNCS, vol. 8057, pp. 159–171. Springer, Heidelberg (2013). doi:10.1007/978-3-642-40131-2_14
8. Valentini, G.: Hierarchical ensemble methods for protein function prediction. ISRN Bioinf. **2014** (2014)
9. Silla, C.N., Freitas, A.A.: Novel top-down approaches for hierarchical classification and their application to automatic music genre classification. In: 2009 IEEE International Conference on Systems, Man and Cybernetics, SMC 2009, pp. 3499–3504. IEEE (2009)
10. Ariyaratne, H.B., Zhang, D.: A novel automatic hierachical approach to music genre classification. In: 2012 IEEE International Conference on Multimedia and Expo Workshops (ICMEW), pp. 564–569. IEEE (2012)
11. Binder, A., Kawanabe, M., Brefeld, U.: Efficient classification of images with taxonomies. In: Zha, H., Taniguchi, R., Maybank, S. (eds.) ACCV 2009. LNCS, vol. 5996, pp. 351–362. Springer, Heidelberg (2010). doi:10.1007/978-3-642-12297-2_34
12. Kramer, G., Bouma, G., Hendriksen, D., Homminga, M.: Classifying image galleries into a taxonomy using metadata and wikipedia. In: Bouma, G., Ittoo, A., Métais, E., Wortmann, H. (eds.) NLDB 2012. LNCS, vol. 7337, pp. 191–196. Springer, Heidelberg (2012). doi:10.1007/978-3-642-31178-9_20
13. Le, B.V., Bang, J.H., Lee, S.: Hierarchical emotion classification using genetic algorithms. In: Proceedings of the Fourth Symposium on Information and Communication Technology, pp. 158–163. ACM (2013)
14. Van Hulse, J., Khoshgoftaar, T.M.: Incomplete-case nearest neighbor imputation in software measurement data. Inf. Sci. **259**, 596–610 (2014)
15. Troyanskaya, O., Cantor, M., Sherlock, G., Brown, P., Hastie, T., Tibshirani, R., Botstein, D., Altman, R.B.: Missing value estimation methods for dna microarrays. Bioinformatics **17**(6), 520–525 (2001)
16. Rahman, M.G., Islam, M.Z.: IDMI: a novel technique for missing value imputation using a decision tree and expectation-maximization algorithm. In: 2013 16th International Conference on Computer and Information Technology (ICCIT), pp. 496–501. IEEE (2014)
17. Bi, W., Kwok, J.T.: Multi-label classification on tree-and dag-structured hierarchies. In: Proceedings of the 28th International Conference on Machine Learning (ICML 2011), pp. 17–24 (2011)
18. Sun, Z., Zhao, Y., Cao, D., Hao, H.: Hierarchical multilabel classification with optimal path prediction. Neural Process. Lett., 1–15 (2016)
19. Cerri, R., Barros, R.C., de Carvalho, A.: Hierarchical classification of gene ontology-based protein functions with neural networks. In: IEEE International Joint Conference on Neural Networks (IJCNN), pp. 1–8 (2015)
20. Clare, A., King, R.D.: Predicting gene function in saccharomyces cerevisiae. Bioinformatics **19**(suppl 2), ii42–ii49 (2003)
21. Chen, Y.L., Hu, H.W., Tang, K.: Constructing a decision tree from data with hierarchical class labels. Expert Syst. Appl. **36**(3), 4838–4847 (2009)
22. Silla, C.N., Freitas, A.A.: A global-model naive bayes approach to the hierarchical prediction of protein functions. In: 2009 Ninth IEEE International Conference on Data Mining, ICDM 2009, pp. 992–997. IEEE (2009)

23. Blockeel, H., Schietgat, L., Struyf, J., Džeroski, S., Clare, A.: Decision trees for hierarchical multilabel classification: a case study in functional genomics. In: Fürnkranz, J., Scheffer, T., Spiliopoulou, M. (eds.) PKDD 2006. LNCS (LNAI), vol. 4213, pp. 18–29. Springer, Heidelberg (2006). doi:10.1007/11871637_7
24. Vens, C., Struyf, J., Schietgat, L., Džeroski, S., Blockeel, H.: Decision trees for hierarchical multi-label classification. Mach. Learn. **73**(2), 185–214 (2008)
25. Otero, F.E.B., Freitas, A.A., Johnson, C.G.: A hierarchical classification ant colony algorithm for predicting gene ontology terms. In: Pizzuti, C., Ritchie, M.D., Giacobini, M. (eds.) EvoBIO 2009. LNCS, vol. 5483, pp. 68–79. Springer, Heidelberg (2009). doi:10.1007/978-3-642-01184-9_7
26. Brown, M.L., Kros, J.F.: Data mining and the impact of missing data. Ind. Manag. Data Syst. **103**(8), 611–621 (2003)
27. Dempster, A.P., Laird, N.M., Rubin, D.B.: Maximum likelihood from incomplete data via the EM algorithm. J. Roy. Stat. Soc.: Ser. B (Methodol.), 1–38 (1977)
28. Hall, M., Frank, E., Holmes, G., Pfahringer, B., Reutemann, P., Witten, I.H.: The weka data mining software: an update. ACM SIGKDD Explor. Newsl. **11**(1), 10–18 (2009)
29. Borges, H.B., Silla, C.N., Nievola, J.C.: An evaluation of global-model hierarchical classification algorithms for hierarchical classification problems with single path of labels. Comput. Math. Appl. **66**(10), 1991–2002 (2013)
30. Japkowicz, N., Shah, M.: Evaluating Learning Algorithms. Cambridge University Press, Cambridge (2011)
31. Dias, T.N., Merschmann, L.H.C.: Adaptação da medida incerteza simétrica para a seleção de atributos no contexto de classificação hierárquica monorrótulo. In: Anais do Encontro Nacional de Inteligência Artificial e Computacional, Natal, RN, Brazil, pp. 142–149 (2015)