PAPER PUBLICATIONS
IMPUTATION METHODS:
AN ALGORITHMIC APPROACH

K. Fayaz & G. A. Ramachandra
Department of Computer Science & Technology, S. K. University, Anantapur, India

Abstract: Today’s real-world databases are highly susceptible to noisy, missing and inconsistent data due to their typically huge size, often several gigabytes or more[1]. The main objective of this research is to advocate the existing models of imputation and the framework proposed in the work. Single imputation models require domain knowledge to fill missing values, whereas multiple imputation are akin, but generate a required pattern and search for the right fit of the data to impute. This is proposed with an information-hash theoretic approach consisting of corpus and database. This approach implements complete scan of database for tuple sampling, generate characteristic weights into characteristic corpus. The characteristic weights of tuples with missing values are scanned in the corpus and the set of plausible values are established in the database. Hash-based algorithms is used to compute characteristic weights.

Keywords: Imputation, Corpus, Missing data, MCAR, MAR, NMAR, Characteristic weights, MVD

INTRODUCTION

Missing data is omnipresent in survey research. Although when we collect information for statistical analysis, complete data for all subjects are desired, the possibility that some data will be unavailable should not be ignored. It may be lost, be costly to obtain or be unusable. When missing data means there is no response obtained for a whole unit in the survey, it is called unit nonresponse. When missing data means responses are obtained for some of the items for a unit but not for other items, it is called item nonresponse. Missing data has to be dealt with before we can do anything meaningful with the dataset. Many statistical problems have been viewed as missing data problems in the sense that one has to work with incomplete data.

Advances in computer technology have not only made previously long and complicated numerical calculations a simple matter but also advanced statistical analysis of missing data. Missing data usually means lack of responses in the data. It is often indicated by “Don’t know”, “Refused”, “Unavailable” and so on [1]. Missing data are problematic because most statistical procedures require a value for each variable. When a data set is incomplete, an analyst has to decide how to deal with it. This requires a missing value procedure.

RELATED WORK

Data may be missing in any type of study due to any reason. For example, subjects in longitudinal studies often drop out before the study is complete. Sometimes this happens because they are not interested anymore, are not able to find time to participate, died or moved out of the area. Whatever the reason is, the study will suffer from missing-data problem.
Missing data causes a variety of problems in data analysis. First, lost data decrease statistical power. Statistical power refers to the ability of an analytic technique to detect a significant effect in a data set. Also, it is well known that a high level of power often requires a large sample. Thus, it appears that missing data may meaningfully diminish sample size and power. Second, missing data produce biases in parameter estimates and can make the analysis harder to conduct and the results harder to present. The bias may be either upward or downward, which means the true score may be either overestimated or underestimated [2].

Imputation: “The action of attributing something, usually a fault or a crime to someone.”

Hitherto, there are methods that deal with missingness, they mean imputation and multiple imputations. Imputation consists of replacing the missing data with values derived from the respondents or from a relationship between the nonrespondents and respondents.

According to Little and Rubin [4], the mechanisms leading to missing data can be classified into three subgroups:

- Missing completely at random (MCAR)
- Missing at random (MAR) and
- Not missing at random (NMAR).

MCAR means that the missing data mechanism is unrelated to the variables under study, whether missing or observed: a missing response happens purely by chance.

That is \( f(M | Y, Q) = f(M | Q) \) for all \( Y, Q \). where \( Y_{obs} \) denote the observed components of \( Y \) and let \( Y_{mis} \) denote the missing components.

Overview of Single Imputation

When a large database will be analyzed by many users, there is a desire to “clean up” the data, which includes dealing with missing values. The reason is that standard procedures cannot be used when there are missing values and corresponding procedures that adjust for missing values may not be easy to derive. Imputation is one of the most common procedures for handling missing values [5]. Single imputation is just as the name suggests, filling in a single value for each missing value. Single imputation is attractive for several reasons.

(i) It saves a great deal of data that listwise deletion drops since it keeps all the rest of the data for an individual for use in analysis. It also saves more data than pairwise deletion since it preserves the data paired with a previously missing value [3]. Overall, it retains data in incomplete cases that would have been discarded if the analyses were restricted to complete cases such as when using listwise deletion and pairwise deletion.

(ii) When descriptive statistics and other statistical measures are of interest, standard complete-data methods of analysis can be used on the filled-in data set. However, some of the statistical measures are biased using the filled-in data set. For example, mean imputation underestimates the variance. Complete data software seems to keep closer pace with the statistical methodological developments than incomplete data software. An example of complete data software that can be used to process the data is SPSS. Mean imputation, which is one single imputation option, appears in several SPSS procedures.
(iii) Sometimes data is missing because it is confidential information. Thus, the public cannot view these data. In those events, the data producers, who have access to all the data, are able to incorporate all the knowledge by imputation into the data set for public use. The public sees the imputed dataset, not the original full dataset, because some of the data are confidential.

SINGLE IMPUTATION METHODS FOR DEALING WITH MISSING DATA

Mean Imputation

Mean imputation is one of the most frequently used imputation methods. Basically, it uses the mean of observed values of a variable in place of missing data values for that same variable. Mean imputation enjoys many of the advantages of single imputation mentioned above. For example, it saves a great deal of data that listwise deletion and pairwise deletion eliminate, although mean imputation may not always give reasonable values.

Regression Imputation

Regression imputation replaces missing variables by predicted values from a regression of the missing variables on variables observed for that unit. This regression is usually calculated from units with present. It involves the use of one or more independent variables. The regression formula is built on the cases with complete data and a well-fitted model is established by using complete cases. An error term, which is computed from the complete cases, can be included in the prediction to maintain the underlying variability in the data, thus reflecting the uncertainty of the predicted value.

PROPOSED WORK

This paper proposes a framework that makes absolute plausible values to be replaced with a predictably possible state. The basic architecture of the framework consists of three important elements, characteristic corpus, missing value digest and an association between missing value digest and value to be imputed.

The current work is not a deviation followed by the great researchers of data mining and particularly of imputation techniques, the similar style of taxonomy is formulated, which contains four important stages as shown below:

<table>
<thead>
<tr>
<th>Little and Rubin Taxonomy</th>
<th>Taxonomy in Current Research Work</th>
</tr>
</thead>
<tbody>
<tr>
<td>Analysis of the complete records</td>
<td>Generating Characteristic Weights</td>
</tr>
<tr>
<td>Weighting procedures</td>
<td>Learning and Sampling of database</td>
</tr>
<tr>
<td>Imputation-based procedures</td>
<td>Generating Missing Value Digest Vector</td>
</tr>
<tr>
<td>Model-based procedures</td>
<td>Search CW in Corpus and replace plausible values</td>
</tr>
</tbody>
</table>

The data is preprocessed and the characteristic corpus is generated for all the tuples available. The selected attributes are chosen to generate the characteristic values. The selected attributes are the basis that works as standing criterion throughout the process. A sampling mechanism can be chosen and then executing the process of generating the characteristic values of each
sampled tuple. From the database the records with missing values are selected of those of the attributes that match to one of the standing criterion.

**Information-Hash Theoretic Approach**

Generally Information Theoretic approach follows the information about the attributes that describe their characteristics with more efficacies that is required for solving the problem. Information-hash theoretic approach, deals with the collection and various combination of the attributes that contain maximum frequency of missing values. An attribute in the tuple that is earmarked as having missing value in other tuples is said to be called existent attribute of that tuple. An attribute in the tuple is having missing value is called a missing attribute of that tuple. The combination of these attributes can be set according to the domain interest or the user choice which eases the algorithm complexity. More number of possible combinations of existent attributes is built for making out more number of possibilities to guess the missing value in the tuple. The hashing mechanism adopted in this algorithmic approach proposed generating a quantum figure for describing the characteristics of a tuple with respect to the missing value attributes and existent attributes.

**Generate the Characteristic Corpus**

The characteristic corpus is contained with attribute, value. This is a key mechanism used to replace a particular missing value in the original tuple of the sampled database. This is also used to calculate the *Missing Value Digest*. The Characteristic Weight Corpus is filtered with the attribute and MVD is found. The corresponding value is considered as the missing value and is imputed in the original database.

**Generating MVD**

MVD is generated with various alternatives of attributes in the database. Alternatives of attributes include, single attributes, combination of attributes. Various MVDs are generated for the collection of existent attributes set from the sampled database. MVDs bring out various possibilities that a tuple can be found with various combination of attributes that contain missing values. All the possible MVDs are stored in the characteristic corpus. For the tuple in the database that is generated:

1. Selection of attribute set
2. Frequency count of attribute set
3. Calculate the characteristic weight for each attribute set based on frequency count
4. MVD is calculated based on rank of each characteristic weight of attribute set.
5. **MVD association** is developed.

(a) *Selection of attribute set:* Selection is a subproblem of more complex problems like nearest neighbor problem the and shortest path problems. The term "selection" is used in genetic algorithms in which genomes are chosen from a population for later breeding. Selection can be reduce to sorting, by sorting the
list and then extracting the desired element or set of elements. This method is efficient when many selections need to be made from a large list of items, in which case only one initial, expensive sort is needed, followed by many cheap extraction operations. In general, this method requires $O(n \log n)$ time, where $n$ is the length of the list.

Finding minimum or maximum of the list of numbers is a linear algorithmic approach. Using the same ideas used in minimum/maximum algorithms, we can construct a simple, but inefficient general algorithm for finding the $k^{th}$ smallest or $k^{th}$ largest item in a list, requiring $O(k)$ time, which is effective when $k$ is small. To accomplish this, we simply find the most extreme value and move it to the beginning until we reach our desired index. This can be seen as an incomplete selection sort. Here is the minimum-based algorithm:

(b) **Frequency count of attribute set:** Counting the attribute containing missing value. Either the attribute or the set of attributes that are selected must be present with a missing value, number tuples that contain the attribute or set of attributes with missing values is counted, which is called frequency count of the attribute set. A normal full-scan sequential algorithmic search is performed on the database, in two phases. In the first phase evaluating the attributes and attribute sets in the database. In the evaluation, the existent proof of the attributes is made. The second phase each attribute or attribute set is posted as a class. Frequency count for each class in the database is prepared which becomes frequency count of the class.

(c) **Calculate the characteristic weight for each attribute set based on frequency count:** Calculating the characteristic weight for each attribute or the attribute set is simply a density fixing algorithm. This is fixing a density value for each tuple. By considering the quantitative values for the attributes in the given tuple for each tuple a weight is calculated and assigned as characteristic weights. This is also alternative done by hashing algorithms. Hashing algorithms to calculate characteristic weights consists of three steps. They are:

Step 1: Represent the key in numerical form

Step 2: Fold and Add

Step 3: Divide by a prime number and use the remainder as the address.

(d) **MVD is calculated based on rank of each characteristic weight of attribute set.**

(e) **MVD association is developed.**

$$\text{MVD} \rightarrow \text{Value to be imputed}$$

**IMPLEMENTATION OF THE PROPOSED ALGORITHMIC SIMULATION FRAME WORK FOR COMPUTING THE MISSING VALUES IN AN ARBITRARILY DATASETS**

An MVD is generated, which is a vector that is capable of filling up the missing values with plausible values that match to various dimensions of the standing criteria.
The above framework demonstrates the process of identifying the missing values in the database. Consider the following fragment of data from the database. This is selected by using appropriate sampling methods.

### Table 1
#### Fragmentation of Data

<table>
<thead>
<tr>
<th>recno</th>
<th>attr1</th>
<th></th>
<th>attr3</th>
<th></th>
<th>attr4</th>
<th></th>
<th>attr5</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>a1</td>
<td></td>
<td>b1</td>
<td></td>
<td>c1</td>
<td></td>
<td>d1</td>
</tr>
<tr>
<td>2</td>
<td>a2</td>
<td></td>
<td>b2</td>
<td></td>
<td>c2</td>
<td></td>
<td>d2</td>
</tr>
<tr>
<td>3</td>
<td>a3</td>
<td></td>
<td>b3</td>
<td></td>
<td>c3</td>
<td></td>
<td>d3</td>
</tr>
<tr>
<td>4</td>
<td>a4</td>
<td></td>
<td>NA</td>
<td></td>
<td>c4</td>
<td></td>
<td>d4</td>
</tr>
<tr>
<td>5</td>
<td>a6</td>
<td></td>
<td>b6</td>
<td></td>
<td>c6</td>
<td></td>
<td>d6</td>
</tr>
<tr>
<td>6</td>
<td>a6</td>
<td></td>
<td>b6</td>
<td></td>
<td>c6</td>
<td></td>
<td>NA</td>
</tr>
<tr>
<td>7</td>
<td>a7</td>
<td></td>
<td>b7</td>
<td></td>
<td>c7</td>
<td></td>
<td>NA</td>
</tr>
<tr>
<td>8</td>
<td>a8</td>
<td></td>
<td>b8</td>
<td></td>
<td>c8</td>
<td></td>
<td>d8</td>
</tr>
<tr>
<td>9</td>
<td>a9</td>
<td></td>
<td>b9</td>
<td></td>
<td>c9</td>
<td></td>
<td>d9</td>
</tr>
</tbody>
</table>

### Choice of Attributes

The attributes as a set or singly are selected as the choice for the MVD generation. Missing Value Digest is the hash generated value for a tuple based on the missing attribute. The missing attribute is a set or singly must be of the choice selected.

### Table 2
#### MVD Generation

<table>
<thead>
<tr>
<th>recno</th>
<th>attr1</th>
<th>attr2</th>
<th>attr3</th>
<th>attr4</th>
<th>attr5</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td></td>
<td></td>
<td></td>
<td>d1</td>
<td>e1</td>
</tr>
<tr>
<td>2</td>
<td></td>
<td></td>
<td></td>
<td>d2</td>
<td>e2</td>
</tr>
<tr>
<td>3</td>
<td></td>
<td></td>
<td></td>
<td>d3</td>
<td>e3</td>
</tr>
<tr>
<td>4</td>
<td></td>
<td></td>
<td></td>
<td>d4</td>
<td>e4</td>
</tr>
<tr>
<td>5</td>
<td></td>
<td></td>
<td></td>
<td>d6</td>
<td>e6</td>
</tr>
<tr>
<td>6</td>
<td></td>
<td></td>
<td></td>
<td>NA</td>
<td>e6</td>
</tr>
<tr>
<td>7</td>
<td></td>
<td></td>
<td></td>
<td>NA</td>
<td>e7</td>
</tr>
<tr>
<td>8</td>
<td></td>
<td></td>
<td></td>
<td>d8</td>
<td>e8</td>
</tr>
<tr>
<td>9</td>
<td></td>
<td></td>
<td></td>
<td>d9</td>
<td>e9</td>
</tr>
</tbody>
</table>
The characteristic weights for the tuples that are having attr1, attr2 values, is calculated. The tuples that contain values in the attributes selected as the choice, are called attribute existent.

### Table 3
**List of Characteristic Weights**

<p>| | |</p>
<table>
<thead>
<tr>
<th></th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>Cw1</td>
</tr>
<tr>
<td>2</td>
<td>Cw2</td>
</tr>
<tr>
<td>3</td>
<td>Cw3</td>
</tr>
<tr>
<td>5</td>
<td>Cw4</td>
</tr>
<tr>
<td>6</td>
<td>Cw5</td>
</tr>
<tr>
<td>7</td>
<td>Cw6</td>
</tr>
<tr>
<td>8</td>
<td>Cw7</td>
</tr>
<tr>
<td>9</td>
<td>Cw8</td>
</tr>
</tbody>
</table>

The characteristic table with characteristic weights for the tuples is built. The table is generated for each and every selected attribute (singly or set)

### Table 4
**Characteristic Weight**

<table>
<thead>
<tr>
<th>Attr1, attr3</th>
<th>attr4</th>
</tr>
</thead>
<tbody>
<tr>
<td>cw1</td>
<td></td>
</tr>
<tr>
<td>cw2</td>
<td>x</td>
</tr>
<tr>
<td>cw3</td>
<td></td>
</tr>
<tr>
<td>cw4</td>
<td>x</td>
</tr>
<tr>
<td>cw5</td>
<td></td>
</tr>
<tr>
<td>cw6</td>
<td>x</td>
</tr>
<tr>
<td>cw7</td>
<td></td>
</tr>
<tr>
<td>cw8</td>
<td>x</td>
</tr>
<tr>
<td>cw9</td>
<td></td>
</tr>
</tbody>
</table>

The existent attribute (attr1, attr3) is selected except for cw2, cw5, cw8 and (attr4) is selected except for cw4, cw6 and cw9. The characteristic values are drawn from the tuples of sampled database for each tuple having the missing value and the corresponding characteristic weight value is lookup in the characteristic table. If any characteristic weight is found matched against the attributes in the Characteristic Weight Table, the plausible value is selected from the sampled database according to the matching Characteristic Weight value at the corresponding attribute.

The Missing Value Digest is generated for the pattern of the tuple with the missing values. The Missing Value Digest consists of Characteristic Weights of the tuples that report missing values of the existent attributes. That is cw1, cw3, cw7 are the Characteristic Weight values for the tuples that are having missing values in the attributes (attr1, attr3) and (attr4), but the cw2, cw5, cw8 are for the tuples having missing values in the attributes (attr4) only.

Thus this approach implements easy lookup of plausible values from the MVD Corpus for the tuples in the sampled database.
RESULTS

1. Artificial data simulation with the algorithm is performed on the selected database. The performance of the algorithm is dependent on the data set that is sampled from the database. Algorithm is tested in Java language. Generation of hash digest values is long integer that uniquely identifies the tuples with missing attributes. Representation of characteristic weights as corpus is a flat file that is accessed throughout the experiment i.e., during the lifetime of the algorithm.

2. Relational database is selected as the choice of input for the experiment. Comparative to the single imputation and multiple imputation methods which depend on probability of the values and the distribution of the values in the plausible value set is completely away from this algorithm. In this algorithm, the exact value that is required to be replaced in the missing location is determined shrewdly without any hesitation.

3. The algorithm complexity is amortized with various auxiliary algorithms that are included for performing important operations like sampling, lookup and replacement. The core of the entire algorithm framework is generating the CW values and building the MVD corpus, which helps all the way procedurally to find the replace the plausible values at the missing locations.

CONCLUSION

Unlike the statistical and probabilistic methods, the work has algorithmic base that finds the exact missing value at the location in the tuples. This can be streamlined with implementation of tiered-framework and efficient implementation of the auxiliary algorithms.

References


[3] Ray Hoare, Missing Values–What They are and Why They are Important, HRS Ltd., Copyright (2003).


AN EFFICIENT IMPUTATION METHOD ON LARGE DATA: INFORMATION-HASH-THEORETIC APPROACH

K.Fayaz*, Dr.G.A.Ramachandra**, Prof.P.Md.Akthar ***and A.V.Sriharsha****
Dept. of Computer Science &Technology, S.K. University, Anantapur, India,

ABSTRACT

Missing data causes a variety of problems in data analysis. First, lost data decrease statistical power. Statistical power refers to the ability of an analytic technique to detect a significant effect in a data set. Second, missing data produce biases in parameter estimates and can make the analysis harder to conduct and the results harder to present. To overcome the missing data problem, a Component-based Framework for imputation is proposed, which efficiently searches the most plausible value for replacement (Jin Zhou 2006). The algorithm developed ensures complete imputation, the fast phase of imputation is based on the complexity of 1) finding the missing value entry, 2) select the category of attribute set, 3) generate the characteristic weight, 4) search the characteristic weight, 5) find the location of missing value and 6) replace the plausible value with missing value.

Key words: Missing Values, Imputation, Data Mining, Weight Functions, Component Framework.

1. INTRODUCTION

Missing data is omnipresent in survey research. Although when we collect information for statistical analysis, complete data for all subjects are desired, the possibility that some data will be unavailable should not be ignored.

It may be lost, be costly to obtain or be unusable. When missing data means there is no response obtained for a whole unit in the survey, it is called unit nonresponse. When missing data means responses are obtained for some of the items for a unit but not for other items, it is called item noresponse (Roderick Little and Micheline Kamber 2006). Missing data has to be dealt with before we can do anything meaningful with the dataset. Many statistical problems have been viewed as missing data problems in the sense that one has to work with incomplete data.

Advances in computer technology have not only made previously long and complicated numerical calculations a simple matter but also advanced the statistical analysis of missing data. Missing data usually means lack of responses in the data. It is often indicated by “Don’t know”, “Refused”, “Unavailable” and

* Research Scholar,Dept. of CS&T,S.K.University, Anantapur(A.P.) skim_fayaz70@rediffmail.com , ** Head, Associate Professor, S.K.University, Anantapur(A.P.), *** Professor, Chairma, BOS, Dept. of Statistics, S.K. University, Anantapur, India, ****Srividyanikethan Engg., College., Tirupati, India,
Missing data are problematic because most statistical procedures require a value for each variable (Jiawei Han and Micheline Kamber 2006). When a data set is incomplete, an analyst has to decide how to deal with it. This requires a missing value procedure.

2. RELATED WORK

Data may be missing in any type of study due to any reason. For example, subjects in longitudinal studies often drop out before the study is complete. Sometimes this happens because they are not interested anymore, are not able to find time to participate, died or moved out of the area. Whatever the reason is, the study will suffer from missing-data problem.

Missing data causes a variety of problems in data analysis. First, lost data decrease statistical power. Statistical power refers to the ability of an analytic technique to detect a significant effect in a data set. Also, it is well known that a high level of power often requires a large sample. Thus, it appears that missing data may meaningfully diminish sample size and power.

Second, missing data produce biases in parameter estimates and can make the analysis harder to conduct and the results harder to present. The bias may be either upward or downward, which means the true score may be either overestimated or underestimated.

Imputation: "The action of attributing something, usually a fault or a crime to someone."

Hitherto, there are methods that deal with missingness, they are mean imputation and multiple imputations. Imputation consists of replacing the missing data with values derived from the respondents or from a relationship between the nonrespondents and respondents.

According to Little and Rubin (2002), the mechanisms leading to missing data can be classified into three subgroups:

♦ Missing completely at random (MCAR)
♦ Missing at random (MAR) and
♦ Not missing at random (NMAR).

MCAR means that the missing data mechanism is unrelated to the variables under study, whether missing or observed: a missing response happens purely by chance.

That is \( f(M \mid Y, Q) = f(M \mid Q) \) for all \( Y, Q \).

Library Progress (International). Vol. 28 (No. 2) 2008
AN EFFICIENT IMPUTATION METHOD ON LARGE DATA: INFORMATION-HASH-THEORETIC APPROACH

Let $Y_{obs}$ denote the observed components of $Y$ and let $Y_{mis}$ denote the missing components (Mary Kynn 2006).

3. PROPOSED WORK

This work has been accomplished using information hash theoretic approach. The complete work has been organized as follows: For the experiment in the underlying work, data plays a vital role. Modeling and Selection of datasets is discussed in Section 4. Development of Framework is explained in the succeeding Section 5. Component Framework is developed to achieve the complete implementation of the algorithms of imputation. In the Section 6, The framework is an umbrella of various functions and components that operate under for imputation. The algorithms and various evaluations have been commented in the succeeding sections. Orange tool is used developed for data mining, particularly used for data selection, sampling and widget development for imputation.

4. MODELING THE DATASET

Data Set in the Working Domain

Data Selection is a very important strategy in data mining experiments. As the Data Mining Systems and their prototypes work on huge data sets, for hypothetical testing of algorithms huge data sets may not be available (Rufus Lynn ... 2006). Instead, the characteristics of the data sets are identified and data is generated using an appropriate data generation algorithms. Pseudo-Random Algorithms are used to generate the data sets from predefined repository of data, which contains instances of data sets with all possible characteristics.

Data Set availability for the data mining experiments is of three categories:

- Real Data Set
- Benchmark Data Set
- Synthetic Data Set

The Real Data Set is the transaction data that is available in the field of interest like business applications. Most of the data mining algorithms are used to solve the problems that arise from the "Market-Basket Analysis". Any real time data that belongs to super market where explosive growth of transactions prevail can be considered as this behalf.

The Benchmark Data Set, the data mining algorithms are designed to solve a particular viewpoint of the problem of the "Market-Basket Analysis". The real
time data set which is available requires lot of purification i.e., preprocessing which is more costlier for the data mining workbench. Hence, all the transactions that exhibit most important properties are considered to be as the candidate database for the algorithm design growth and testing (Jiawei Han and Micheline Kamber 2006).

The Synthetic Data Set, the candidate databases are freely available, as for the study purpose, the properties of the data in the database can be assumed and using Random Sampling methods of Statistics, the data can be generated with the required properties to test and algorithm and its implementation.

The exclusive data set is selected for the experiments performed in this work, which contains all the possible models of the data for missing value analysis. The data set selected is already used by various data mining algorithms and their propensity is tested successfully. The data set is also used in this research work for missing value analysis (Ray Hoare 2003, Analyzing data...2001).

This data set includes descriptions of hypothetical samples corresponding to 23 species of gilled mushrooms in the Agaricus and Lepiota Family. Each species is identified as definitely edible, definitely poisonous, or of unknown edibility and not recommended. This latter class was combined with the poisonous one. The Guide clearly states that there is no simple rule for determining the edibility of a mushroom; no rule like "leaflets three, let it be" for Poisonous Oak and Ivy. Missing attribute values: 2480 of them (denoted by "?"), all for Attribute #11 i.e., for stalk root.

stalk-root:
bulbous=b,club=c,cup=u,equal=e, rhizomorphs=z,rooted=r,missing=?

Following is the fragment of the dataset selected, also illustrates the missing value entries in the dataset.

```
p,k,y,e,f,y,f,c,n,b,t,?,k,s,p,w,p,w,o,e,w,v,l
e,b,s,w,f,n,f,w,b,w,e,?,s,s,w,w,p,w,t,p,w,n,g
```

```
e,x,s,n,f,n,a,c,b,o,e,?,s,s,o,o,p,o,o,p,n,v,l
```

```
e,k,s,w,f,n,f,w,b,p,e,?,s,s,w,w,p,w,t,p,w,n,g
```

```
e,k,s,n,f,n,a,c,b,o,e,?,s,s,o,o,p,n,o,p,b,v,l
```

```
p,k,y,e,f,y,f,c,n,b,t,?,k,k,p,p,p,w,o,e,w,v,d
p,f,y,c,f,m,a,c,b,y,e,c,k,y,c,c,p,w,n,n,w,c,d
```

```
e,x,s,n,f,n,a,c,b,y,e,?,s,s,o,o,p,o,o,p,o,v,l
```

```
p,k,y,n,f,s,f,c,n,b,t,?,s,k,p,w,p,w,o,e,w,v,l
```
5. DEVELOPMENT OF FRAMEWORK

The research work taken over currently proposes a framework that makes absolute plausible values to be replaced with a predictably possible state (Alireza Farhangfar et al 2007). The basic architecture of the framework consists of three important elements, characteristic corpus, missing value digest and an association between missing value digest and value to be imputed.

The current work is not a deviation followed by the great researchers of data mining and particularly of imputation techniques, the similar style of taxonomy is formulated, which contains four important stages as shown below:

The data is preprocessed and the characteristic corpus is generated for all the tuples available. The selected attributes are chosen to generate the characteristic values. The selected attributes are the basis that works as standing criterion throughout the process. A sampling mechanism can be chosen and then executing the process of generating the characteristic values of each sampled tuple. From the database the records with missing values are selected of those of the attributes that match to one of the standing criterion.

6. COMPONENTS

The total research work is centered on the component based design of the framework. The framework consists of static and dynamic components that support all the operations of the framework.

*Static components* support in specifying the data and the transformed data into various stages of the processes in the framework. *Dynamic components* support in specifying the processes that are underlying the principles of missing value finding backed by the framework.

**Database Component**

The database component presents the logical view of the database. This is a static component. The stake data of the entire process is presented in the relevant data structure (table) to be given as input to the framework. The data set selected from the mushroom database is made available in the table that is given as input to the framework.

**Attribute Selection Process**

The attribute selection process presents the logic for selecting the relevant attributes. The relevancy of the attribute is known based on the background knowledge of the database domain and the attribute are selected. A sampling mechanism may also be sought for the selection of the group of attributes.
process also classifies the attributes into existent attributes and non-existent attributes. The group of selected sets of attributes is the outcome of this process.

**Characteristic Weight Generation**
This is a process of generating the characteristic weight for the given set of values. The CW can be generated for the selected number of attributes of any tuple from the sampled (stake) database. Each CW is remembered by the candidate key whichever represents the unique data, for performing further search operation to search the CW and appropriately locate the tuple that consists of plausible value for imputation.

**Characteristic Corpus**
This is a static component that stores the CW of all the tuples for all the possible attribute sets. This is a multiversion storage of all CW of all the tuples. A set of values (CW) belonging to the attributes set for all the tuples is a Missing Value Digest.

**Impute**
This is a component that performs exactly imputation, replacing the missing value with the plausible value.

1. The tuples consisting of missing values are selected
2. Attributes set are selected (Attribute Selection)
3. Find the suitable MVD from the Characteristic Corpus
4. Generate the CW for the tuple to perform imputation
5. Seek the CW in the MVD
6. If found the generated CW in MVD then
   Identify the attribute to replace the missing value with plausible value
   End if

7. **FRAMEWORK**

![Imputation Framework using Characteristic Corpus and MVD](image)

*Figure: Imputation Framework using Characteristic Corpus and MVD*
Information-hash theoretic approach:

Generally Information Theoretic approach follows the information about the attributes that describe their characteristics with more efficacies that is required for solving the problem. Information-hash theoretic approach, deals with the collection and various combination of the attributes that contain maximum frequency of missing values. An attribute in the tuple that is earmarked as having missing value in other tuples is said to be called existent attribute of that tuple. An attribute in the tuple is having missing value is called a missing attribute of that tuple. The combination of these attributes can be set according to the domain interest or the user choice which eases the algorithm complexity. More number of possible combinations of existent attributes is built for making out more number of possibilities to guess the missing value in the tuple. The hashing mechanism adopted in this algorithmic approach proposed generating a quantum figure for describing the characteristics of a tuple with respect to the missing value attributes and existent attributes.

**Definition 1:** An *attribute set* is selection of attributes chosen to generate the characteristic weight for the tuple.

**Definition 2:** An *existent attribute* is an attribute that consists of a value in the selected tuple while replacing the missing value in other tuple.

**Definition 3:** A *non-existent attribute* is an attribute that consists of a missing value in the selected tuple for replacing the missing value from other tuple when characteristic weights match.

**Definition 4:** *Characteristic Weight (CW)* is calculated for all tuples of the attributes in the attribute set.

**Definition 5:** *Missing Value Digest (MVD)* is a vector consisting of all Characteristic weights of an attribute-set.

**Definition 6:** Collection of various MVDs (Missing Value Digest) is represented as *Characteristic Corpus*.

**Axiom 1:** Selection of the attributes into the attribute set is based on the frequency of missing values. The attribute containing less number of missing values is selected for the attribute set.

**Sampling**

Sampling methods are classified as either probability or nonprobability. In probability samples, each number of the population has a known non-zero probability of being selected. Probability methods include random sampling.
systematic sampling, and stratified sampling. In nonprobability sampling, numbers are selected from the population in some nonrandom manner. These include convenience sampling, judgment sampling, quota sampling, and snowball sampling.

An optimistic method of sampling is implemented in order to select the sample tuples that contains all tuples having the missing values and adequate number of tuples that propose the suggestion for the plausible values.

8. PROPOSED ALGORITHMS

Generate the Characteristic Corpus:
The characteristic corpus is contained with attribute, value. This is a key mechanism used to replace a particular missing value in the original tuple of the sampled database. This is also used to calculate the Missing Value Digest. The Characteristic Weight Corpus is filtered with the attribute and MVD is found. The corresponding value is considered as the missing value and is imputed in the original database.

Generating MVD:
MVD is generated with various alternatives of attributes in the database. Alternatives of attributes include, single attributes, combination of attributes. Various MVDs are generated for the collection of existent attributes set from the sampled database. MVDs bring out various possibilities that a tuple can be found with various combination of attributes that contain missing values. All the possible MVDs are stored in the characteristic corpus. For the tuple in the database that is generated:

1) Selection of attribute set
2) Frequency count of attribute set
3) Calculate the characteristic weight for each attribute set based on frequency count
4) MVD is calculated based on rank of each characteristic weight of attribute set.
5) MVD association is developed.

1) Selection of attribute set
Selection is a subproblem of more complex problems like the nearest neighbor problem and shortest path problems. The term "selection" is used in genetic algorithms in which genomes are chosen from a population for later breeding. Selection can be reduced to sorting, by sorting the list and then extracting the desired element or set of elements. This method is efficient when many selections need to be made from a large list of items, in which case only one initial, expensive sort is needed.
AN EFFICIENT IMPUTATION METHOD ON LARGE DATA: 
INFORMATION-HASH-THEORETIC APPROACH

followed by many cheap extraction operations. In general, this method requires $O(n \log n)$ time, where $n$ is the length of the list.

Finding minimum or maximum of the list of numbers is a linear algorithmic approach. Using the same ideas used in minimum/maximum algorithms, we can construct a simple, but inefficient general algorithm for finding the $k^{th}$ smallest or $k^{th}$ largest item in a list, requiring $O(k_a)$ time, which is effective when $k$ is small. To accomplish this, we simply find the most extreme value and move it to the beginning until we reach our desired index. This can be seen as an incomplete selection sort. Here is the minimum-based algorithm:

```
function select(list[1..n], k)
  for i from 1 to k
    minIndex = i
    minValue = list[i]
  for j from i+1 to n
    if list[j] < minValue
      minIndex = j
      minValue = list[j]
    swap list[j] and list[minIndex]
  return list[k]
```

2) Frequency count of attribute set:

Counting the attribute containing missing value. Either the attribute or the set of attributes that are selected must be present with a missing value, number tuples that contain the attribute or set of attributes with missing values is counted, which is called frequency count of the attribute set. A normal full-scan sequential algorithmic search is performed on the database, in two phases. In the first phase evaluating the attributes and attribute sets in the database. In the evaluation, the existent proof of the attributes is made. The second phase each attribute or attribute set is posted as a class. Frequency count for each class in the database is prepared which becomes frequency count of the class.

```
function frequencycount(sets_of_attributes s)
  declare count[ length of s];
  for i from 1 to s
    for j from 1 to |D|
      if s[i] exists in tuple[j] of D
        then count[i]++;
  Return
```

Library Progress(International). Vol.28(No.2)2008

275
3) Calculate the characteristic weight for each attribute set based on frequency count.

Calculating the characteristic weight for each attribute or the attribute set is simply a density fixing algorithm. This is fixing a density value for each tuple. By considering the quantitative values for the attributes in the given tuple for each tuple a weight is calculated and assigned as characteristic weights. This is also alternatively done by hashing algorithms. Hashing algorithms to calculate characteristic weights consists of three steps. They are:

**Step 1:** Represent the key in numerical form

**Step 2:** Fold and Add

**Step 3:** Divide by a prime number and use the remainder as the address.

```java
function generatecharacteristicweight(sets_of_attributes s)
declare cw[length of s];
for i from 1 to s
    if maxcount(s) then
        cw[i] = generatehashweight(i);
return;

function maxcount(count)
declare cw_max;
cw_max=0;
for i from 1 to s
    if count[i]>cw_max then
        cw_max=count[i];
return cw_max;

function generatehashweight(i)
declare cw_hash;
cw_hash = // implement hash value generation technique for s[i];
return cw_hash;
```

2) **MVD is calculated based on rank of each characteristic weight of attribute set.**

3) **MVD association is developed.**

9. EVALUATION

In this section, we report a systematic empirical study using real data sets and synthetic data sets. All the experiments were conducted on a PC computer.

*Library Progress(International). Vol.28(No.2)2008*

276
AN EFFICIENT IMPUTATION METHOD ON LARGE DATA: INFORMATION-HASH-THEORETIC APPROACH

running the Microsoft Windows XP Professional Edition operating system, with a 3.0 GHz Pentium 4 CPU, 1.0 GB main memory, and a 160 GB hard disk. Our algorithms were implemented in Java. By default, our method was implemented. The greedy search method with the help of contribution score, and set the order \( q = 1 \) in the correlation based sample quality score.

Artificial data simulation with the algorithm is performed on the selected database. The performance of the algorithm is dependent on the data set that is sampled from the database. Algorithm is tested in Java language. Generation of hash digest values is long integer that uniquely identifies the tuples with missing attributes. Representation of characteristic weights as corpus is a flat file that is accessed throughout the experiment i.e., during the lifetime of the algorithm.

Relational database is selected as the choice of input for the experiment. Comparative to the single imputation and multiple imputation methods which depend on probability of the values and the distribution of the values in the plausible value set is completely away from this algorithm. In this algorithm, the exact value that is required to be replaced in the missing location is determined shrewdly without any hesitation.

The algorithm complexity is amortized with various auxiliary algorithms that are included for performing important operations like sampling, lookup and replacement. The core of the entire algorithm framework is generating the CW values and building the MVD corpus, which helps all the way procedurally to find the replace the plausible values at the missing locations.

**Sampling using “The Orange Data Sampler”**

The File widget is used to select the input dataset. The input dataset for the experiment is presented in various formats. The Orange tool allows the user to specify the data in delimited data format, tabbed data format and any other user defined data formats. The input data set for the experiment is Mushroom (agaricus lepiota) Population Data. The data set consists of 8124 entries. The data set is a benchmark data set for testing the data mining algorithms, whose propensity is huge. The algorithms that have huge propensity means, they produce huge sizes of patterns like classification, clustering. The data set selected contains more illustrious number of entries with all the possible combinations of the characteristics, and particularly the missing values are denoted by “?”. The missing value dataset is chosen with various types of criteria, all the different types of datasets are supplied to the framework to prove the efficiency.
of the framework An example of the criteria built for sampling the dataset is shown below:

<table>
<thead>
<tr>
<th>Data Selection Criteria</th>
</tr>
</thead>
<tbody>
<tr>
<td>Active Condition</td>
</tr>
<tr>
<td>1</td>
</tr>
<tr>
<td>2</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Data In</th>
<th>Data Out</th>
</tr>
</thead>
<tbody>
<tr>
<td>2239 examples</td>
<td>326 examples</td>
</tr>
<tr>
<td>23 attributes</td>
<td>18 attributes</td>
</tr>
</tbody>
</table>

With the criteria specified as above, the size of sample data is 2239 entries. On random sampling the 326 instances are selected as input and 30% percent of instances are produced as output i.e., 98 instances. On the resulting sample of data 98 instances, the algorithm for imputation proposed in the framework is executed.

A best case sampling will develop more number of instances where the strength of criteria is low. A worst case sampling develops less number of instances where the strength of criteria is high therefore the resulting sample will be small.

Empirical Analysis of the Framework

The International Software Benchmarking Standards Group (ISBSG) database is no exception (Ingunn et al. 2001). It has a large fraction of missing data, in some variables more than 40 percent. There are several reasons why observations may have missing values. High data collection cost may cause missing values. The cost of gathering and reporting data from software projects is non-negligible. For example, it is more difficult, and, therefore, costly to collect data on Interfaces and Effort than on Users, Sites and Modules. Therefore, we expected, and did indeed find, that there are more missing values in Interfaces and Effort than in Users, Sites, and Modules (Alireza Farhangfar et al. 2007).

Another reason for missing values is that some values are so called wild values. A wild value is a value we know is untrue.

Using Orange for Sampling

It is easy to select a subset of examples in Orange. The key idea is the use of indices: first construct a list of indices, one corresponding to each example. Then you can select examples by indices, say take all examples with index 3. Or with
AN EFFICIENT IMPUTATION METHOD ON LARGE DATA: 
INFORMATION-HASH-THEORETIC APPROACH

index other than 3. It is obvious that this is useful for many typical setups, such as 70-30 splits or cross-validation.

Orange provides methods for making such selections, such as ExampleTable's select method. And, of course, it provides methods for constructing indices for different kinds of splits. For instance, for the most common used sampling method, cross-validation, the Orange's class MakeRandomIndices prepares a list of indices that assign a fold to each example.

Sampling with Orange is visually mapped as shown below. Several widgets are selected and connected to develop the samples of datasets. The widgets (any interactive graphical object in a document) used for sampling in this research work are: File, Select-Data, Data-Sampler, and Attribute-Statistics.

10. CONCLUSION

Unlike the statistical and probabilistic methods, the work has algorithmic base that finds the exact missing value at the location in the tuples. This can be streamlined with implementation of tiered-framework and efficient implementation of the auxiliary algorithms.

REFERENCES


