Reconstruction of a Complete Dataset from an Incomplete Dataset by PCA (Principal Component Analysis) Technique: Some Results

Sameer S. Prabhune
Assistant Professor & Head- Department Of IT, Shri Sant Gajanan Maharaj College Of Engineering, Shegaon, Dist: Buldana, (M.S.) India.

S.R. Sathe
Professor, Dept Of E & CS, Visvesvaraya National Institute Of Nagpur (M.S.), India.

Summary
Many data analysis applications such as data mining web mining, information retrieval system, require various forms of data preparation. Mostly all this works on the assumption that the data they work is complete in nature, but that is not true! In data preparation, one takes the data in its raw form, removes as much as noise, redundancy and incompleteness as possible and brings out that core for further processing. Indeed, data preparation often presents a less glamorous but in fact a most critical step than other in data analysis applications. It is a data processing technique such that minor data quality adjustments may lead to wrong interpretation that deteriorates the overall effectiveness of any techniques (viz. Data Mining). The input to any algorithm for interpretation is assumed a nice data distribution, containing no missing, inconsistent or incorrect values. But in real world databases, information is missing, incomplete, imprecise, incorrect. For such cases, we introduce the novel idea of conceptual reconstruction, in which we create effective conceptual representations on which the data mining algorithms can be directly applied. The attraction behind the idea of conceptual reconstruction is to use the correlation structure of the data in order to express it in terms of concepts rather the original dimensions. As a result, the reconstruction procedure estimates only those conceptual aspects of the data which can be mined from the incomplete data set, rather than force errors created by extrapolation. We demonstrate the effectiveness of the approach on a variety of real data sets. This paper describes a theory and implementation of a new filter PCA to the **WEKA** workbench, for estimating the complete dataset from an incomplete dataset.

Keyword
Data Preprocessing, Principal Component Analysis, Missing data.

1. Introduction

The issue of missing values (or missing data) has been studied extensively in the statistical and machine learning literature. According to the missing data mechanism, statistician have identified three classes of missing data [11]: missing completely at random(MCAR), missing at random(MCR), and not missing at random(NMAR). MCAR is when the probability of missing a value is the same for all variables, MCR is when the probability of missing a value is only dependent on other variables, and NMAR is when the probability of missing a value is also dependent on the value of the missing variable. MCR has received the most attentions, for which various "imputation" methods have been designed to predict the missing values before building models.

In recent years, a large number of data sets which are available for data mining tasks are incompletely specified. An incompletely specified data set is one in which a certain percentage of the values are missing. This is because the data sets for data mining problems are usually extracted from real world situations in which either not all measurements maybe available or not all the entries may be relevant to a given record. In other cases, where data is obtained from users directly, many users may be unwilling to specify all the attributes because of privacy concerns [1,13]. In many cases, such situations result in data sets in which a large percentage of the entries are missing. This is a problem since most data mining algorithms assume that the data set is completely specified. Common solutions to the missing data problem include the use of imputation, statistical or regression based procedures [1,11] in order to estimate the entries. Unfortunately, these techniques are also prone to estimation errors with increasing dimensionality and incompleteness.This is because when a large percentage of the entries are missing, each attribute can be estimated to a much lower degree of accuracy. Furthermore, some attributes can be estimated to a much lower degree of accuracy than others, and there is no way of knowing a-priori which estimations are the most accurate. A discussion and examples of the nature of the bias in using direct imputation based procedures may be found in [11].We note that any missing data mechanism would rely on the fact that the attributes in a data set are not independent from one another, but that there is some

Manuscript received December 5, 2010
Manuscript revised December 20, 2010

predictive value from one attribute to another. If the attributes in a data
set are truly uncorrelated, then any loss in attribute entries leads to a true loss of information. In such cases, missing data mechanisms cannot provide any estimate to the true value of a data entry. Fortunately, this is not the case in most real data sets, in which there are considerable redundancies and correlations across the data representation. In this paper, we discuss the novel concept of conceptual
reconstruction , in which we express the data in terms of the salient concepts of the correlation structure of the data. This conceptual structure is determined using techniques such as Principal Component Analysis [10]. These are the directions in the data along which most of the variance occurs, and
are also referred to as the conceptual directions. We note that even though a data set may contain thousands of dimensions, the number of concepts in it may be quite small. For example, in text data sets the number of dimensions (words) are over 100,000 but there are often only 200-400 salient concepts [12]. In this paper, we will provide evidence of the fact that even though predicting the data along arbitrary
directions (such as the original set of dimensions) is
fraught with errors, the components along the conceptual directions can be predicted quite reliably. This is because the conceptual reconstruction method uses these redundancies in an optimum way so as to estimate whatever conceptual representations are reliably possible rather than force extrapolations on the original set of attributes. Such a strategy
is advantageous, since it only tries to derive whatever
information is truly available in the data.

1.1 Contribution of this paper

This paper discusses a technique for mining incomplete data sets by exploiting the correlation structure of data sets. We use the correlation behavior in order to create a new representation of the data which predicts only as much information as can be reliably estimated from the data set. This results in a new full dimensional representation of the data which does not have a one-to-one mapping with the original set of attributes. However this new representation reflects the available concepts in the data accurately and can be used for many data mining algorithms, such as clustering, similarity search or classification.

2. Initial Idea Of Conceptual Reconstruction

In order to facilitate further discussion, we will define the percentage of attributes missing from a data set as the in-completeness factor. The higher the incompleteness factor, the more difficult it is to obtain any meaningful structure from the data set. The conceptual reconstruction technique is tailored towards mining massively incomplete data sets for high dimensional problems. As indicated earlier, the attributes in high dimensional data are often correlated. This results in a natural conceptual structure of the data. For instance, in a market basket application, a concept may consist of groups or sets of closely correlated items. A given customer
may be interested in particular kinds of items which
are correlated and may vary over time. However, her conceptual behavior may be much clearer at an aggregate level, since one can classify the kinds of items that she is most interested in. In such cases, even when a large percentage of the attributes are missing, it is possible to obtain an idea of the conceptual behavior of this customer. A more mathematically exact method for finding the aggregate conceptual directions of a data set is Principal Component Analysis (PCA) [10]. Consider a data set with N points and dimensionality d. In the first step of the PCA technique, we generate the covariance matrix of the data set. The covariance matrix is a d * d matrix in which the (i, j)th entry is equal to the covariance between the dimensions
i and j. In the second step we generate the eigenvectors $\{\bar{e}i \ldots \bar{e}d\}$ of this covariance matrix. These are the directions in the data, which are such that when the data is projected along these directions, the second order correlations are zero. Let us assume that the eigenvalue for the eigenvector ei is equal to λi. When the data is transformed to this new axis-system, the value λi is also equal to the variance of the data along the axis $\bar{e}i$. The property of this
transformation is that most of the variance is retained in a small number of eigenvectors corresponding to the largest values of λi. We retain the $k < d$ eigenvectors which correspond to the largest eigen values. An important point to understand is that the removal of the smaller eigen values for highly correlated high dimensional problems results in
a new data set in which much of the noise is removed, and the qualitative effectiveness of data mining algorithms such as similarity search is improved. This is because these few eigenvectors correspond to the conceptual directions in the data along which the non-noisy aspects of the data are preserved. One of the interesting results that this paper will show is that these relevant directions are also the ones along which the conceptual components can be most accurately predicted by using the data in the neighborhood of the relevant record. We will elucidate this idea with the help of an example[1,11].

3. Preliminary Tools knowledge

To implement the PCA filter for the WEKA workbench, we have used the following technologies. These are as follows:

3.1 Weka 3-5-4

Weka is an excellent workbench [4] for learning about machine learning techniques. We used this tool and the package because it was completely written in java and its package gave us the ability to use **ARFF** datasets in our filter. The weka package contains many useful classes which were required to code our filter. Some of the classes from weka package are as follows [4].

weka.core
weka.core.instances
weka.filters
weka.core.matrix.package
weka.filters.unsupervised.attribute;
weka.core.matrix.Matrix;
weka.core.matrix.EigenvalueDecomposition;
etc.

We have also studied the working of a simple filter by referring to the filters available in java [8,9].

3.2 Java

We used java as our coding language because of two reasons:
1. As the weka workbench is completely written in java and supports the java packages, it is useful to use java as the coding language.
2. The second reason was that we could use some classes from java package and some from weka package to create the filter.

3.3 Eclipse 3.1.2

Eclipse is an **IDE** (Integrated Development Environment) for java. For gaining more efficiency in coding we used the **ECLIPSE 3.1.2** which is a freely available **JAVA IDE** [2]. The most important feature of Eclipse is the de-bugger interface where one can practically trace a portion of the code or the whole code.

4. Pseudo Code

This pseudo code is designed to give the user an understanding of the PCA algorithm based on conceptual reconstruction. PCA is a one of the technique for estimating the missing patterns in data of high dimensions [1,10,11].

Step 1
Get input data:-
a. In our implementation we take an ARFF format as input
b. It will be taken as a two dimensional matrix

Step 2
a. Take 1st column as X data item and 2nd column as Y data item
b. Take the mean of each column.
c. Take the deviation of each element with their corresponding mean.
d. Resultant matrix is known as data adjust matrix.

Step 3
a. Calculate the covariance matrix by the formula
C^x*y =(Cij,Cij = cov(Dimi,Dimj))

Step 4
a. Calculate the eigenvector and eigen values. (Use advance math class of java).

Step 5
a. Find the eigenvector with the largest eigen value.
b. The resultant largest value will be the principle component of the data set.
c. Make a set of those eigen vectors whose values is near to the principle component.
d. The resultant set is known as the feature vector.

Step 6
Final step
a. Transpose the feature vector.
b. The resultant is known as row feature vector.
c. Transpose the data adjust matrix.
d. Resultant is known as row data adjust.
e. Multiply these two results
Final data = Row feature vector * Row data adjust.

5. Implementation details

We were using datasets in ARFF format (Weka default format) as an input to this algorithm and the PCA filter [3,6,7]. The filter would then take ARFF dataset as input and find out the missing values in the input dataset represented by ?. Then we would apply the PCA algorithm and estimate the missing values to reconstruct the whole dataset.
We have created a PCA filter class [3,6,7] which is an extension of the SimpleBatchFilter class which is an abstract class [8,9]. The algorithm first of all takes an ARFF format database as input then it converts the whole database into a matrix by inserting a zero at the place of missing value. Then we have calculated the mean of each column and created a new matrix in which each element is a deviation from the mean. This matrix is known as the **data adjust matrix**. In the next step we have created the

covariance matrix with the help of the various classes available in WEKA such as

weka.core.matrix.
weka.core.matrix.Matrix;
weka.core.matrix.EigenvalueDecomposition.

The covariance matrix will lead us to eigenvalues and thus the eigenmatrix. The resultant set or the resultant matrix is known as **feature vector**.
Then we have taken the transpose of the feature vector. This has given us the new matrix known as the **row feature vector.** In this way we have also taken the transpose of **mean data adjusts** matrix known as **row mean data adjust.**
Then we have created a new matrix which stores the result calculated from the multiplication of the following two matrixes
1. Row feature vector
2. Row data adjust.
Thus in this way we have got the new matrix which is having no missing values and thus we can compare this matrix with the original matrix and can fill the missing values.

6. Results

In figure 1 and figure 2 shown below are the results of data reconstruction on Horse colic dataset of UCI machine learning repository [14].

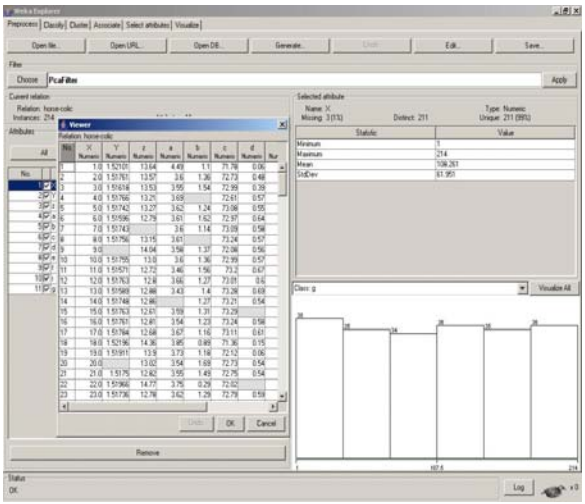

Figure 1: Incomplete Dataset*

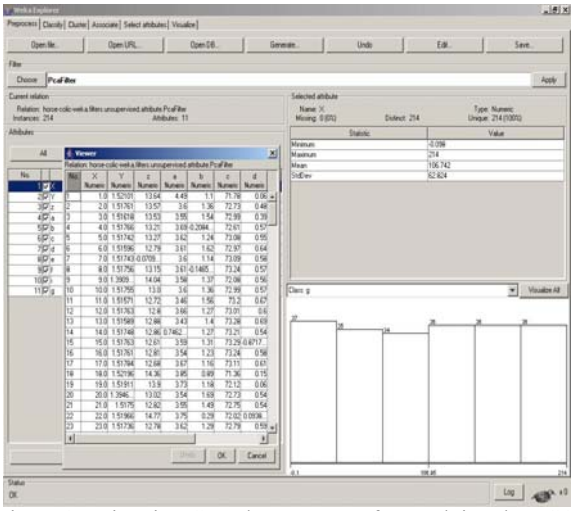

Figure 2 Estimating Complete Dataset after applying the PCA filter*

- UCI Machine Learning Repository of Horse Colic Dataset for 214 Instance[14].

Conclusion

In this paper, we discuss the theory and implementation of new filter, based on conceptual reconstruction i.e. principal component analysis, for estimating the complete dataset from an incomplete dataset. We demonstrate the complete methodology starting from theory to implementation of making the PCA filter by means of available technologies and also addition of this filter as an extension to the WEKA workbench for further analysis.

Acknowledgements
Our special thanks to Mr.Peter Reutemann, of University of Waikato, fracpete@waikato.ac.nz , for providing us the support as and when required.

References

[1] S.Parthsarthy and C.C. Aggarwal, “On the Use of Conceptual Reconstruction for Mining Massively Incomplete Data Sets, “IEEE Trans. Knowledge and Data Eng., pp. 1512-1521,2003.
[2] Eclipse Home Page : http://www.eclipse.org/
[3] http://weka.sourceforge.net/wiki/index.php/Writing_your_own_Filter
[4] wekaWiki link : http://weka.sourceforge.net/wiki/index.php/Main_Page
[5] Ian H. Witten and Eibe Frank , “Data Mining: Practical Machine Learning Tools and Techniques” Second Edition, Morgan Kaufmann Publishers. ISBN: 81-312-0050-7.
[6] http://weka.sourceforge.net/wiki/index.php/CVS
[7] http://weka.sourceforge.net/wiki/index.php/Eclipse_3.0.x
[8] weka.filters.SimpleBatchFilter

[9] weka.filters.SimpleStreamFilter
[10] I.T. Jolliffe. Principal Component Analysis, Springer-Verlag, New York, 1986.
[11] R.J.A.Little, D.Rubin. Statastical Analysis with Missing Data. Wiley Series in Prob. And Stat., 2002.
[12] K. V. Ravikanth, D. Agrawal, A. Singh. "Dimensionality Reduction for Similarity Searching in Dynamic Databases." In ACM SIGMOD Conference, 1998.
[13] R. Agrawal, R. Srikant.," Privacy Preserving Data Mining." In ACM SIGMOD Conference, 2000.
[14] UCI Machine Learning Repository, http://www.ics.uci.edu/umlearn/MLsummary.html

Sameer S. Prabhune received the B.E. and M.E. degrees in Computer Science and Engineering, from SGB Amravati University, Amravati. He is pursuing his Ph.D in CSE from Visvesvaraya National Institute Of Technology, Nagpur. He now with Shri Sant Gajanan Maharaj College Of Engineering, Shegaon, M.S., India. His research interests include Database System, Data Mining and Temporal Database. His current research is on incomplete datasets.

Dr. S. R. Sathe received the B.E.in 1987 and M.Tech. degrees in Computer Science and Engineering from IIT, Bombay in 1989. He completed his Ph.D.in Computer Sci. from Nagpur University in 2004. He is currently working as a Professor in Computer Sci. in the Dept. Of Electronics and computer Science Engineering at Visvesvaraya National Institute Of Technology, Nagpur. His research interests include Database System, Data Mining, Mobile Agent and Parallel Computing.