Supervised Parametric Classification on Simulated Data via Box-Cox Transformation

M. M. Rahman[#1], M. M. Hossain[#2], M. K. Uddin[#3] and A. K. Majumder[#4]

[#1]Dept. of Statistics, Islamic University, Kushtia, Bangladesh, Phone no. +8801718698811
[#2]Dept. of Statistics, Islamic University, Kushtia, Bangladesh, Phone no. +8801716657066
[#3]Dept. of Statistics, Islamic University, Kushtia, Bangladesh, Phone no. +8801714130235
[#4]Dept. of Statistics, Jahangirnagar University, Savar, Dhaka, Bangladesh, Phone no. +8801711145041

ABSTRACT

Most of the classification techniques are developed under the normality assumption. In practical situations data set of course may be non-normal. Hence, we are motivated to apply Box-Cox transformation for transforming non-normal data set to near normal data set. In this paper we consider different parametric classification techniques to classify objects into classes and make a comparative study among these classification techniques to recognize the suitable one for a given situation. There is no unique classification technique that is suitable for all the situations. In most of the situations classification techniques gives few misclassifications under transformed data set. Also, the classification accuracy through Naive Bayes technique is better than the other classification techniques. We also investigate the effect of Box-Cox transformation and observe that, the classification accuracy under transformed data set is higher than the simulated data set.

Key words: Fisher’s Linear Classification, Quadratic Classification, Bayesian Network, Naïve Bayes, Logistic Classification, Parametric Classification, Box-Cox Transformation.

Corresponding Author: M. M. Rahman

INTRODUCTION

Traditionally, there have been two main approaches to classification (Christopher M. Bishop) [1] - *supervised classification* and *unsupervised classification* (usually referred to as segmentation or clustering). In supervised classification we have a set of data samples that have class labels associated with them. This set is called the training data set and is used to estimate the parameters of the classifier. The classifier is then tested on an unknown datasets referred to as the test dataset. Classification is perhaps the most familiar and most popular data mining technique (M. H. Dunham) [2]. Examples of classification applications include image and pattern recognition, medical diagnosis, loan approval, detecting faults in industry applications, and classifying financial market trends. Estimation and prediction may be viewed as types of classification. When someone estimates your age or guesses the number of marbles in a jar, these are actually classification problems. Before the use of current data mining techniques, classification was frequently performed by simply applying knowledge of the data. This is illustrated in the following example. Credit card companies must determine whether to authorize credit card purchases (M. H. Dunham) [2]. Suppose that based on past historical information about purchases, each purchase is placed into one of the four classes: (i) authorize, (ii) ask for further identification before authorization, (iii) do not authorize and (iv) do not authorize but contact police. Here the historical data must be examined first to determine how the data fit into the four classes. Then the problem is to apply this model to each new purchase. Statistical classification is a procedure in which individual items are placed into groups based on quantitative information on one or more characteristics inherent in the items (referred to as traits, variables, characters, etc) and based on a training set of previously labeled items. We still do not have single classifier that can reliably outperform all others on a given data set. The accuracy of a particular parametric classifier on a given data set will clearly depend on the relationship between the classifier and the data (C. M. Van Der Walt and E. Barnard) [3]. By developing statistical classification methods we can asses the performance of the assignment rule, the relative sizes of the classes can be measured formally the differences between classes can also be tested (D. J. Hand) [4].

Classification technique cannot usually provide an error-free method of assignment (R. A. Johnson and D. W. Wichern) [5]. This is because there may not be a clear distinction between the measured characteristics of the populations: that is the groups may overlap. A good classification procedure should result in few misclassifications. In other words the chances or probabilities of misclassification should be small. In practice, labeling large amounts of data may sometimes require considerable human resources or expertise (M. R. Amini and P. Gallinari) [6]. This is for example the case for many information retrieval tasks where the relevance of retrieved information has to be evaluated by a human. For this type of application, although data are usually widely available, the development of labeled datasets is a long and resource consuming process. For other applications like medical diagnosis, labeling datasets may require expensive tests and be therefore very costly. For rapidly evolving domains or databases there is simply no time to process by hand large datasets.

The supervised parametric classification techniques (Fisher's Linear Discrimination, Quadratic Classification, Naïve Bayes, Bayesian Network and Logistic Classification) has been proposed as a solution to this type of problem when simulated data are available and not time consuming. Since there is a belief that simulated data contain relevant information about the class, it is a natural idea to extract this information to provide a classifier more evidence. In this paper, we investigate the performance of different classification techniques and observed that, some of the techniques give few misclassifications under Box-Cox transformed data set than the simulated data set. Hence, the classification accuracy under Box-Cox transformed data set is higher than the simulated data set. In this paper, we also investigate that, there is no unique classification technique that gives better result in all the situations. Different classification technique gives better result in different situation. Considering all the situations, the classification accuracy is achieved by the Naïve Bayes Classification technique is better than the other classification techniques.

The paper is organized as follows: first we present the formal framework of our representation, and then discuss the Box-Cox transformation methods and classification techniques used in this study. We also apply these classification techniques on the simulated data set and transformed data set. Finally, we make a comparative study among the classification techniques used in this study to recognize the effectiveness of Box-Cox transformation and also identify the suitable technique.

BOX-COX TRANSFORMATION METHODS

Some of the methods of classification are developed on the basis of normality assumption (R. A. Johnson and D. W. Wichern) [5]. If the normality assumption is not satisfied then using this type of method is theoretically wrong and gives the misleading results. Then we need to make non-normal data more "normal looking" by considering Box-Cox transformation of the data.

Transforming Univariate Observation

A convenient analytical method is available for choosing a power transformation (R. A. Johnson and D. W. Wichern) [5]. In case of univariate analysis, Box and Cox considers slightly modified family of power transformations.

$$x^{\lambda}=\begin{cases}\dfrac{x^{\lambda}-1}{\lambda}\ ; & \lambda\neq 0\\ \ln x\ ; & \lambda=0\end{cases}\quad \ldots \qquad (1)$$

is continuous in λ *for* $x>0$. Given the observations $x_1,x_2,\ldots,x_n$ the Box-Cox solution for the choice of an appropriate power λ is the solution that maximize the expression

$$l(\lambda)=-\frac{n}{2}\ln\left[\frac{1}{n}\sum_{j=1}^{n}\left(x_j^{(\lambda)}-\overline{x}^{(\lambda)}\right)^2\right]+(\lambda-1)\sum_{j=1}^{n}\ln x_j$$

where $x_j^{(\lambda)}$ is defined in equation (1) and $\overline{x}^{(\lambda)}=\frac{1}{n}\sum_{j=1}^{n}x_j^{(\lambda)}=\frac{1}{n}\sum_{j=1}^{n}\left(\frac{x_j^{\lambda}-1}{\lambda}\right)$ is the arithmetic average of the transformed observations.

Transforming Multivariate Observation

With multivariate observations a power transformation must be selected for each of the variables (R. A. Johnson and D. W. Wichern) [5]. Let $\lambda_1,\lambda_2,\ldots,\lambda_p$ be the power transformations for the *p*-measured characteristics. Each λ_k can be selected by maximizing the equation.

$$l_k(\lambda)=-\frac{n}{2}\ln\left[\frac{1}{n}\sum_{j=1}^{n}\left(x_{jk}^{(\lambda_k)}-\overline{x}^{(\lambda_k)}\right)^2\right]+(\lambda_k-1)\sum_{j=1}^{n}\ln x_{jk}\quad\ldots\qquad(2)$$

where $x_{1k},x_{2k},\ldots,x_{nk}$ are the *n* observations on the k^{th} variable, $k=1,2,\ldots,p$. Here

$\overline{x}_k^{(\lambda_k)}=\frac{1}{n}\sum_{j=1}^{n}x_{jk}^{(\lambda_k)}=\frac{1}{n}\sum_{j=1}^{n}\left(\frac{x_{jk}^{\lambda_k}-1}{\lambda_k}\right)$ is the arithmetic average of the transformed observations.

The j^{th} transformed multivariate observation is $x_j^{(\hat{\lambda})}=\left[\frac{x_{j1}^{\hat{\lambda}_1}-1}{\hat{\lambda}_1}+\frac{x_{j2}^{\hat{\lambda}_2}-1}{\hat{\lambda}_2}+\ldots+\frac{x_{jp}^{\hat{\lambda}_p}-1}{\hat{\lambda}_p}\right]'$

where $\hat{\lambda}_1,\hat{\lambda}_2,\ldots,\hat{\lambda}_p$ are the values that individually maximize the equation (2).

CLASSIFICATION TECHNIQUES

Data mining is a process to mine and organize data in useful and coherent collections (J. Han and M. Kamber; B. A. Aski and H. A. Torshizi) [7, 8]. Data mining is sometimes used to discover and show some knowledge in an understandable form. The aim of data mining is description and prediction. There are many strategies in data mining which can be led to the prediction. One of them is classification. A classification is first trained on a given labelled set of training samples. A given test sample is then assigned to a particular class by the classifier (R. O. Duda, P. E. Hart, D. G. Stork; M. P. Sampat, A. C. Bovik, J. K. Aggarwal and K. R. Castleman) [9, 10].

SUPERVISED PARAMETRIC CLASSIFICATION TECHNIQUES

Classification maps data into predefined groups or classes. It is often referred to as supervised learning because the classes are determined before examining the data (M. H. Dunham) [2]. This section gives a brief review of the different supervised parametric classification techniques that are used in this paper.

Fisher's Linear Classification

Fisher-LDA considers maximizing the following objective (M. Welling) [11]:

$$J(w) = \frac{w^T S_B w}{w^T S_W w}$$

where S_B is the "between classes scatter matrix" and S_W is the "within classes scatter matrix". Note that due to the fact that scatter matrices are proportional to the covariance matrices we could have defined J using covariance matrices the proportionality constant would have no effect on the solution. The definitions of the scatter matrices are:

$$S_B = \sum_c (\mu_c - \bar{x})(\mu_c - \bar{x})^T$$

$$S_W = \sum_c \sum_{i \in c} (x_i - \mu_c)(x_i - \mu_c)^T$$

where, $\bar{x}$ is the overall mean of the data cases. Oftentimes you will see that for 2 classes S_B is defined as $S'_B = (\mu_1 - \mu_2)(\mu_1 - \mu_2)^T$. This is the scatter of class 1 with respect to the scatter of class 2 and hence corresponds to computing the scatter relative to a different vector. By using the general transformation rule for scatter matrices:

$$S_{\mu+v} = S_\mu + Nvv^T + 2Nv(\mu - \bar{x})^T$$

with $S_\mu = \sum_i (x_i - \mu)(x_i - \mu)^T$ we can deduce that the only difference is a constant shift not depending on any relative distances between points. A study concerned with this function maximally separates the two populations and used to classify new observations.

Quadratic Classification

A quadratic discriminant analysis is a general extension of a linear discriminant analysis that assumes the same variance-covariance matrix of different classes (K. M. Lee, T. J. Herrman, S. R. Bean, D. S. Jackson and J. Lingenfelser) [12]. The individual variance-covariance matrix of each class is used as a classification criterion in a quadratic discriminant analysis. Among several alternative classification rules used to discriminate among classes, the Bayes rule was used to compute the posterior probability to assign an observation x to a single class *(G)*. According to this rule, given prior probabilities p_i *and* p_j, the observation x belongs to class G_i, if

$$P(x/G_i).p_i > P(x/G_j).p_j \quad for\ i \neq j$$

where, $P(x/G_i)$ *and* $P(x/G_j)$ are the probability densities. A quadratic discriminant assigns the observation x to class G_i when the discriminant score $D_i(x)$, a measure of the generalized squared distance between x and class *G*, is minimized.

$$D_i(x) = 0.5(x - \mu_i)' \sum_i^{-1} (x - \mu_i) + 0.5\log(|\Sigma_i) - \log(p_i)$$

where, μ_i is the mean of class i, and Σ_i is the population variance-covariance matrix of class G_i. The posterior probability for each of the possible classifications is then obtained using the computed discriminant score $D_i\left(x\right)$. An observation x is assigned to the class with the largest posterior probability. In a linear discriminant analysis, the notation $\sum_i$ of the different population covariance matrix is replaced with $\sum$ due to the same variance-covariance matrix assumption

$$D_i\left(x\right)=0.5\left(x-\mu_i\right)'\sum{}^{-1}\left(x-\mu_i\right)-\log\left(p_i\right)$$

Naive Bayes

Bayes theorem with independent assumptions between predictors is core concept of Naive Bayes classifier. The simplest approach of Bayesian network is naive bayes in which all attribute of a dataset is independent to its class variable value. So, Naive Bayes classifier is a Bayesian network where the class has no parents and each attribute has the class as its sole parent (M. P. Sampat, A. C. Bovik, J. K. Aggarwal and K. R. Castleman; W. Buntine; Daniel Grossman and Pedro Domingos) [10, 13, 14]. The approach is called “Naïve” because if assumes the independence between the various attribute values (M. H. Dunham) [2]. Given a data values x_i the probability that a related tuple, t_i, is in class C_j is described by $P\left(C_j \mid x_i\right)$. Training data can be used to determine $P\left(x_i\right)$, $P\left(x_i \mid C_j\right)$ and $P\left(C_j\right)$. From these values, Bayes theorem allows us to estimate the posterior probability $P\left(C_j \mid x_i\right)$ *and then* $P\left(C_j \mid t_i\right)$.

Given a training set, the Naïve Bayes algorithm first estimates the prior probability $P\left(C_j\right)$ for each class by counting how often each class occurs in the training data. For each attribute, x_i the number of occurrences of each attribute values x_i can be counted to determine $P\left(x_i\right)$. Similarly, the probability $P\left(x_i \mid C_j\right)$ can be estimated by counting how often each value occurs in the class in the training data. Naïve Bayes classification can be viewed as both a descriptive and a predictive type of algorithm. The probabilities are descriptive and are then used to predict the class membership for a target tuple. Suppose that tuple t_i has p independent attribute values $\left\{x_{i1}, x_{i2}, .., x_{ip}\right\}$. From a descriptive phase, we know $P\left(x_{ik} \mid C_j\right)$, for each class C_j and attribute x_{ik}. We then estimate $P\left(t_i \mid C_j\right)$ by $P\left(t_i \mid C_j\right)\prod_{k=1}^{p} P\left(x_{ik} \mid C_j\right)$.

At this point in the algorithm, we then have the needed prior probabilities $P\left(C_j\right)$ for each class and the conditional probability $P\left(t_i \mid C_j\right)$. To calculate $P\left(t_i\right)$, we can estimate the likelihood that t_i is in each class. The posterior probability $P\left(C_j \mid t_i\right)$ is then found for each class. The class with the highest probability is the one chosen for the tuple.

Bayesian Network

A Bayesian network *B* is an annotated acyclic graph that represents a joint probability distribution (JPD) over a set of random variables $\boldsymbol{V}^{\mathbf{11}}$ [15] (F. Ruggeri, F. Faltin and R. Kenett). The network is defined by a pair $B=\langle G,\Phi\rangle$, where *G* is the directed acyclic graph (DAG) whose nodes $X_1, X_2, ..., X_n$ represents random variables, and whose edges represent the direct dependencies between these variables. The graph *G* encodes independence assumptions, by which each variable X_i is independent of its nondescendents given its parents in *G*. The second component Φ denotes the set of parameters of the network. This set contains the parameter $\theta_{x_i|\pi_i} = P_B\left(x_i \mid \pi_i\right)$ for each realization x_i *of* X_i conditioned on π_i, the set of parents of X_i in *G*. Accordingly, *B* defines a unique JPD over ***V***, namely:

$$P_B\left(X_1, X_2, ..., X_n\right) = \prod_{i=1}^{n} P_B\left(X_i \mid \pi_i\right) = \prod_{i=1}^{n} \theta_{X_i|\pi_i}$$

For simplicity of representation we omit the subscript *B* henceforth. If X_i has no parents, its local probability distribution is said to be unconditional, otherwise it is conditional.

Logistic Classification

Logistic regression is a generalization of linear regression. It is basically used for estimating binary or multi-class dependent variables (De Mantaras and E. Armengol; Yugal kumar and G. Sahoo) [16, 17]. It is a well known technique for classification (M. R. Amini and P. Gallinari) [6]. The only distributional assumption with this method is that the log likelihood ratio of class distributions is linear in the observations (3), this assumption is verified by a large range of exponential density families, e.g. normal, beta, gamma, etc.

$$\log\left(\frac{f_1(x)}{f_2(x)}\right) = \beta_0 + \beta^t.x \ \ ... \ \ (3)$$

where, the f_k, $k = \{1,2\}$ are class conditional parametric densities and $\beta = \{\beta\}_{k=0}^{d}$ is the set of parameters of the model. An advantage of such a model is that it gives the posterior probabilities a simple form:

$$p\left(P_1 / x\right) = \frac{1}{1+\exp\left(-\left(\beta_0 + \beta^t.x\right)\right)} \ \ and \ \ p\left(P_2 / x\right) = 1 - p\left(P_1 / x\right) \ ... \ (4)$$

The β 's are trained to optimize the following log-likelihood:

$$L\left(P, \beta\right) = \sum_{k=1}^{2} \sum_{x_i \in P_k} \log\left(p\left(P_k / x_i ; \beta\right)\right) \ ... \ (5)$$

where $\sum_{x_i \in P}$ is a summation over all examples x_i in the partition P_k. Criterion (5) is a convex function of the model parameters (3). The latter are estimated in order to maximize (5), gradient techniques are generally used to this end.

This model could be implemented using a simple logistic unit *G* whose parameters are $\left(\beta_0, \beta\right)$, i.e. $G\left(x\right) = \frac{1}{1+\exp\left(-\left(\beta_0 + \beta^t.x\right)\right)}$. After the estimation of $\beta, G\left(x\right)$ *and* $1 - G\left(x\right)$ are used to estimate $p\left(P_1 / x\right)$ and $p\left(P_2 / x\right)$.

DATA USED IN THIS STUDY

In this section, we generate simulated data set from Uniform, *F* and Gamma distribution. Also, apply Box-Cox transformation methods to transform this data set as normal looking and used as a transformed data set.

RUSULTS AND DISCUSSION

In this section, we apply different parametric classification techniques and discusses about the results.

Table1. Results of Fisher's Linear Classification Technique

		Apparent Error Rate (APER) in %	
Size of the Data Set	Simulated Distributions	Simulated Data Set	Transformed Data Set
100	Group 1: Uniform Distribution Group 2: $F_{75,25}$ Distribution	50.0	10.0
100	Group 1: Uniform Distribution Group 2: $F_{65,35}$ Distribution	11.0	11.0
200	Group 1: Uniform Distribution Group 2: Gamma Distribution	47.0	46.0
500	Group 1: Uniform Distribution Group 2: Gamma Distribution	51.5	49.0

From Table1, we may conclude that, transformed data set performs better than the simulated data set.

Table2. Results of Quadratic Classification Technique

		Apparent Error Rate (APER) in %	
Size of the Data Set	Simulated Distributions	Simulated Data Set	Transformed Data Set
100	Group 1: Uniform Distribution Group 2: $F_{75,25}$ Distribution	50.0	10.0
100	Group 1: Uniform Distribution Group 2: $F_{65,35}$ Distribution	50.0	11.0
200	Group 1: Uniform Distribution Group 2: Gamma Distribution	50.0	41.0
500	Group 1: Uniform Distribution Group 2: Gamma Distribution	45.0	44.0

From Table2, we may conclude that, transformed data set performs better than the simulated data set.

Table3. Results of Bayesian Network Classification

		Apparent Error Rate (APER) in %	
Size of the Data Set	Simulated Distributions	Simulated Data Set	Transformed Data Set
100	Group 1: Uniform Distribution Group 2: $F_{75,25}$ Distribution	11.0	7.0
100	Group 1: Uniform Distribution Group 2: $F_{65,35}$ Distribution	6.0	6.0
200	Group 1: Uniform Distribution Group 2: Gamma Distribution	50.0	50.0
500	Group 1: Uniform Distribution Group 2: Gamma Distribution	50.0	50.0

From Table3 we observed that, Bayesian Network gives better results for simulated data set as well as transformed data set compare with Fisher's Linear Classification, Quadratic

Classification and Logistic Classification. Also gives slightly better results under transformed data set.

Table4. Results of Naive Bayes Technique

		Apparent Error Rate (APER) in %	
Size of the Data Set	Simulated Distributions	Simulated Data Set	Transformed Data Set
100	Group 1: Uniform Distribution Group 2: $F_{75,25}$ Distribution	12.0	12.0
100	Group 1: Uniform Distribution Group 2: $F_{65,35}$ Distribution	11.0	11.0
200	Group 1: Uniform Distribution Group 2: Gamma Distribution	42.0	41.0
500	Group 1: Uniform Distribution Group 2: Gamma Distribution	46.0	42.5

From Table4 we observed that, Naïve Bayes Classification gives better results for simulated data set as well as transformed data set compare with Fisher’s Linear Classification, Quadratic Classification but not constantly Bayesian Network Classification. Consider all the situations we also observed that, Naïve Bayes Classification technique performs slightly better than the Bayesian Network Classification technique and gives slightly better results under transformed data set.

Table5. Results of Logistic Classification Technique

		Apparent Error Rate (APER) in %	
Size of the Data Set	Simulated Distributions	Simulated Data Set	Transformed Data Set
100	Group 1: Uniform Distribution Group 2: $F_{75,25}$ Distribution	11.0	11.0
100	Group 1: Uniform Distribution Group 2: $F_{65,35}$ Distribution	18.0	18.0
200	Group 1: Uniform Distribution Group 2: Gamma Distribution	47.0	46.0
500	Group 1: Uniform Distribution Group 2: Gamma Distribution	51.5	50.0

From Table5 we observed that, Logistic Classification gives better results for simulated data set as well as transformed data set compare with Fisher’s Linear Classification and Quadratic Classification but not Naïve Bayes Classification. Also gives slightly better results under transformed data set.

SUMMARY AND CONCLUSION

In this section, we discuss the results and investigate the performance of different parametric classification techniques. Hence make a comparative study among the techniques to identify the effectiveness of Box-Cox transformation. Also identify the appropriate technique in a given situation.

In this analysis, we investigate that transformed data set step up the classification techniques and reduces the apparent error rate. The apparent error rate for first simulated data set under Fisher’s Linear Classification technique is 50%, whereas 10% for the transformed data set (see, Table1). We also apply Quadratic Classification technique and observed that the apparent error rate of first two simulated data set under Quadratic Classification technique is 50%, whereas 10% and 11% for the transformed data set respectively (see, Table2). Under the Bayesian Network Classification technique the apparent error rate of first simulated data set is 11%, where as 7% for the transformed data set. Hence we also investigate that Bayesian Network and Naïve Bayes gives better results for simulated data set as well as transformed data set compare with Fisher's Linear Classification, Quadratic Classification and Logistic Classification (see, Table3 & Table4). Also, we investigate that, in most of the situations Logistic Classification gives better results for simulated data set as well as transformed data set compare with Fisher's Linear Classification and Quadratic Classification (see, Table5).

In this paper, we also observed that transformed data set significantly reduce the apparent error rate under Fisher's and Quadratic classification technique, where as Bayesian Network, Naïve Bayes and Logistic classification techniques slightly reduce the apparent error rate and gives better results than the former two methods. We observed that, Logistic Classification technique performs better than Fisher's and Quadratic classification techniques, where as Bayesian Network gives better results than Logistic classification in some situation and vice versa. We also observed that, in most of the situations Naïve Bayes classification technique comparatively gives better results than the other parametric classification techniques.

From the above results, we observed that there is no unique classification technique, gives better results in all the situations. We also observed that, transformed data set gives better results than simulated data set for all classification rules used in this study "Thus we may conclude that, in case of performing classification techniques, it is suggested to apply the Box-Cox Transformation if the data set is non-normal. Also, suggested to apply Naïve Bayes classification technique to classify objects".

REFERENCES

[1] Christopher M. Bishop; Neural Networks for Pattern Recognition. Oxford University Press, 1995.

[2] M. H. Dunham; Data Mining Introductory and Advanced Topics, Pearson Education (Singapor) Pte. Ltd, 2003.

[3] C. M. Van Der Walt and E. Barnard; Data Characteristics that Determine Classifier Performance, *South African Institute of Electrical Eng.*, Vol. 98(3), pp. 87-93 Sept. 2007. *(Available at http://www.saiee.org.za/publications/2007/Sept/98_3_3.pdf)*

[4] D. J. Hand; Discrimination and Classification, John Wiley and Sons, New York, 1981.

[5] R. A. Johnson and D. W. Wichern; Applied Multivariate Statistical Analysis, 5th ed., Pearson Education (Singapor) Pte. Ltd., 2002.

[6] M. R. Amini and P. Gallinari; Semi-Supervised Logistic Regression, *Computeer Science Laboratory of Paris, University of Pierre et Marie Curie, Paris*.
(Available at http://www.yaroslavvb.com/papers/amini-semi.pdf)

[7] J. Han and M. Kamber; Data Mining: Concepts and Techniques, Elsevier Science & Technology Books, 2006.

[8] B. A. Aski and H. A. Torshizi; The Use of Classification Techniques to Enrich e-Learning Environments, *Online Proceedings of the University of Salford Fifth Education in a Changing Environment Conference Critical Voices,* pp. 135-39, Critical Times September 2009.
(Available at http://www.ece.salford.ac.uk/cms/resources/uploads/File/Paper%2012.pdf)

[9] R. O. Duda, P. E. Hart, D. G. Stork, Pattern Classification, 2nd Edition, Wiley-Interscience, San Diego, 2001.

[10] M. P. Sampat, A. C. Bovik, J. K. Aggarwal and K. R. Castleman, Supervised Parametric and Non-Parametric Classification of Chromosome Images. *Preprint submitted to Elsevier Science*, 18 October 2004.
(Available at http://live.ece.utexas.edu/publications/2005/mps-2005-patrec-mfish.pdf)

[11] M. Welling; Fisher Linear Discriminant Analysis, *Dept. of Computer Science, University of Toronto, Canada, This is a note to explain Fisher Linear Discriminant Analysis. (Available at http://www.cs.huji.ac.il/~csip/Fisher-LDA.pdf)*

[12] K. M. Lee, T. J. Herrman, S. R. Bean, D. S. Jackson and J. Lingenfelser; Classification of Dry-Milled Maize Grit Yield Groups Using Quadratic Discriminant Analysis and Decision Tree Algorithm, *AACC International, Inc.*, Cer. Che. 84 (2), pp.152-161, 2007.
(Available at http://www.ars.usda.gov/SP2UserFiles/Place/54300510/2007% 20-%20Classification %20of%20Dry-Milled%20Maize%20Grit.pdf.)

[13] W. Buntine; Theory refinement on Bayesian networks. *In B. D. D'Ambrosio, P. Smets, & P.P. Bonissone (Eds.), In Press of Proceedings of the Seventh Annual Conference on Uncertainty Artificial Intelligent*, pp. 52-60, San Francisco, CA, 1991.

[14] Daniel Grossman and Pedro Domingos; Learning Bayesian Network Classifiers by Maximizing Conditional Likelihood. *In Press of Proceedings of the 21st International Conference on Machine Learning*, Banff, Canada, pp. 1-8, 2004.

[15] F. Ruggeri, F. Faltin and R. Kenett; Bayesian Networks, *Encyclopedia of Statistics in Quality & Reliability*, Wiley and Sons, p.1-2, 2007.
(Available at http://www.eng.tau.ac.il/~bengal/BN.pdf)

[16] De Mantaras and E. Armengol; Machine learning from example: Inductive and Lazy methods, *Data and Knowledge Engineering* 25: 99-123, 1998.

[17] Yugal kumar and G. Sahoo, Analysis of Parametric and Non Parametric Classifiers for Classification Technique using WEKA, *I.J. Information Technology and Computer Science*, 2012, 7, 43-49. *(Available at http://www.mecs-press.org/)*