Meta Analysis of Classification Algorithms for Pattern Recognition

So Young Sohn

Abstract—Various classification algorithms became available due to a surge of interdisciplinary research interests in the areas of data mining and knowledge discovery. We develop a statistical meta-model which compares the classification performances of several algorithms in terms of data characteristics. This empirical model is expected to aid decision making processes of finding the best classification tool in the sense of providing the minimum classification error among alternatives.

Index Terms—Data mining, meta analysis, logit model, multivariate statistics.

1 Introduction

Large scale real-time inspection data become available due to advanced technology in scientific sensing instruments and computer systems. Data mining is the process of extracting valid, previously unknown, and ultimately comprehensible information from such a large database so as to make significant contributions to crucial management decision processes [8]. Detailed process for the knowledge discovery and data mining can be explained by the following chained activities: selection of target data, preprocessing of the selected data, transformation of preprocessed data, pattern recognition from transformed data, and knowledge extraction from mined patterns. Related fields to each of knowledge discovery and data mining process are databases, on-line analytic processing, data warehousing, visualization, pattern recognition, and expert systems.

A particular concern of this paper among these related fields is several classification algorithms used for pattern recognition in the process of data mining. The topic of classification has attracted many researchers in wide application areas, such as marketing [8], real-time quality inspection [12], maintenance engineering [22], [23], [25], [26], [27], [28], and medical science [16], to name a few. Classification is the assignment of objects to one of several predefined classes based on the associated individual features. Typically, classification functions or rules are established based on randomly selected training samples from each class or group and are applied to test samples to evaluate their classification accuracy. The construction of a classification procedure from a set of data for which the true classes are known is termed supervised learning in order to distinguish it from unsupervised learning. Numerous algorithms developed for classification purposes can be largely divided into three areas: statistical approach, machine learning, and neural networks. Performance of each algorithm would be closely related to the characteristics of the data to be mined. Therefore, careful consideration of data characteristics is essential for selecting the proper classification algorithm for effective data mining. But, the choice of classification algorithm can depend on individual analyzer's preference and background rather than a scientific guideline.

Few efforts were made to compare various classification methods within a limited selection [1], [17]. Among them, a recent STATLOG project [17] compared the most extensive selection of several classification algorithms based on some empirical data sets and provided a meta-level machine learning rule on the algorithm selection. (Note that STATLOG is an acronym for an ESPRIT project (1990-1993) involved in comparative testing of Statistical and Logical machine learning algorithms.)

However, this meta-level rule based on C4.5 [18] has some drawbacks: It may not appeal to those who are not quite familiar with machine learning approaches; it is multidimensional and the presentation of the rule is not comprehensive; it evaluates each algorithm separately and simply sentence if a particular algorithm can be applicable to a given data set or not; and the reliability of the rule-based advice cannot be assessed. This kind of meta-level rule may not be helpful when we need the ranking of classification algorithms or the best one among them.

In this paper, we propose a statistical meta-model to predict the expected classification performance of each algorithm as a function of data characteristics given. This information can then be used to find the relative ranking of classification algorithms.

We first briefly review classification algorithms to be compared. Next, we introduce a statistical meta-model for classification performances of the algorithms compared. Finally, results are discussed.

2 Review of Classification Algorithms

In this section, we review classification algorithms we intend to compare in three areas: traditional statistical approach, neural nets, and machine learning. Traditionally,

• *The author is with the Department of Computer Science and Industrial Systems Engineering, Yonsei University, Shichondong 134, Seoul, Korea. E-mail: sohns@bubble.yonsei.ac.kr.*

Manuscript received 5 Mar. 1998; revised 18 June 1999.
Recommended for acceptance by A. Webb.
For information on obtaining reprints of this article, please send e-mail to: tpami@computer.org, and reference IEEECS Log Number 107640.

parametric statistical approaches such as a discriminant analysis [11] have been extensively used to classify one group from others based on the associated individual characteristics (or features). The main assumption required for the discriminant analysis is that these features follow a multivariate normal distribution with distinct means for each group and a common variance-covariance matrix. When this common variance-covariance matrix assumption is met for k groups, Fisher's linear discriminant function (DISC) is used for classification. When this assumption is violated, instead of a linear discriminant function, a quadratic discriminant function (QDISC) is estimated based on an individually estimated variance-covariance matrix.

Due to a certain distributional assumption required for the features of discriminant analysis, many authors have used a logistic discriminant function (LOGID) [10] or a nonparametric classification method such as K-nearest neighborhood (KNN) [32].

In logistic regression, the maximum likelihood estimation is used to find the probability of classifying a test item to one group as a function of associated individual features, where this probability is considered as a parameter of Bernoulli or multinomial distribution depending on the number of classes. Nonparametric methods do not make any distributional assumptions and classify a test case based on the training samples in the neighborhood of the item in terms of associated individual features. One of the most popular nonparametric methods is voting K-nearest neighbor method. The KNN method classifies a test case into the class that supplies the largest number of neighbors among the K-nearest neighbors of the case.

In recent years, artificial neural nets (ANN) have been suggested as an alternative methodology for classification to which traditional statistical techniques have long been applied [3]. Some examples of the supervised learning networks are Multilayer Perception (MLP) and Radial Basis Function networks (RBF). MLPs are general-purpose, flexible, nonlinear models that, given enough hidden neurons and enough data, can approximate virtually any function to any desired degree of accuracy. In the MLP, the net input to the hidden layer is a linear combination of inputs as specified by the weights. MLPs are usually trained by an algorithm called the generalized delta rule, which computes derivatives by a simple application of the chain rule called backpropagation (BACK). In RBF network, the hidden neurons compute radial basis functions of inputs, which are similar to kernel functions in kernel regression [31].

Unsupervised learning explores the structure of data without guidance in the form of class information. The clusters found offer a model of the data in terms of cluster centers, sizes, and shapes. Iterative clustering algorithms, such as Kohonen network, are often used. In the Kohonen network, unsupervised competitive learning constructs binary features. Each binary feature represents a subset or cluster of the observations. In the network, only one output neuron is activated with an output of 1, while all the other output neurons are forced to be 0. Learning Vector Quantifizer (LVQ) is also based on a Kohonen net and the essential difference is that LVQ uses supervised learning.

At the same time, a great deal of classification literature has been published in the area of decision tree and rule based learning techniques [4], [6], [30]. Decision tree learning follows the general-to-specific approach, while most of the rule-based learning takes specific-to-general approach.

Decision trees are usually constructed beginning with the root of the tree and proceeding down to its leaves. This approach can be used for predictive or descriptive purposes. Decision tree induction is free from parametric structural assumptions that most statistical induction methods, such as discriminant analysis, are based on. CART is a binary decision tree algorithm [2] which has exactly two branches at each internal node. IndCART differs from CART in using a different way of handling missing values, in not implementing a regression part of CART, and in the different pruning settings. Baytree is a Bayesian approach to decision trees [5] requiring the specification of prior class probabilities and a probability model for the decision tree. CART and Quinlan's C4.5s [18] pruning starts with growing a tree that is much too large. The methods adopted for CART and C4.5 for backward pruning follow different approaches.

More extensive review of classification tools is given in [17].

3 Statistical Meta-Model for Classification Algorithms

In STATLOG project, several classification algorithms were applied to 22 sets of actual data. Classification results were summarized in terms of the classification error associated with and the time taken for training, as well as test data sets. Three data sets have cost information involved in misclassification, instead of the classification error itself. So, omitting those three which are not comparable, we reuse the remaining 19 study results of the classification error rate for test data sets.

These 19 sets of data encompass various data characteristics and can be categorized as credit (*cred.man*, *cr.aust*), image (*dig44*, *kl*, *vehicle*, *letter*, *chrom*, *satim*, *segm*, *cut50*, *cut20*), and other (*shuttle*, *diab*, *dna*, *tech*, *belg*, *belg2*, *faults*, *tsetse*), depending upon the nature of each data set.

A brief description about these data set is given in the Appendix. These data sets were summarized in terms of all or part of the following basic descriptive, as well as multivariate statistics:

N: total number of observations in the whole data set;
r: number of observations used for training;
t: number of observations used for test;
p: number of feature variables;
b: number of binary feature variables;
k: number of classes;
SD: geometric mean ratio of the pooled standard deviations to standard deviations of the individual populations which can be obtained as $exp(M/p\sum_{i=1}^{k}(n_i-1))$, where $M=\gamma\sum_{i=1}^{k}(n_i-1)log\mid S_i^{-1}S\mid$; n_i is the number of observations in class i; S_i and S are the unbiased estimators of the

TABLE 1
Descriptive Statistics of Data Sets

Variable	Mean	Std Dev	Minimum	Maximum
N	12585	14483	570.00	58000
r	8077	9906	513.00	43500
t	4758	5097	570.00	20000
p	33.18	37.22	7.00	180.00
b	11.39	39.69	0	180.00
k	10.54	19.61	2.000	91.00
SD	1.54	0.68	1.03	4.00
$CORR$	0.23	0.15	0.05	0.60
$CANCOR1$	0.80	0.16	0.53	0.99
$FRACT1$	0.66	0.34	0.15	1.00
$SKEW$	1.83	1.85	0.18	6.72
$KURT$	21.84	42.48	-0.34	157.31
HC	1.94	1.51	0.29	4.88
HX	3.78	1.71	0.37	6.55
MCX	0.32	0.34	0.02	1.31
b/p	0.11	0.23	0	1.00
p/N	0.01	0.02	0.000155	0.079
p/t	0.09	0.18	0.000621	0.79
$ERATE$	0.13	0.11	0.000200	0.495

ith class covariance matrix and pooled covariance matrix, respectively; and $\gamma = 1 - \frac{2p^2+3p-1}{6(p+1)(k-1)} \sum \frac{1}{n_i-1} - \frac{1}{n-k}$.

$CORR$: mean absolute correlation coefficients between two features;

$CANCOR1$: first canonical correlation between a linear combination of class variables and a linear combination of features;

$FRAC1$: proportion of total variation explained by the first canonical discriminant;

$SKEW$: mean skewness of features;

$KURT$: mean kurtosis of features;

HX: average entropy of discrete features is defined as $p^{-1}\sum_l(-\sum_j q_{lj} log_2 q_{lj})$, where q_{lj} is the probability that lth feature variable takes on the jth value. For continuous features, one can discretize all numerical data into equal length intervals and apply the same definition.

HC: entropy of classes is defined as $-\sum_i \pi_i log_2 \pi_i$ where π_i is the prior probability for class i;

HCX_l: joint entropy of a class variable and lth attribute is defined as $-\sum_{ij} p_{ij} log_2 p_{ij}$, where p_{ij} denotes the joint probability of observing class i and the jth value of lth attribute variable;

MCX: mutual information of class and feature $HC/p + HX - \sum_l HCX/p$.

Out of 19 data sets, one (*tech*) data set does not have associated information concerning the average correlation of feature variables ($CORR$). Summary statistics of these descriptive measures of the remaining 18 data sets are given in Table 1. Details about the whole features of these data sets are given in [17].

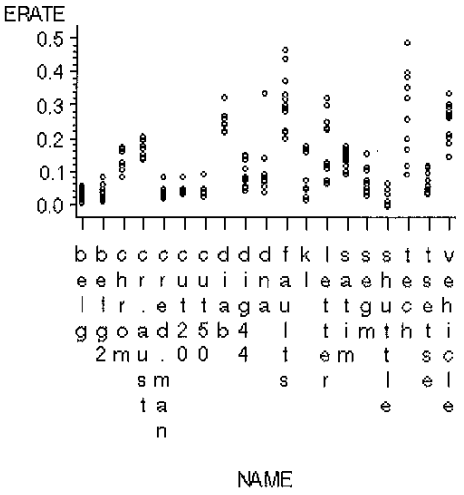

Fig. 1. Distribution of classification error rates dues to use of different algorithms.

In developing a statistical meta-model for classification algorithms, we use 11 classification algorithms described in the earlier section: statistical methods (DISC, QDISC, LOGID, KNN), neural network methods (BACK, LVQ, KOHONEN, RBF), and machine learning methods (indCART, C4.5, BAYTREE). This selection is mainly due to their relative popularity and availability of performance data provided in the empirical study. Despite the popularity, CART is eliminated due to the frequent missing classification performance information (classification error rate) provided in [17]. For instance, the performance of CART was reported only 12 times compared to 17 to 18 observations available for the selected 11 algorithms.

We plot the observed performances of these classification algorithms with respect to their classification errors for each test data set, as shown in Fig. 1. It clearly indicates that the classification error rate (ERATE) of the same data set could vary widely depending upon what kind of classification algorithm is applied to it.

In the statistical meta-model, we take a logit transformation on the classification error in order to ensure the fitted rate be within 0 to 1. This transformed response variable is then related to the study characteristics described in Table 1.

That is, consider the following regression model:

$$y_{ij} = ln[e_{ij}/(1-e_{ij})] = \beta 0_j + \beta 1_j x 1_i + , \ldots , + \beta m_j x m_i + \epsilon_{ij}, \quad (1)$$

where e_{ij} is the classification error obtained from data set i ($i = 1, \ldots, n_j (= 17 \text{ or } 18)$) using classification algorithm j ($j = 1, \ldots, 11$), $0 < e_{ij} < 1$; $x1_i, \ldots, xm_i$ are characteristics of data set i; n_j is the number of data sets with all m data characteristics to which each classification algorithm j is applied; and ϵ_{ij} is assumed to follow an $N(0, var(\epsilon_{ij}))$. $var(\epsilon_{ij})$ is proportional to $1/(t_i e_{ij}(1-e_{ij}))$ and t_i is the number of observations used in data set i to find the test classification error.

When fitting model (1), we use the weighted least square estimator by taking the weight as the inverse of variance of each estimated classification error. That is, $w_{ij} = t_i e_{ij}(1 - e_{ij})$. This is to take into account the heterogeneity in the variance of the estimated classification error.

4 Model Estimation

The main purpose of the statistical meta-analysis is to provide a tool by which one can compare the performance of a classification model in terms of the characteristics of empirical data.

In order to fit such a meta-model for the classification accuracy, instead of using all parameters described in the earlier section as they are, we take some transformation. For instance, we add $pb = b/p$ and $pt = p/r$ to the list of feature variables while eliminating N, b, p, and r from it. This is in view of the fact that the number of binary feature variables (b) or the total number of feature variables (p) does not mean much unless it is compared to the number of training cases (r) applied to each classification algorithm. Many researchers consider N as an important characteristic. But, it is very rare to use all N observations for the purpose of data mining. Next, we plot y_{ij} against each data characteristic summarized in Table 1. Based on this preliminary analysis, some variables are further transformed and are added to the model. Examples of such variables are $CORR2 = CORR \times CORR$, $SKEW2 = SKEW \times SKEW$, $HX2 = HX \times HX$, $pb2 = (b/p)^2$, and $spr = \sqrt{p/r}$. Variables such as $FRAC1$, $KURT$, and $CANCOR1$ turn out to be highly correlated to $CANCOR1$, $SKEW$, and MCX, respectively, and are omitted from the model fitting.

Since we have only a few observations (17 to 18) to fit the model with many candidate variables representing the data characteristic, we use Bootstrap resampling approach [7] to evaluate significant data characteristic variables. That is, we fit model (1) first with the following 12 data characteristics: $SD, CORR, CORR2, CANCOR1, SKEW, SKEW2, HC, HX, HX2, K, pb2, spr$ for $i = 1, \ldots, n_j$ ($= 17$ *or* 18) and $j = 1, \ldots, 11$.

Estimated parameters such as $(\hat{\beta}0_j^*, \ldots, \beta\hat{1}2_j^*)$ and $\hat{var}(\epsilon_{ij})^*$ are treated as true parameters. Next, we generate new y_{ij}^*s from

$$y_{ij}^* = ln[e_{ij}/(1 - e_{ij})] = \hat{\beta}0_j^* + \hat{\beta}1_j^* x1_i +, \ldots, +\hat{\beta m}_j^* xm_i + \epsilon_{ij}, \tag{2}$$

where $\epsilon_{ij} \sim N(0, \hat{var}(\epsilon_{ij})^*)$.

This model is fitted again and a new set of $\hat{\beta}0_j, \hat{\beta}1_j, \ldots, \hat{\beta m}_j$ is obtained. When this step is repeated several times, one can assess the $(1 - \alpha)100$ percent bootstrap intervals for $\beta 0_j, \beta 1_j, \ldots, \beta m_j$ based on both $(\alpha/2)100$th and $(1 - \alpha/2)100th$ percentiles of $\hat{\beta}_j$s. By checking if each 95 percent bootstrap interval for each β contains 0 or not, we found significant data characteristics on the classification performance βs at $\alpha = 0.05$, as summarized in Table 2.

In general, a data characteristic associated with a significantly positive $\hat{\beta}$ has a negative impact on the performance of a classification algorithm since it represents the log odds ratio of the classification error rate with respect to associated data characteristic.

We do not observe any specific outliers in residuals based on the fitted model. Additionally, a leaving-one-out resampling approach is applied to estimate the average prediction error of the meta-model. That is, we first fit the meta-model after deleting one observation (one classification algorithm-data set pair). Next, the deleted case is predicted based on the fitted model. We repeat this procedure for all data and calculate the overall average prediction error of $\hat{e}_{(ij)} = exp(\hat{y}_{(ij)})/(1 + exp(\hat{y}_{(ij)}))$ as follows: $\sqrt{\sum_{j=1}^{11} \sum_{i=1}^{n_j} (e_{ij} - \hat{e}_{(ij)})^2 / \sum_{j=1}^{11} n_j}$, where $\hat{y}_{(ij)}$ is fitted without the ijth case. The average prediction error of $\hat{e}_{(i)j}$ for each classification algorithm j can then be obtained as $\sqrt{\sum_{i=1}^{n_j} (e_{ij} - \hat{e}_{(i)j})^2 / n_j}$. These results are given in Table 3 and the overall average prediction error turns out to be 0.018 compared to the average error rate of observed values, 0.1254. This can be considered as relatively low error. Therefore, we make the following general inferences on the relationship between the data characteristics and the performance of each classification algorithm.

Departure from the homogeneity of covariance matrices among classes (SD) was expected to have a negative impact on the performance of DISC. But, the empirical evidence indicates that it is insignificant on the performance of DISC. Even classification accuracy of some algorithms such as Baytree, C4.5, KNN, LOGID, and LVQ appears to improve as the degree of homogeneity increases.

As the first canonical correlation increases ($CANCOR1$), we expected that the performances of most classical algorithms would increase. It seems that this applies to some neural net classification algorithms (BACK, RBF). But, this characteristic turns out to have insignificant impact on the classification performances of both statistical and machine learning algorithms.

Data characteristics related to normal distribution, such as $SKEW$ and $KURT$, were expected to have influence on the statistical classification algorithm, such as DISC, because it requires a multivariate normality assumption on the feature variables. From the fitted model, it is interesting to observe that the classification error rate of both logistic and quadratic discriminant function (LOGID, QDISC) decrease as $SKEW$ exceeds a certain value.

Unlike the canonical correlation, the entropy of classes (HC) was expected to have negative effect on classification algorithms in general, since it represents indirectly the number of effective groups to be classified neglecting very infrequent classes. While most classification algorithms appear to be sensitive to the change of the entropy of classes, performances of some of the statistical (QDISC) approaches appear not to be affected by it.

The entropy of feature variables (HX) is expected to have a positive effect on the classification performances, in general. However, at 5 percent significance level, effects of

TABLE 2
Meta Classification Model with 95 Percent Bootstrap Interval

Data Characteristic x_i	Classification Algorithm j	Percentile 2.5th	of 50th	$\hat{\beta}_j$ 97.5th
Intercept	back	3.6448	-9.6533	16.9430
	baytree	-5.9818	-16.1352	4.1716
	c4.5	-5.5598	-15.0454	3.9259
	disc	-3.1401	-11.5046	5.2244
	indcart	-6.3916	-15.9715	3.1883
	knn	-10.5834	-19.6283	-1.5385
	koho	-3.4270	-19.8411	12.9871
	logid	-6.9738	-15.8605	1.9130
	lvq	-8.5532	-17.4751	0.3687
	qdisc	-3.9458	-12.2623	4.3706
	rbf	-5.6016	-13.9633	2.7601
SD	back	-0.9960	-2.6766	0.6847
	baytree	-1.7473	-3.3580	-0.1366
	c4.5	-1.5438	-3.0069	-0.0806
	disc	-1.1661	-2.5010	0.1689
	indcart	-1.5633	-3.0972	-0.0295
	knn	-1.4275	-2.6978	-0.1572
	koho	0.8798	-3.5376	5.2971
	logid	-1.9042	-3.5818	-0.2266
	lvq	-1.4633	-2.7936	-0.1330
	qdisc	-0.5243	-1.6647	0.6161
	rbf	-1.2098	-2.5786	0.1589
CORR	back	2.8092	-12.7025	18.3209
	baytree	-1.4640	-16.6062	13.6782
	c4.5	-1.1681	-15.5978	13.2616
	disc	4.9116	-8.5428	18.3661
	indcart	0.1792	-14.6388	14.9971
	knn	1.5746	-13.4194	16.5686
	koho	-13.2591	-55.4997	28.9815
	logid	-0.0061	-14.3201	14.3078
	lvq	2.5333	-11.9133	16.9798
	qdisc	-2.4791	-16.4826	11.5244
	rbf	4.8136	-9.1736	18.8009
$CORR^2$	back	-0.3483	-22.9572	22.2607
	baytree	-0.7609	-24.5937	23.0719
	c4.5	-1.0045	-23.4946	21.4857
	disc	-5.6495	-25.7263	14.4273
	indcart	-3.1018	-26.3446	20.1410
	knn	-4.3122	-27.4735	18.8491
	koho	15.7761	-36.4425	67.9946
	logid	0.0472	-20.9759	21.0703
	lvq	-4.4098	-26.8636	18.0439
	qdisc	5.1887	-15.7745	26.1520
	rbf	-6.4274	-28.0028	15.1480

TABLE 2
Continued

Data Characteristic x_i	Classification Algorithm j	Percentile 2.5th	of 50th	$\hat{\beta}_j$ 97.5th
CANCOR1	back	-8.2983	-14.4300	-2.1666
	baytree	-2.4797	-8.8537	3.8943
	c4.5	-2.5943	-8.2872	3.0987
	disc	-3.2721	-7.2967	0.7525
	indcart	-3.0705	-9.0296	2.8887
	knn	-3.1736	-8.1698	1.8227
	koho	-8.0917	-22.1977	6.0143
	logid	-4.0409	-8.5431	0.4613
	lvq	-3.3089	-8.2342	1.6165
	qdisc	-2.95499	-7.0308	1.1209
	rbf	-4.80517	-9.3012	-0.3092
SKEW	back	0.85460	-1.2733	2.9825
	baytree	1.04625	-0.6522	2.7447
	c4.5	0.82553	-0.7296	2.3807
	disc	1.32582	-0.2722	2.9239
	indcart	1.00855	-0.5831	2.6002
	knn	1.71681	0.1250	3.3086
	koho	-4.85206	-17.0648	7.3606
	logid	2.43050	0.3743	4.4868
	lvq	1.62714	0.0317	3.2226
	qdisc	1.51729	0.0702	2.9643
	rbf	1.34189	-0.2084	2.8921
$SKEW^2$	back	-0.15137	-0.4439	0.1412
	baytree	-0.13413	-0.3795	0.1113
	c4.5	-0.11178	-0.3398	0.1163
	disc	-0.20136	-0.4320	0.0292
	indcart	-0.12852	-0.3613	0.1043
	knn	-0.16506	-0.3797	0.0495
	koho	0.70996	-1.0083	2.4283
	logid	-0.35198	-0.6476	-0.0563
	lvq	-0.18999	-0.4103	0.0303
	qdisc	-0.22693	-0.4261	-0.0278
	rbf	-0.17107	-0.3941	0.0519
HC	back	2.62164	0.4619	4.7814
	baytree	2.48039	0.5446	4.4162
	c4.5	2.33467	0.5445	4.1248
	disc	2.21915	0.3133	4.1250
	indcart	2.45385	0.5869	4.3208
	knn	1.87751	0.2453	3.5098
	koho	2.96798	0.1072	5.8287
	logid	3.03669	0.5876	5.4858
	lvq	1.73326	0.0941	3.3724
	qdisc	0.79551	-0.8681	2.4591
	rbf	2.16140	0.3399	3.9829

TABLE 2
Continued

Data Characteristic x_i	Classification Algorithm j	Percentile of $\hat{\beta}_j$ 2.5th	50th	97.5th
IIX	back	-0.38798	-5.5425	4.7665
	baytree	2.36313	-1.7399	6.4662
	c4.5	2.11204	-1.7245	5.9486
	disc	1.25853	-2.5774	5.0945
	indcart	2.43959	-1.4970	6.3762
	knn	3.45633	-0.1916	7.1043
	koho	4.72471	-1.1127	10.5621
	logid	3.47904	-1.0331	7.9912
	lvq	2.72594	-0.8978	6.3497
	qdisc	1.33964	-2.2731	4.9523
	rbf	1.99019	-1.7142	5.6946
HX^2	back	-0.04183	-0.6144	0.5307
	baytree	-0.29602	-0.7873	0.1953
	c4.5	-0.26251	-0.7141	0.1891
	disc	-0.20907	-0.6547	0.2366
	indcart	-0.29656	-0.7644	0.1713
	knn	-0.39487	-0.8172	0.0275
	koho	-0.56855	-1.2028	0.0657
	logid	-0.4635	-1.0011	0.0741
	lvq	-0.3127	-0.7309	0.1054
	qdisc	-0.1704	-0.5866	0.2459
	rbf	-0.2600	-0.6926	0.1727

TABLE 2
Continued

Data Characteristic x_i	Classification Algorithm j	Percentile of $\hat{\beta}_j$ 2.5th	50th	97.5th
$(b/p)^2$	back	-1.0792	-8.2719	6.1136
	baytree	0.1857	-6.6741	7.0454
	c4.5	0.0458	-6.4203	6.5118
	disc	0.6058	-5.7534	6.9651
	indcart	0.5061	-6.2119	7.2241
	knn	0.6968	-4.9535	6.3471
	koho	9.7545	-6.8599	26.3690
	logid	3.3835	-4.0608	10.8277
	lvq	1.6747	-4.1388	7.4882
	qdisc	0.2793	-5.2857	5.8443
	rbf	0.4525	-5.8229	6.7278
$\sqrt{p/r}$	back	0.3938	-18.3338	19.1215
	baytree	8.3257	-10.7260	27.3775
	c4.5	9.1364	-7.5617	25.8346
	disc	-2.4769	-17.1920	12.2383
	indcart	9.5821	-8.2155	27.3798
	knn	17.8830	3.8858	31.8802
	koho	1.9817	-15.6182	19.5815
	logid	-6.2988	-25.4889	12.8913
	lvq	14.9809	1.4676	28.4942
	qdisc	4.1628	-9.3769	17.7024
	rbf	9.7071	-5.0252	24.4395
k	back	-0.2689	-0.6003	0.0625
	baytree	-0.2693	-0.5736	0.0350
	c4.5	-0.2475	-0.5200	0.0250
	disc	-0.2475	-0.5095	0.0145
	indcart	-0.2584	-0.5481	0.0313
	knn	-0.1371	-0.3853	0.1111
	koho	-0.3999	-0.9492	0.1494
	logid	-0.3319	-0.6678	0.0040
	lvq	-0.1262	-0.3715	0.1190

HX are not noticeable. But, it can be said that, at 10 percent significance level, performances of KNN, KOHO, and LOGID are influenced by HX.

The relative number of feature variables to the training cases ($\sqrt{p/r}$) does not appear to have any significant effects on the performances of other algorithms except for KNN and LVQ. Interestingly, performances of these two algorithms decrease as $\sqrt{p/r}$ increases.

New ranks of the classification algorithms are obtained by fitting the error rate of individual cases based on the meta-model in Table 2. These new ranks can then be compared to the originally observed ranks by way of Spearman's rank correlation. Spearman's rank correlation is obtained as the correlation of the new rank and the observed rank of each classification algorithm applied to a certain data set: $1 - 6(\sum_{j=1}^{L} d_j^2)/L(L^2-1)$, where d_j is the difference between the observed rank and the fitted rank of classification algorithm j. Relatively high Spearman's rank correlation displayed in Table 4 supports the use of meta-model for the ranking purpose of classification algorithms. If more meaningful data characteristics were available, the accuracy of the statistical meta-model could have been improved.

The statistical meta-model derived in this paper can be compared to the meta-level classification rule trained using C4.5 in [17]. The rule trained by C4.5 sentences each classification algorithm to be either applicable or not to the data set given based on associated data characteristics. In order to train the meta-level classification rule based on C4.5, each classification algorithm is categorized into two (applicable vs. nonapplicable) for each empirical data. Then, the characteristics of each data are related to the classification result of each algorithm. C4.5 in turn generates a rule for each classification algorithm based on some selected data characteristics. However, presentation of the rule is multidimensional and is not easily comprehensible for all classification algorithms. On the other hand, the statistical meta-model derived in this paper is one-dimensional and can be easily used to estimate the expected classification error for a given set of data characteristics.

TABLE 3
Average Prediction Error

Tool	Average Error Rate	n_j	Average Prediction Error
All	0.13	203	0.018
back	0.11	17	0.069
baytree	0.12	19	0.054
c4.5	0.12	19	0.057
disc	0.13	19	0.020
indcart	0.12	19	0.060
knn	0.12	19	0.084
koho	0.18	16	0.063
logid	0.12	19	0.048
lvq	0.12	18	0.079
qdisk	0.13	19	0.063
rbf	0.12	19	0.055

TABLE 4
Spearman's Rank Correlation

name	belg	belg2	chrom	cr.aust	cred.man	cut20
correlation	0.92	0.70	0.99	0.46	0.95	0.97
name	cut50	diab	dig44	dna	faults	kl
correlation	0.89	0.78	1.00	0.62	0.93	0.93
name	letter	satim	segm	shuttle	tsetse	vehicle
correlation	0.99	0.96	0.88	0.84	0.63	0.96

5 Discussion

We develop a statistical meta-model which can be used to rank several classification algorithms based on the data characteristics given. Performance of the statistical meta-model fitted in this paper appears to be high. But, in order to improve the accuracy of the meta-model, in addition to the data characteristics considered in this paper, more measurements which could properly describe the multivariate data characteristics should be developed. Additionally, in view of the characteristics of the real-time monitored data, which is often encountered in data mining situation, information regarding the degree of the autocorrelation of feature variables is essential [24], [29].

What has to be further investigated is a meta-level multivariate ranking model for both classification error and classification time. It appears that both classification error and classification time are important response variables to consider when choosing an appropriate classification algorithm. These two variables could be potentially related and multivariate analysis is necessary to find the algorithm with respect to these two responses. One more aspect to be included in performance comparison is the ability of feature selection of each classification algorithm [10], [15], [16], [17], [21], [22], [23].

Feature extraction is one of the important topics in data mining research. However, comparison of classification algorithms in the aspect of feature selection accuracy has not been extensively investigated and is left as one of further study areas.

Appendix

Description of Data Set

cred.man was donated by a major British engineering company and comes from the general area of credit management including assessing methods for pursuing debt discovery.

cr.aust was used to assess applications for credit card.

dig44 consists of 18,000 examples of the digits 0 to 9 gathered from postcodes on letters in Germany. The handwritten examples were digitised onto images with 16×16 pixels and 256 gray levels.

kl uses the first 40 principal components of dig44 data as feature variables.

vehicle was originally gathered at the Turing institute to find a method of distinguishing 3D objects within a 2D image by application of an ensemble of shape feature extractors to the 2D silhouettes of the objects.

letter was constructed to classify each of large number of black and white rectangular pixel displays as one of the 26 capital letters of the English alphabet.

chrom was obtained from the routine amniotic 2,668 cell data set. Features are grouped into four depending upon the difficulty of observation.

satim is a subarea of a scene, consisting of 82×100 pixel covering an area on the ground of approximately 80×80 meters. The information given for each pixel consists of the class value and the intensities in four spectral bands.

segm was randomly drawn from a database of seven outdoor color images.

*cut*50 was constructed during an investigation into the problem of segmenting individual characters from joined written text with 50 real valued attributes.

*cut*20 uses best 20 attributes selected by stepwise regression on the *cut*50.

shuttle was originated from NASA and concerned with the position of radars within the Space Shuttle.

diab was used to predict whether a patient would test positive for diabetes given a number of physiological measurements and medical test results.

dna was used to recognize, given a sequence of DNA, the boundaries between exons and introns.

tech contains information of commercial interest to Daimler-Benz AG Germany. Very little is known about this data set as the nature of the problem domain is secret.

belg is used to find a fast and reliable indicator of instability in large scale power system. Thus, data has been constructed by simulating up to five minutes of the system behavior.

*belg*2 is drawn from a larger simulation than *belg*.

faults involves the financial aspect of mechanical maintenance and repair.

tsetse contains information concerning climatic conditions under which tsetse fly thrives or not.

ACKNOWLEDGMENTS

This work was partly supported by the Yonsei University Research Fund of 1997 and by grant no. 1999-1-303-005-3 from the Interdisciplinary Research Program of the KOSEF..

REFERENCES

[1] S. Aeberhard, D. Coomans, and O. De Vel, "Comparative Analysis of Statistical Pattern Recognition Methods in High Dimensional Settings," *Pattern Recognition*, vol. 27, no. 8, pp. 1,065-1,077, 1994.
[2] L. Breiman, J.H. Friedman, R.A. Olshen, and C.J. Stone, *Classification and Regression Trees*. Monterey, Calif.: Wadsworth and Brooks, 1984.
[3] F.Z. Brill, D.E. Brown, and W. Martin, "Fast Genetic Selection of Features for Neural Network Classifiers," *IEEE Trans. Neural Networks*, vol. 3, no. 2, pp. 324-328, 1992.
[4] C. Brodlly and P. Utgoff, "Multivariate Decision Trees," *Machine Learning*, vol. 19, pp. 45-77, 1995.
[5] W. Buntine, "Learning Classification Trees," *Statistics and Computing*, vol. 2, pp. 63-73, 1992.
[6] B.A. Draper, C.E. Brodley, and P.E. Utgoff, "Goal-Directed Classification Using Linear Machine Decision Trees," *IEEE Trans. Pattern Analysis and Machine Intelligence*, vol. 16, no. 9, pp. 888-893, 1994.
[7] B. Efron, *The Jackknife, the Bootstrap and Other Resampling Plans*. Philadelphia: SIAM, 1982.
[8] U. Fayyad and E. Simoudis, "An Introduction to Effective Data Mining," *Tutorial Notes, First Pacific-Asia Conf. Knowledge Discovery & Data Mining*, Singapore, Feb. 1997.
[9] R. Gnanadesikan and J.R. Kettenring, "Weighting and Selection of Variables for Cluster Analysis," *J. Classification*, vol. 12, no. 1, p. 113, 1995.
[10] D.W. Hosmer and S. Lemeshow, *Applied Logistic Regression*. John Wiley & Sons, 1989.
[11] E.A. Joachimsthaler and A. Stam, "Four Approaches to the Classification Problem in Discriminant Analysis: An Experimental Study," *Decision Sciences*, vol. 19, pp. 323-333, 1988.
[12] B.J. Jung and S.Y. Sohn, "Determination of an Economic Lot Size of Color Filters in TFT-LCD Manufacturing," *IE Interfaces*, vol. 10, pp. 47-56, 1997.
[13] D. Koller and M. Sahami, "Toward Optimal Feature Selection," *Proc. 13th Int'l Conf. Machine Learning (ML)*, Bari, Italy, July 1996.
[14] J. Livarinen, K. Valkealahti, A. Visa, and O. Simula, "Feature Selection with Self-Organizing Feature Map," *Proc. Int'l Conf. Artificial Neural Networks*, vol. 1, Sorrento, Italy, May 1994.
[15] D. Lowe and A.R. Webb, "Optimized Feature Extraction and the Bayes Decision in Feed-Forward Classifier Networks," *IEEE Trans. Pattern Analysis and Machine Intelligence*, vol. 13, no. 4, pp. 355-364, 1991.
[16] O.L. Mangasarian, W.N. Street, and W.H. Wolberg, "Breast Cancer Diagnosis via Linear Programming," *Operations Research*, vol. 43, pp. 570-577, 1995.
[17] D. Michie, D.J. Spiegelhalter, and C.C. Taylor, *Machine Learning, Neural and Statistical Classification*. New York: Ellis Horwood, 1994.
[18] J.R. Quinlan, *C4.5: Programs for Machined Learning*. San Mateo, Calif.: Morgan Kaufmann, 1993.
[19] F.E. Shaudys and T.K. Leen, "Feature Selection for Improved Classification," *Proc. Int'l Joint Conf. Neural Networks*, Baltimore, 1992.
[20] W. Siedlecki and J. Sklansky, "On Automatic Feature Selection," *Int'l J. Pattern Recognition and Artificial Intelligence*, vol. 2, no. 2, pp. 197-220, 1988.
[21] J.E. Smith, T.C. Fogarty, and I.R. Johnson, "Genetic Feature Selection for Clustering and Classification," *Proc. IEE Colloquium Genetic Algorithms in Image Processing Vision*, p. 193, 1994.
[22] S.Y. Sohn, "Accelerated Life-Tests for Intermittent Destructive Inspection with Logistic Failure-Distribution," *IEEE Trans. Reliability*, vol. 46, pp. 122-129, 1997.
[23] S.Y. Sohn, "Mining Large Maintenance Database," *Facility Maintenance Eng. Proc.*, pp. 157-160, Seoul, Korea, May 1997.
[24] S.Y. Sohn, "Variable Selection with Correlated Binary Data," submitted for publication, 1997.
[25] S.Y. Sohn, "Bayesian Dynamic Forecasting for Attribute Reliability," *Computers and IE*, vol. 33, nos. 3-4, pp. 741-744, 1997.
[26] S.Y. Sohn, "Statistical Analysis of Environmental Effects on TOW Missile Stockpile Deterioration," *IIE Trans.*, vol. 28, pp. 995-1,002, Dec. 1996.
[27] S.Y. Sohn, "Growth Curve Analysis Applied to Ammunition Deterioration," *J. Quality Technology*, vol. 27, no. 4, pp. 71-80, 1996.
[28] S.Y. Sohn, "Monitoring Declining Quality of Ammunition Stockpile under Step-Stress," *Naval Research Logistics*, vol. 41, pp. 707-718, Mar. 1994.
[29] S.Y. Sohn, "Variable Selection in a Linear Growth Curve Model with Autoregressive Within-Individual Errors," *J. Statistical Computation and Simulation*, vol. 40, no. 2, pp. 247-255, 1992.
[30] K. Srinivasan and D. Fisher, "Machine Learning Approaches to Estimating Software Development Effort," *IEEE Trans. Software Eng.*, vol. 21, no. 2, pp. 126-136, 1995.
[31] P.D. Wasserman, *Advanced Methods in Neural Computing*. New York: Van Nostrand Reinhold, 1993.
[32] M.A. Wong and T. Lane, "A kth Nearest Neighbor Clustering Procedures," *J. Royal Statistical Soc.*, Ser. B, 45, pp. 362-368, 1983.

So Young Sohn received a PhD in industrial engineering from the University of Pittsburgh in 1989 and has MS degrees in statistics and management science. She is a professor of industrial systems engineering at Yonsei University. She was a recipient of the American Statistical Association/National Science Foundation/U.S. Bureau of Labor Statistics Research Associateship (1990) and the British Foreign and Commonwealth Office awards (1985-1986). Prior to joining Yonsei University, she gained valuable teaching and research experience at Rensselaer Polytechnic Institute (1995-1996), the Naval Postgraduate School (1990-1995), the U.S. Bureau of Labor Statistics, and the Korea Institute for Defense Analysis (1983-1985) as an assistant professor, applied statistician, and an operations research analyst, respectively.