New Evolutionary Approaches to High-Dimensional Data

Luis Matoso[1], Felipe Junior[1], Adriano Machado[2], Adriano Veloso[1], Wagner Meira Jr.[1]
Computer Science Department
[1]Universidade Federal de Minas Gerais
[2]Centro Federal de Educação Tecnológica de Minas Gerais
{lmatoso, felipe, adrianoc, adrianov, meira}@dcc.ufmg.br

ABSTRACT

High-dimensional data often threatens the performance of classification algorithms. We propose a two-step approach for dealing with high-dimensional data. In the first step, features are arranged into bins, where each bin corresponds to a much smaller sub-space of features. In the second step, classifiers are independently applied to the set of features within each sub-space, and their results are then aggregated. We consider slicing a space R^d into smaller sub-spaces as a multi-objective search problem, which can be solved by evolutionary algorithms. We performed a systematic evaluation using three classification algorithms on high-dimensional data.

Categories and Subject Descriptors

H.3.3 [**Information Search and Retrieval**]: Information Search and Retrieval

Keywords

Evolutionary Algorithms, High-Dimensional Data, Dimensionality Reduction

1. INTRODUCTION

The typical input of a classification algorithm is a set of instances, each one being composed of d features in addition to a target variable. Therefore, instances are essentially interpreted as points in a R^d feature space, and we say that the data is high-dimensional when d is beyond the hundreds. In such cases, the size of R^d leads to complications that rapidly increase learning times and decrease learning accuracy. Approaches to this problem include: (i) finding a representative subset of all features, or (ii) mapping of the original high-dimensional data onto a lower-dimensional space. Ideally, these approaches must preserve as much information from the original data as possible. In practice, however, information is lost as dimensionality is reduced.

In this paper we propose a novel two-step approach. In the first step (Slice step), the R^d space is sliced into multiple sub-spaces $\{R_1^\alpha, R_2^\beta, \ldots, R_n^\zeta\}$, where each sub-space R_i^m is much smaller than the original space R^d. Then, in the second step (Aggregate step), classifiers are independently applied to each sub-space R_i^m, and the corresponding results are finally aggregated. We hypothesize that effective aggregation needs each sub-space R_i^m to be accurate with regard to R^d. Thus, we consider slicing R^d as a search problem, which is solved using evolutionary algorithms.

GECCO'12 Companion, July 7–11, 2012, Philadelphia, PA, USA.
ACM 978-1-4503-1178-6/12/07.

2. SLICE AND AGGREGATE

2.1 Slicing Approaches

Slicing a feature space R^d can be viewed as a search problem, in which possible solutions are given as a combination of sub-spaces $\{R_1^m, R_2^m, \ldots, R_n^m\}$, such that features composing each sub-space R_i^m are selected in a way that optimizes a established criterion. We consider the application of evolutionary algorithms for searching optimal solutions. Next we precisely define an individual.

Definition 1: *An individual is a possible combination of sub-spaces (i.e., a candidate solution), and is encoded as a sequence of d integers $[v_1, v_2, \ldots, v_d]$, where each v_i ranges from 1 to n and indicates the sub-space for which the i^{th} feature is associated with.*

A fitness function is associated with each individual in order to make them directly comparable, so that the population can evolve towards optimal solutions. The ideal criterion, for the sake of dimensionality reduction, is to find sub-spaces composed of features that jointly have the largest dependency on the target variable c [1]. Maximizing the dependency on c can be approximated by maximizing relevance while minimizing redundancy [3].

Definition 2: *The relevance of a sub-space R_i^m, denoted as $F(R_i^m)$, is given by the mean value of all mutual information values between features $x_a \in R_i^m$ and the variable c, as shown in Equation 1. The redundancy of a sub-space R_i^m, denoted as $S(R_i^m)$, is given by the mean value of all mutual information values between pairs of features $\{x_a, x_b\} \in R_i^m$, as shown in Equation 2.*

$$F(R_i^m) = \frac{1}{m} \times \sum_{x_a \in R_i^m} I(x_a, c) \tag{1}$$

$$S(R_i^m) = \frac{1}{m^2} \times \sum_{\{x_a, x_b\} \in R_i^m} I(x_a, x_b) \tag{2}$$

Definition 3: *An optimal solution is a combination of sub-spaces $\{R_1^m, R_2^m, \ldots, R_n^m\}$, satisfying Equation 3.*

$$\text{maximize} \quad \phi(R_i^m)\ \forall R_i^m : \phi = \frac{F(R_i^m)}{S(R_i^m)} \tag{3}$$

Searching for optimal solutions, therefore, is a multi-objective search problem, in which the value of ϕ must be maximized for each of the n sub-spaces that compose an optimal solution.

Averaging Objectives (AO). A simple approach for solving a multi-objective search problem is to combine all n objectives by maximizing the average of all ϕ values.

Pareto-Optimal Slicing (POS). A more general approach is to exploit Pareto-dominance tests, in order to find solutions that

are not dominated by others. These non-dominated solutions lie in the so-called Pareto frontier, and thus, the evolutionary algorithm evolves the population towards producing individuals that are located closer to the Pareto frontier [5].

2.2 Aggregation Approaches

Once the optimal combination of sub-spaces $\{R_1^m, R_2^m, \ldots, R_n^m\}$ is found, a classifier is applied to each sub-space independently. In this case, the classifier takes as input the training-set $\mathcal{D}$ and the test-set $\mathcal{T}$, but instead of considering all d features, only features in R_i^m are considered. Then, the classifier outputs a probability $\hat{p}_i(c|t)$ $\forall t \in \mathcal{T}$, which shows the likelihood of t being associated with the target variable c. The same procedure is performed for all n sub-spaces, resulting in n different probabilities associated with the same instance t: $[\hat{p}_1(c|t), \hat{p}_2(c|t), \ldots, \hat{p}_n(c|t)]$, and a final probability $\hat{p}(c|t)$ is obtained by aggregating these probabilities. Next we consider two aggregation approaches.

Averaging Probabilities (AP). This approach simply returns, for each instance $t \in \mathcal{T}$, the average of the n probabilities.

Pareto-Efficient Aggregation (PEA). This approach interprets each probability $\hat{p}_i(c|t)$ as a coordinate in a n-dimensional scatter-gram, and each instance $t \in \mathcal{T}$ is associated to a point in this scatter gram. Higher probabilities $\hat{p}(c|t)$ are assigned to instances that are associated with non-dominated points. Such instances lie on the Pareto frontier of the scatter gram. Stripping off instances from successive Pareto frontiers yields a partial ordering of instances, defined as a Pareto-optimal rank, that is, instances in t are ranked as $\{t_1, t_2, \ldots, t_n\}$, such that there is no instance t_i that dominates instance t_j, given that $i > j$. Finally, a probability $\hat{p}(c|t)$ is assigned to each instance $t \in \mathcal{T}$, according to Equation 4, where $r(t)$ gives the position of instance t in the rank, which is given according to the corresponding dominance count of t.

$$\hat{p}(c|t) = \frac{|\mathcal{T}| - r(t)}{|\mathcal{T}|} \quad (4)$$

3. EXPERIMENTAL EVALUATION

We employ measures such as precision/recall curves, which are summarized by means of their area (i.e., AUC). We use several baselines, including PCA [2], mRMR [3], and LDA [4]. Three different classifiers were used in order to evaluated our approaches: associative classifiers (AC), SVMs, and Naive Bayes (NB). We conducted five-fold cross validation, and significance tests were performed ($p < 0.05$) using the paired t-test. If a result is statistically different from the result obtained by best baseline, we show it in bold. In addition, the best overall results are marked with a †.

Our application scenario concerns the retrieval of images, based on their content. Each image is represented as a vector of descriptors, and our task is to rank higher images that are most similar to a given query-image. Table 1 shows results obtained by each evaluated classifier. Four scenarios are considered: (i) the original data is given to the classifier (i.e., no dimensionality reduction is performed), or dimensionality is first reduced using either (ii) mRMR, (iii) PCA, or (iv) LDA. Table 1 shows the results for each scenario.

Table 2 shows AUC values obtained for different Slice-Aggregate configurations. When sub-spaces are created using AO or POS approaches, the results achieved by AP and PEA approaches are much better than the results obtained by considering sub-spaces in isolation. Further, as shown in Table 2, effectiveness increases with the number of sub-spaces involved in the process, and we can see that multi-objective search POS approach is superior than the AO approach. The same trend is observed when analyzing aggregation approaches. Specifically, PEA is superior when compared with AP.

Table 1: Average AUC values for the baselines.

	Original	mRMR	PCA	LDA
AC	0.413	0.402	0.389	0.391
SVM	0.404	0.398	0.382	0.386
NB	0.416	0.400	0.393	0.397

Table 2: Average AUC values for different classifiers. Baseline for each classifier is shown between parentheses.

		Avg. Objectives (AO)			Pareto Opt. Slicing (POS)		
	n	isolated	AP	PEA	isolated	AP	PEA
AC (0.402)	2	0.399	0.408	0.411	0.404	0.414	**0.418**†
	3	0.398	0.412	0.418	0.401	0.417	**0.422**†
	4	0.396	**0.418**	**0.421**	0.399	**0.421**	**0.427**†
	6	0.394	**0.422**	**0.424**	0.398	**0.424**	**0.430**†
SVM (0.398)	2	0.395	0.400	0.403	0.399	0.405	0.410
	3	0.399	0.406	0.408	0.397	0.407	**0.415**
	4	0.395	**0.412**	**0.416**	0.397	**0.416**	**0.420**†
	6	0.392	**0.412**	**0.414**	0.396	**0.416**	**0.424**†
NB (0.400)	2	0.402	0.407	0.411	0.405	0.414	**0.420**†
	3	0.401	0.414	0.417	0.402	**0.419**	**0.425**†
	4	0.399	**0.420**	**0.423**	0.400	**0.423**	**0.428**†
	6	0.396	**0.422**	**0.426**	0.398	**0.428**	**0.432**†

4. CONCLUSIONS

We modeled the dimensionality reduction task as a search problem which can be efficiently solved using evolutionary algorithms. Once the optimal solution is found, each sub-space is given as input to a classifier, and the corresponding outputs are finally aggregated into a final output. We propose slicing and aggregation approaches, and in order to evaluate our approaches, we use high-dimensional data obtained from content-based image retrieval databases.

5. ACKNOWLEDGMENTS

This work was partially sponsored by Universo OnLine S. A. - UOL (www.uol.com.br) and partially supported by the Brazilian National Institute of Science and Technology for the Web (CNPq grant no. 573871/2008-6), CAPES, CNPq, Finep, and Fapemig.

6. REFERENCES

[1] C. Ding and H. Peng. Minimum redundancy feature selection from microarray gene expression data. *J. Bioinformatics and Computational Biology*, 3(2):185–206, 2005.

[2] I. Jolliffe. *Principal Component Analysis*. Springer, 1986.

[3] H. Peng, F. Long, and C. Ding. Feature selection based on mutual information: Criteria of max-dependency, max-relevance, and min-redundancy. *IEEE Trans. Pattern Anal. Mach. Intell.*, 27(8):1226–1238, 2005.

[4] H. Yu and J. Yang. A direct LDA algorithm for high-dimensional data - with application to face recognition. *Pattern Recognition*, 34(10):2067–2070, 2001.

[5] E. Zitzler and L. Thiele. Multio-bjective evolutionary algorithms: a comparative case study and the strength pareto approach. *IEEE Trans. Evolutionary Computation*, 3(4):257–271, 1999.