Knowledge-Based Systems 000 (2018) 1–9

Contents lists available at ScienceDirect

Knowledge-Based Systems

journal homepage: www.elsevier.com/locate/knosys

Principal Components Analysis Random Discretization Ensemble for Big Data

Diego García-Gil[a,*], Sergio Ramírez-Gallego[a], Salvador García[a], Francisco Herrera[a,b]

[a] Department of Computer Science and Artificial Intelligence, University of Granada, CITIC-UGR, Granada 18071, Spain
[b] Faculty of Computing and Information Technology, King Abdulaziz University, Jeddah 21589, Saudi Arabia

ARTICLE INFO

Article history:
Received 4 September 2017
Revised 6 February 2018
Accepted 8 March 2018
Available online xxx

Keywords:
Big Data
Discretization
Spark
Decision tree
PCA
Data reduction.

ABSTRACT

Humongous amounts of data have created a lot of challenges in terms of data computation and analysis. Classic data mining techniques are not prepared for the new space and time requirements. Discretization and dimensionality reduction are two of the data reduction tasks in knowledge discovery. Random Projection Random Discretization is a novel and recently proposed ensemble method by Ahmad and Brown in 2014 that performs discretization and dimensionality reduction to create more informative data. Despite the good efficiency of random projections in dimensionality reduction, more robust methods like Principal Components Analysis (PCA) can improve the performance.

We propose a new ensemble method to overcome this drawback using the Apache Spark platform and PCA for dimension reduction, named Principal Components Analysis Random Discretization Ensemble. Experimental results on five large-scale datasets show that our solution outperforms both the original algorithm and Random Forest in terms of prediction performance. Results also show that high dimensionality data can affect the runtime of the algorithm.

1. Introduction

Nowadays everything is constantly creating and storing data. This massive data accumulation can be found in a broad spectrum of real-world areas [42]. In 2014 IDC predicted that by 2020, the digital universe will be 10 times as big as it was in 2013, totaling an astonishing 44 zettabytes [23]. Big Data is not only a huge amount of data, but a new paradigm and set of technologies that can store and process this data. Many of the classic knowledge extraction techniques are not capable of working with these amounts of data, mostly because they were not conceived to work in a Big Data environment.

This scenario becomes particularly important using data reduction techniques. Data reduction techniques are frequently applied to reduce the size of the original data and to clean some errors that it may contain [18,19]. Two important data reduction techniques are discretization and dimensionality reduction. Discretization is the process of splitting a continuous variable into different categories depending on which interval it falls into [32]. Dimensionality reduction is the mapping of data to a lower dimensional space [39]. One of the most popular methods for dimensionality reduction is Principal Components Analysis (PCA) [25]. PCA is a linear transformation from a high dimensional data space to a principal component feature space. It has been widely used as a dimension reduction technique in many applications.

Ensembles are methods that combine a set of base classifiers to make predictions [13,29]. These methods have been attracting increasing attention over the last few years due to their ability to correct errors across many diverse base classifiers. Classifier ensembles have shown to be very effective in a broad spectrum of real-world applications. These classifiers have been proven to be *accurate* and *diverse* [31,41]. Ensembles of decision trees like Random Forest are well known for creating diverse decision trees [34]. This diversity is usually introduced via randomization. Through small changes in input data, diverse decision trees are created and better ensembles are obtained.

This principle is followed by Ahmad and Brown in [2]. Random Projection Random Discretization (RPRD) is an ensemble method that applies two data reduction techniques to the input data and joins the results to create a more informative dataset. First it performs their proposed Random Discretization (RD), which selects $s-1$ random instances to create s categories and discretize the data using the selected values as limits. Their next step is to perform Random Projection (RP) [24] in the original data to select d

[*] Corresponding author.
E-mail addresses: djgarcia@decsai.ugr.es (D. García-Gil), sramirez@decsai.ugr.es (S. Ramírez-Gallego), salvagl@decsai.ugr.es (S. García), herrera@decsai.ugr.es (F. Herrera).

https://doi.org/10.1016/j.knosys.2018.03.012
0950-7051/© 2018 Elsevier B.V. All rights reserved.

features that are the linear combinations of the original m features ($d < m$). Finally the algorithm joins the results of RD and RP to create a new $m + d$ dataset and trains a decision tree with it.

The RPRD Ensemble algorithm has shown to be competitive and to outperform other popular ensemble methods. However, despite its good performance, it still has three main drawbacks: (1) As the projected dimension is decreased, as it drops below $\log k$, random projection suffers a gradual degradation in performance [10]. (2) RP is highly unstable - different random projections may lead to radically different results [17]. There are more informative dimensionality reduction methods like PCA. (3) RPRD is not prepared for working with Big Data. This therefore limits the potential use of the ensemble. For example, in cases with thousands of features or millions of instances, RPRD cannot be used.

In order to fill this gap and inspired by the RPRD ensemble algorithm, we propose a new ensemble method under Apache Spark using PCA, called Principal Components Analysis Random Discretization Ensemble (PCARDE) for Big Data. In our design we use PCA instead of RP for improving the dimensionality reduction step. The choice of PCA is motivated by the fact that it is not an iterative method and can be parallelized. The usage of Big Data frameworks like Apache Spark enables the use of PCA over datasets with thousands of features and millions of instances. It also allows to perform huge matrix multiplications for methods like RP. In Big Data problems, the computational cost difference between PCA and RP becomes unclear, since the multiplication of large matrices in RP is a computationally demanding operation.

We have also designed an algorithm named $\mathcal{X}^2$ Random Discretization Ensemble ($\mathcal{X}^2$ RD), which uses $\mathcal{X}^2$ for performing feature selection, in order to show the importance and impact of the addition of PCA to our proposed ensemble method. The choice of $\mathcal{X}^2$ comes motivated by the fact that $\mathcal{X}^2$ is a more informed method than RP and this can lead to a performance improvement.

Apache Spark is a fast and general engine designed for large-scale data processing based on in-memory computation [22,37]. Apache Spark has its own Machine Learning library named MLlib [30]. Among our objectives is designing an ensemble algorithm for Big Data and to integrate the algorithm into the MLlib Library as a third-party package. Spark's implementation of the algorithm can be downloaded from the Spark's community repository[1].

To show the effectiveness of our approach, we have carried out an experimental evaluation with five large datasets, namely *poker, SUSY, HIGGS, epsilon and ECBDL14*. These datasets have very different properties and allow us to test all aspects of our implementation. Finally, we show a comparative study of the performance of PCARDE, $\mathcal{X}^2$ RD, RPRD and Random Forest [7,35].

The remainder of this paper is organized as follows: Section II outlines the main concepts of the RPRD Ensemble. Section III explains the new ensemble design based on PCA. Section IV describes the experiments carried out to check the effectiveness of this proposal. Finally, Section V concludes the paper.

2. Background

In this section we first introduce the RPRD Ensemble algorithm used as reference in our ensemble interpretation and its two components, RD and RP. Then we introduce PCA and MapReduce Model.

2.1. Random discretization

Discretization is the process of partitioning a set of continuous attributes into discrete attributes by associating categorical values to the intervals [20].

To create s categories we need $s - 1$ different intervals. There are different methods to create these intervals like entropy minimization [15], implemented in Apache Spark. The main problem with these methods is that they create the same discretized dataset after different executions. In an ensemble some randomization is necessary in order to introduce diversity to the decision trees.

In RD randomization is introduced to the discretization process. In Algorithm 1 we describe the mechanism of RD from lines 5 to 12. First $s - 1$ data points are randomly selected from the training data to create s categories. Then for each feature, every $s - 1$ data points are sorted. Finally the dataset is discretized into s categories using these $s - 1$ sorted data points. These thresholds are selected randomly each iteration of the ensemble. It is possible that in some features, all selected data points will have the same value. In this case, $s - 1$ are selected randomly between the minimum and the maximum values of the feature.

Algorithm 1 RPRD Ensemble.

```
1:  Input: Dataset $T$ with $m$ continuous features and $k$ classes
    $c_1, c_2, \ldots, c_k$. $L$ the size of the ensemble.
2:  Learning Phase
3:  for $i = 1 \ldots L$ do
4:      Random Discretization
5:      For $s$ categories in each dimension, select $s - 1$ data points
        randomly from the training data.
6:      for $j = 1 \ldots m$ do
7:          Get the $j$th feature values of $s - 1$ points and sort them.
8:          If all points have the same value, select $s - 1$ points randomly until there are at least two different values.
9:      end for
10:     for $i = 1 \ldots N$ do
11:         Discretize the $i$th data point using the values got in the previous step to create $m$ discretized features $S_i$.
12:     end for
13:     Random Projection
14:     Use Random Projection $RP_i$ to create $d$ features $R_i$.
15:     Combine $S_i$ and $R_i$ to create $m + d$ dimensional dataset $T_i$.
16:     Learning Model
17:     Treat dataset $T_i$ as continuous and learn $D_i$ decision tree on
        it.
18: end for
19: Prediction Phase
20: for $i = 1 \ldots L$ do
21:     Convert $x$ into $m + d$ dimensional data point $x_i$ using Random Discretization ($RD_i$) and Random Projection ($RP_i$).
22:     Let $p_{i,j}(x)$ be the probability for $x_i$ by the decision tree $D_i$ to
        the hypothesis that $x$ comes from class $c_j$. Calculate $p_{i,j}(x)$
        for all classes $j = 1.k$.
23: end for
    Calculate the confidence $C(j)$ for each class $c_j (j = 1.k)$ by the
    average contribution method,
    $C(j) = \frac{1}{L} \sum_{i=1}^{L} p_{i,j}(x)$
    The class with the largest confidence will be the class of $x$.
```

2.2. Random projection

The objective of dimensionality reduction techniques is to produce a compact low-dimensional encoding of a given high dimensional dataset. Random projection (RP) has emerged as a novel method for the dimensionality reduction problem.

[1] http://spark-packages.org/package/djgg/PCARD.