Ludwig-Maximilians-Universität München
Institut für Informatik
Dr. Eirini Ntoutsi
PD Dr. Matthias Schubert

Knowledge Discovery in Databases II

WS 2015/2016

Übungsblatt 6: Distributed and Parallel Data Mining

Aufgabe 6-1 Sparse Matrix Multiplication in Spark

Matrix Multiplication is a basis operation in many machine learning and data mining algorithms. Examples are the linear computation of PCA, matrix factorizations and the computation of kernel matrices. Often feature vectors in large data sets are sparse, i.e. the majority of feature values is 0. Given a sparse data set, matrix multiplication can be efficiently implemented in parallel frameworks like Spark.

(a) Download the Code Template spark_mult_template.py and study it.

(b) Implement the method to_triple which taken a matrix and transforms it into row, column and value pairs.

(c) Write a method which multiplies the RDD representations of the matrices A and B.

Aufgabe 6-2 Privacy Preservation in Standard Classifiers

Given the following classifiers: decision trees, nearest neighbor classification, support vector machines, and naive bayes.

- Discuss whether pre-trained classifiers can be distributed to third parties without giving access to parts of the training set.
- How could encountered problems be solved?

Aufgabe 6-3 Parallele Association Rules

Discuss the advantages and disadvantages of horizontal and vertical partitionings in the parallel generation of association rules using the apriori algorithm.

Aufgabe 6-4 Parallel Naive Bayes Classification with Map Reduce

Describe a program which calculates all required probabilities for a Naive Bayes classifier using MapReduce.

Assume that each class can be modeled by a multivariate axis-parallel normal distribution and that the training set D is given as tupels $< ID, object >$ with $object$ having attributes c and v. Let ID be a key for each object, $c \in C$ be the class, and $v \in \mathbb{R}^d$ be a feature vector.

Specify a function for the mapper and a function for the reducer in pseudo-code.