Data Mining: Learning from Large Data Sets - Fall Semester 2015

mmarti@student.ethz.ch
trubeli@student.ethz.ch

November 8, 2015

Large Scale Image Classification

In this task, the goal was to use support vector machines (SVM) to classify images into two classes, nature or people. We were supposed to implement the SVM model for mapreduce and apply data transformations and parameter optimizations to get the best score possible. Our goal was to find a vector w of weights, such that it minimizes a regularized hinge loss function.

$$w = \min_w \lambda w^T w + \sum_i \max(0, 1 - y_i w^T x_i)$$

0.1 Stochastic Gradient Descent

Our first approach used a stochastic gradient descent (SGD) classifier in the mapper and computed the average of the weights received from all mappers in the reducer, according to the slides. The stochastic gradient descent classifier used many parameters, the most important ones of which were the alpha coefficient, which restricts the set of possible solutions and the L1 ratio. To find a good solution, we tried to vary the parameters and observe the effect. In doing so, we managed to push our score to 0.77527 which was very close to the easy baseline. It became clear, that we would need to apply some form of data transformation to make the data linearly separable.

0.2 Highest Variance Features

Our first idea was to reduce the number of features to only contain features that matter for the classification. To do so, we selected the nine features with the highest variance in our training data and only used them to do the classification. The score did not reflect our expectation though and we needed to look for further solutions.

0.3 Polynomial Features

The next idea was to add polynomial features to cope with nonlinearity in our data. We first tried to add polynomial features for all of our original features up to a degree of two, but doing so resulted in feature vectors with more than 80000 features which made the solution unfeasible. Our next idea was

to combine the highest variance feature approach with the polynomial feature approach and as such, we once again took the features with a variance higher than a certain threshold and added polynomial features up to degree two for all of them. Interestingly, this approach got a very bad score, which meant that when we inverted the labels, we got a very good score locally. The score of the inverted classifier was not reflected by the data set on the hadoop cluster online though, which indicated that we were overfitting our classifier to the training data. Reducing the number of features did not increase our score though.

0.4 Problems

We had a lot of problems with this task. First, we struggled to get an initial solution running on the hadoop cluster. Our solution was working locally, but the cluster reported an error which we could only resolve once we asked the assistants for more details on the error message. Second, while the concept of the SVM model were quite clear to us, we struggled with transforming our data accordingly. While we tried different solutions that all seemed logic to us, the results were not getting better. Our best solution was therefore our initial solution where we simply applied a stochastic gradient descent and optimized our parameters by hand.