Text Categorization via Ellipsoid Separation

Andriy Kharechko John Shawe-Taylor
University of Southampton, UK
{*ak03r,jst*}*@ecs.soton.ac.uk*

Ralf Herbrich Thore Graepel
Microsoft Research Ltd., Cambridge, UK
{*rherb,thoreg*}*@microsoft.com*

Abstract

We present a new batch learning algorithm for text classification in the vector space of document representations. The algorithm uses ellipsoid separation [3] in the feature space which leads to a semidefinite program. An approximation of the latent semantic feature extraction approach using Gram-Schmidt orthogonalization [2] is used for the feature extraction. Preliminary results demonstrate some potential for the presented approach.

1 Introduction

The problem of document classification based on semantic content (text categorization) may arise when the documents from some set have to be ranked according to their relevance to some usually predefined set of topics (i.e. classification of news articles based on their dealing with business topics). In this work we to present a new batch learning algorithm for text classification. Our method applies non-linear ellipsoid separation [3] to the vector space representation of text documents. We use the bag-of-words vector representation of text documents, the maximal separation ratio method for pattern separation via ellipsoids [3], and the approximation of the latent semantic feature extraction technique with Gram-Schmidt orthogonalization (GSK algorithm) [2]. We present some preliminary results which indicate some potential for the given approach. The rest of the paper is organized as follows: we describe the general formulation of the algorithm in Section 2, the specific problems of applying it to text documents in Section 3, and show how latent semantic feature extraction can help dealing with some of the resulting problems in Section 4, followed by numerical results in Section 5.

2 Ellipsoid Separation

Pattern classification via ellipsoids has been of interest for the learning community because it possesses a nice feature of independence from invertible linear transformations of the coordinate system, and it leads to a semidefinite program (SDP) [3] which can be efficiently solved by the state-of-the-art interior point methods. Moreover, if we consider that a set of points lying between two parallel hyperplanes is a degenerate ellipsoid then we can treat ellipsoid separation as a generalization of hyperplane separation. Ellipsoid separation is most effectively applicable to such types of binary classification problems where a set of examples with one label (e.g. positive set) is much smaller and single-clustered than the other set because in this case we can find a mapping to some feature space where we can separate the two classes by enclosing the smaller class inside the inner ellipsoid, and keeping the larger one outside the outer ellipsoid. One such area is document categorization since the class of relevant documents is usually of much smaller size than the set of all available documents. We consider an ellipsoid defined in the following way.

Definition 1. *An ellipsoid $\mathcal{E} \subset \mathbb{R}^n$ is a set of points given by its centre $\mathbf{c}$ and an $n \times n$ symmetric positive semidefinite matrix $\mathbf{E}$ such that*

$$\mathcal{E} = \{\mathbf{x} \in \mathbb{R}^n : (\mathbf{x} - \mathbf{c})^\top \mathbf{E}(\mathbf{x} - \mathbf{c}) \leq 1\}\,. \tag{1}$$

The idea of our approach is to construct two ellipsoids with the same centre, $\mathbf{c}$, and axis directions, $\mathbf{E}$, while one of them is of minimal size to include all the positive examples, and the other is of maximal size such that it does not include any example of the other class. Therefore the optimization criterion is to maximize the squared ratio between corresponding half-axes of outer and inner ellipsoids. This leads to an SDP, and if the data are separable then the SDP is feasible and bounded so its solution exists and is unique. Finally, we construct a third ellipsoid with the same center, $\mathbf{c}$, and axis directions, $\mathbf{E}$, but its half-axes are means of the half-axes of the previous two. We use the latter ellipsoid as a classification function which assigns a label to the point depending whether the point belongs to its interior or exterior (see Figure 1).

Let us assume we have computed vector representations of the set of m labeled documents $((\hat{\mathbf{d}}_1, y_1), \ldots, (\hat{\mathbf{d}}_m, y_m)) \in (\mathbb{R}^n \times \{-1, +1\})^m$ where $\hat{\mathbf{d}}_i$ are the feature vectors and y_i are the document labels for all $i \in \{1, \ldots, m\}$. Moreover, we shall use a mapping $\boldsymbol{\phi} : \hat{\mathbf{d}} \to (1, \hat{\mathbf{d}}^\top)^\top$ in order to search for the separation ellipsoid in its canonical homogeneous form, i.e. an ellipsoid in $n+1$-dimensional space centered at the origin [3]. We will denote the mapped inputs $\hat{\mathbf{d}}_i$ by $\mathbf{x}_i := \boldsymbol{\phi}(\hat{\mathbf{d}}_i)$.

We consider a class of ellipsoid classifiers $h_{t,\mathbf{E}} : \mathbb{R}^n \to \mathbb{R}$ parameterized by a symmetric positive definite matrix $\mathbf{E} \in \mathbb{R}^{(n+1)\times(n+1)}$ and some positive value $t \in \mathbb{R}_+$ (which actually is a squared separation ratio) such that

$$h_{t,\mathbf{E}}(\hat{\mathbf{d}}) := \operatorname{sign}(f_{t,\mathbf{E}}(\hat{\mathbf{d}}))\,, \qquad f_{t,\mathbf{E}}(\hat{\mathbf{d}}) := \frac{t}{2} + 1 - \mathbf{x}^\top \mathbf{E}\mathbf{x}\,.$$

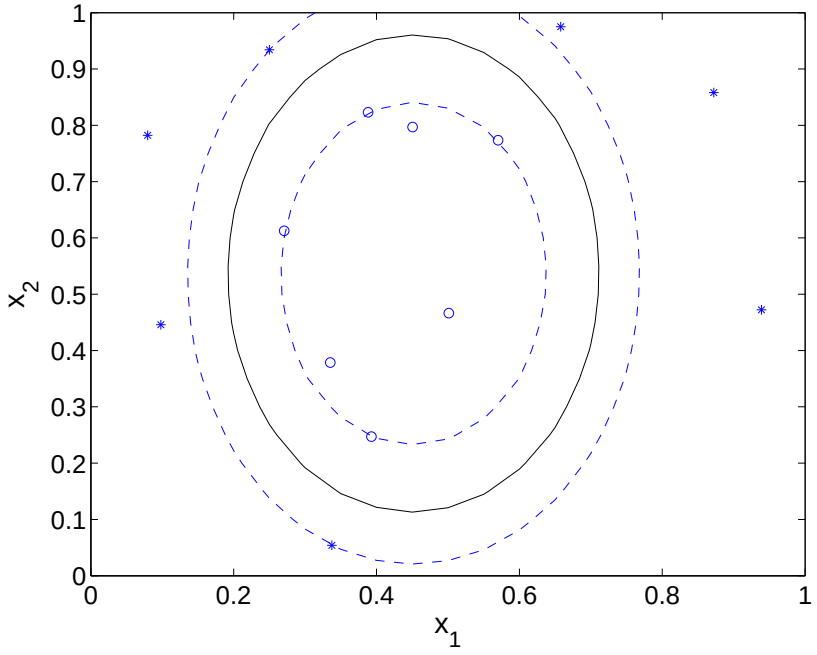

Figure 1: Example of ellipsoid separation. The maximal margin ellipsoid is plotted as a solid curve; two separating ellipsoids are plotted as dashed curves. The positive points are marked with circles and the negative ones with asterisks.

In order to find the optimal values for t and $\mathbf{E}$ we enclose all positive examples inside an inner ellipsoid, and then look for the co-centered ellipsoid of maximal size with the same axis directions to keep all negative examples outside. In other words we require $\mathbf{x}_i^\top \mathbf{E} \mathbf{x}_i \leq 1$ for all $i \in \{1, \ldots, m : y_i = +1\}$ (for positive examples) and $\mathbf{x}_i^\top \mathbf{E} \mathbf{x}_i \geq 1 + t$ for all $i \in \{1, \ldots, m : y_i = -1\}$ (for negative examples) where t is a squared distance between two separating ellipsoids in a metric space with the norm $\|\mathbf{x}\|_2 = \sqrt{\mathbf{x}^\top \mathbf{E} \mathbf{x}}$. This implies that $t + 1$ is the squared separation ratio, i.e. the ratio between corresponding half-axes of outer and inner ellipsoids. Combining these two sets of inequalities we can pose the following semidefinite maximization program:

$$\begin{array}{lll} \max\limits_{t, \mathbf{E}} & t & \\ \text{such that} & y_i(1 - \mathbf{x}_i^\top \mathbf{E} \mathbf{x}_i) \geq t\tau_{y_i}\,, & i = 1, \ldots, m\,, \\ & \mathbf{E} \succeq \mathbf{0}\,. & \end{array} \tag{2}$$

where $\tau_{+1} = 0$ and $\tau_{-1} = 1$.

In [3] the following properties of the SDP above and its solution are proved.

Algorithm 1 Gram-Schmidt Kernel (GSK) Feature Extraction Algorithm

Require: A training set $((\mathbf{d}_i, y_i), \ldots, (\mathbf{d}_m, y_m)) \in \left(\mathbb{R}^l \times \{-1, +1\}\right)^m$, bias $B \in \mathbb{R}_+$ and a dimension of the subspace, $k \in \mathbb{N}$
  for $i = 1, \ldots, m$ **do**
    $n_i = \mathbf{d}_i^\top \mathbf{d}_i$
  end for
  for $j = 1, \ldots, k$ **do**
    for $i = 1, \ldots, m$ **do**
      $b_i = B^{\frac{y_i+1}{2}} \cdot n_i$
    end for
    $i_j = \operatorname{argmax}_i b_i$
    for $i = 1, \ldots, m$ **do**
      $\hat{\mathbf{D}}_{i,j} \leftarrow \frac{1}{\sqrt{\mathbf{n}_{i_j}}} \cdot \left(\mathbf{d}_i^\top \mathbf{d}_{i_j} - \sum_{t=1}^{j-1} \hat{\mathbf{D}}_{i,t} \hat{\mathbf{D}}_{i_j,t}\right)$
      $n_i \leftarrow n_i - \hat{\mathbf{D}}_{i_j}^2$
    end for
  end for
  return matrix $\hat{\mathbf{D}}$ with the training set mapped into the feature subspace stored in its rows

Theorem 1 ([3]). *If there exists a separating ellipsoid, the optimal solution of (2) has $t > 0$ and the optimal homogeneous ellipsoid is canonical. If there is no separating ellipsoid, the optimal solution of (2) has $t = 0$.*

The fact that the separating ellipsoid is canonical in $n+1$-dimensional space (homogeneous form) means that the separation ratio is the same in both spaces, and the ellipsoid matrix $\mathbf{E}$ and center $\mathbf{c}$ in the original space can be computed using simple formulas.

Corollary 1 ([3]). *If there exists a separating ellipsoid, the semidefinite program (2) provides an ellipsoid with the highest possible separation ratio.*

Theorem 2 ([3]). *Given two sets of points to be separated, the maximal separation ratio and the ellipsoid(s) that achieve it are independent of the coordinate system.*

3 Applying Categorization to Text Documents

The most common approach used in learning for text categorization problems is mapping the set of documents to some linear metric space (feature space), and then applying learning classification techniques based on distance functions in that space. For this purpose the documents are often mapped using the so-called bag-of-words approach when the occurrence of every distinct word from the set of documents (excluding stop-words, articles, and prepositions) is counted as a separate dimension, and thus every document is represented as a vector with

Algorithm 2 Gram-Schmidt Kernel (GSK) Algorithm for New Examples

Require: A new example $\mathbf{d} \in \mathbb{R}^l$
  for $j = 1, \ldots, k$ **do**
    $\tilde{d}_j = \frac{1}{\sqrt{n_{i_j}}} \cdot \left(\mathbf{d}^\top \mathbf{d}_{i_j} - \sum_{t=1}^{j-1} \mathbf{d}_t \hat{\mathbf{D}}_{i_j,t} \right)$
  end for
  return the image $\tilde{\mathbf{d}}$ of $\mathbf{d}$ in the feature subspace

word frequencies as its components. For convenience those vectors are often normalized.

The obvious drawback of the bag-of-words vector representation of the document set is high dimensionality of the vector space because of the high number of distinct terms in the text. As a result the SDP is often computationally impractical. Moreover the dimensionality of the space is usually greater than the size of the dataset so even if a solution of the SDP can be computed it is usually degenerate since we need at least $n + 1$ points in order to define an ellipsoid in an n-dimensional space.

In order to solve this problem we need to look for some subspace of the bag-of-words vector representation of the text documents. This subspace must have much lower dimensionality while preserving the ellipsoid separability of the original space. For this purpose we need to use some feature extraction algorithm. In the next section a variant of latent semantic feature extraction will be described that meets both requirements.

4 Latent Semantic Feature Extraction

The weakness of the bag-of-words approach is its total ignorance of the occurrence of semantically similar words, e.g. synonyms. Ideally, semantically similar documents are mapped to the same directions in the feature space. Although the explicit computation of the co-occurrence of semantically similar words is a rather expensive procedure, the latent semantic indexing approach from information retrieval constructs a feature space based on semantic similarity between different words.

The Gram-Schmidt kernel (GSK) feature extraction algorithm algorithm [2] is based on the latent semantic indexing approach and it projects document feature vectors onto a subspace spanned by k representations of training examples in the feature space. The subspace is selected by applying the Gram-Schmidt orthogonalization procedure to documents in feature space. This subspace has lower dimension than the feature space, and at the same time it incorporates some semantic similarity between documents. In our algorithm, in order to construct the subspace we use a generalized GSK algorithm that has a bias towards positive examples (see Algorithms 1 and 2). The pseudo-code is taken from [2]. Note that this algorithm is equivalent to the partial Cholesky decomposition of the kernel or inner product matrix with elements $\mathbf{d}_i^\top \mathbf{d}_j$.

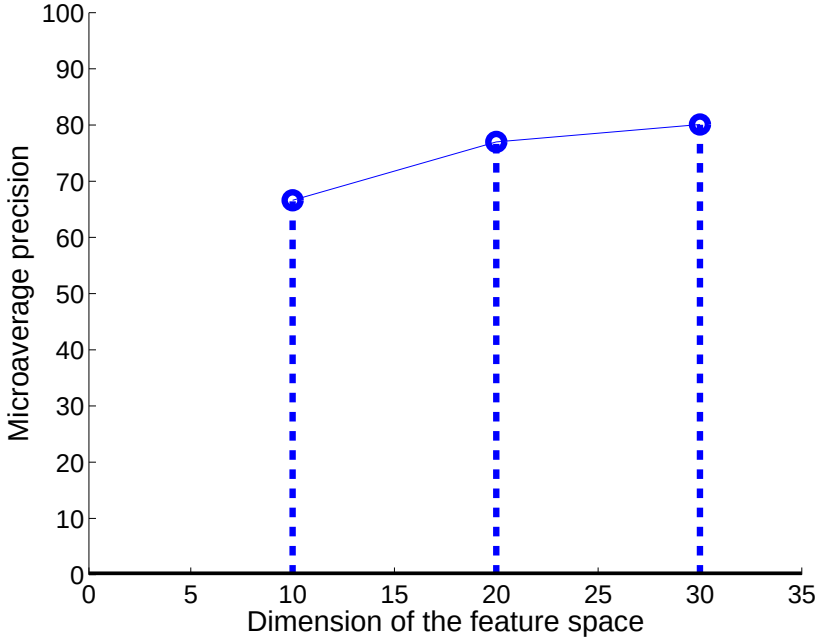

Figure 2: Choosing dimension of the feature space. The value of microaverage precision achieved with ellipsoid separation is plotted for different dimensions of the feature space for the category `acq`.

5 Preliminary Results

We have implemented the above strategy for ranking a set of test documents with respect to their relevance to the training topic. The documents were ranked according to the value of $f_{t,\mathbf{E}}(\mathbf{d})$. The quality of the ranking was assessed using microaverage precision[1]. We used a MATLAB implementation of the algorithm based on YALMIP [5] and the SDPT3 semidefinite program solver package [6]. The value of the bias B for the GSK algorithm was chosen to be equal to either the inverse fraction of positive examples in the category, or 10 whichever is smaller.

As benchmark data we used the 'Mod-Apte' split of the Reuters-21570 document collection available from the home page of David D. Lewis. The Mod-Apte sample contains 9603 training and 3299 test documents, and 90 categories. We used the 10 most popular categories for which we computed SVM classification with a second order polynomial kernel using SVM$^{\text{light}}$ [4] to compare our results with.

[1]The microaverage precision is defined as the average of all precisions computed at the threshold $f_{t,\mathbf{E}}(\mathbf{d})$ where $\mathbf{d}$ ranges over all positive documents only. The precision at some threshold u is the fraction of positive documents among all documents for which $f_{t,\mathbf{E}}(\mathbf{d}) \geq u$.

	Ellipsoid ($k = 30$)	SVM ($k = 30$)	SVM ($n = 20494$)
earn	**97.6**	**97.6**	99.7
acq	80.9	**85.0**	98.6
money-fx	58.9	**75.1**	80.9
grain	89.0	**94.8**	98.4
crude	82.1	**90.4**	96.1
trade	46.7	**80.3**	84.3
interest	56.0	**77.1**	87.1
ship	79.6	**84.5**	92.8
wheat	88.7	**93.9**	93.1
corn	84.9	**90.9**	92.2

Table 1: Comparison of the ellipsoid separation against SVM on ten categories of the Reuters-21578 dataset using microaverage precision (in percent) as a performance measure. The ellipsoid separation was done on $n = 30$ features extracted using the GSK algorithm (second column), and the SVM classification was performed on both: the same set 30 features (third column) and the whole set of 20494 features (fourth column).

In Figure 2 we present our results for a different numbers of dimensions of the feature subspace for the category `acq`. We stopped increasing the value of dimensions n at $n = 30$ because of the computational complexity of the algorithm. The average CPU time of the algorithm performance on one category on a Pentium 4, 2.2GHz machine with 512 MB RAM was less than 4 hours.

In Table 1 we compare our results obtained with $n = 30$ dimensions of feature space with the best ones obtained by SVM classification with a second order polynomial kernel. As one can see we reached the accuracy of the SVM in one case, performed slightly worse in 6 cases and much worse on the `trade`, `interest` and `money-fx` categories.

Figure 2 indicates that increasing the dimension a little has the potential to further improve the performance. This is not practical using our current implementation for complexity reasons. We anticipate, however, that using a chunking approach similar to the one adopted for the SVM, we should be able to scale the algorithm to higher dimensional feature spaces. This will be the subject of further research.

6 Conclusions

The technique proposed is attractive theoretically in that it attempts to place an ellipsoid in feature space. Preliminary experiments are encouraging since they demonstrate that the algorithm can perform document classification up to the level of the state-of-the-art SVM algorithm. Further research is needed to investigate the method and its strengths and weaknesses, in particular using

non-linear inner product functions (i.e., kernels) and introducing soft-margins similarly to SVMs (see [1]). It will be of particular interest to test its performance on the categories with very few positive examples. The low dimensionality and ellipsoid approach would appear to be suited to this type of problem.

Acknowledgments We would like to thank Huma Lodhi for technical assistance in working with Reuters-21578 dataset, Yaoyong Li for interesting discussions, and Sándor Szedmák for his help in dealing with some technical problems.

References

[1] C. Cortes and V. Vapnik. Support vector networks. *Machine Learning*, 20:273–297, 1995.

[2] N. Cristianini, J. Shawe-Taylor, and H. Lodhi. Latent semantic kernels. *Journal of Intelligent Information Systems*, 18(2/3):127–152, 2002.

[3] F. Glineur. Pattern separation via ellipsoids and conic programming. Mémoire de D.E.A., Faculté Polytechnique de Mons, Mons, Belgium, Sept. 1998.

[4] T. Joachims. Making large-scale support vector machine learning practical. In B. Schölkopf, C. Burges, and A. Smola, editors, *Advances in Kernel Methods: Support Vector Machines*. MIT Press, Cambridge, MA, 1999.

[5] J. Löfberg. *YALMIP. Yet another LMI parser*, 2003.

[6] K. C. Toh, M. J. Todd, and R. H. Tütüncü. SDPT3 – a MATLAB software package for semidefinite programming. Technical Report TR1177, Cornell University, 1996.