Feature Selection for Unsupervised and Supervised Inference: the Emergence of Sparsity in a Weighted-based Approach

Lior Wolf and Amnon Shashua
School of Engineering and Computer Science,
The Hebrew University,
Jerusalem 91904, Israel

Abstract

The problem of selecting a subset of relevant features in a potentially overwhelming quantity of data is classic and found in many branches of science including — examples in computer vision, text processing and more recently bio-informatics are abundant. In this work we present a definition of ”relevancy” based on spectral properties of the Affinity (or Laplacian) of the features' measurement matrix. The feature selection process is then based on a continuous ranking of the features defined by a least-squares optimization process. A remarkable property of the feature relevance function is that sparse solutions for the ranking values naturally emerge as a result of a “biased non-negativity” of a key matrix in the process. As a result, a simple least-squares optimization process converges onto a sparse solution, i.e., a selection of a subset of features which form a local maxima over the relevance function. The feature selection algorithm can be embedded in both unsupervised and supervised inference problems and empirical evidence show that the feature selections typically achieve high accuracy even when only a small fraction of the features are relevant.

1. Introduction

As visual recognition, text classification, speech recognition and more recently bio-informatics aim to address larger and more complex tasks the problem of focusing on the most relevant information in a potentially overwhelming quantity of data has become increasingly important. Examples from computer vision, text processing and Genomics are abundant. For instance, in visual recognition the pixel values themselves often form a highly redundant set of features; methods using an “over-complete” basis of features for recognition are gaining popularity [16], and recently methods relying on abundance of simple efficiently computable features of which only a fraction of are relevant were proposed for face detection [23] — and these are only few examples from the visual recognition literature.

From a practical perspective, large amounts of irrelevant features affects learning algorithms at three levels. First, most learning problems do not scale well with the growth of irrelevant features — in many cases the number of training examples grows exponentially with the number of irrelevant features [11]. Second, is a substantial degradation of classification accuracy for a given training set size [1, 9]. The accuracy drop affects also advanced learning algorithms that generally scale well with the dimension of the feature space such as the Support Vector Machines (SVM) as recently observed in [25]. The third aspect has to do with the run time of the learning algorithm on test instances. In most learning problems the classification process is based on inner-products between the features of the test instance and stored features from the training set, thus when the number of features is overwhelmingly large the run-time of the learning algorithm becomes prohibitively large for real time applications, for example. Another practical consideration is the problem of determining how many relevant features to select. This is a difficult problem which is hardly ever addressed in the literature and consequently it is left to the user to choose manually the number of features. Finally, there is an issue of whether one is looking for the *minimal* set of (relevant) features, or simply a possibly redundant but relevant set of features.

The potential benefits of feature selection include, first and foremost, better accuracy of the inference engine and improved scalability (defying the curse of dimensionality). Secondary benefits include better data visualization and understanding, reduce measurement and storage requirements, and reduce training and inference time. Blum and Langley [2] in a survey article distinguish between three types of methods: *Embedded*, *Filter* and *Wrapper* approaches. The filter methods apply a preprocess which is independent of the inference engine (a.k.a the predictor or the classification/inference engine) and select features by ranking them with correlation coefficients or make use of mutual information measure. The Embedded and Wrapper approaches construct and select feature subsets that are useful to build a good predictor. The issue being the notion of *relevancy*, i.e., what constitutes a good set of features. The modern approaches, therefore, focus on building feature selection algorithms in the context of a *specific* inference engine. For example, [25, 3] use the Support Vector Machine (SVM) as a subroutine (wrapper) in the feature selection process with the purpose of optimizing the SVM accuracy on the resulting subset of features. These wrapper and embedded methods in general are typically computationally expensive and often criticized as being "brute force". Further details on relevancy versus usefulness of features and references to historical and modern literature on feature selection can be found in the survey papers [2, 10, 8].

In this paper the inference algorithm is not employed directly in the feature selection process but instead general properties are being gathered which indirectly indicate whether a feature subset would be appropriate or not. Specifically, we use clustering as the predictor and use spectral properties of the candidate feature subset to guide the search. This leads to a "direct" approach where the search is conducted on the basis of optimizing desired spectral properties rather than on the basis of explicit clustering and prediction cycles. The search is conducted by the solution of a least-squares optimization function using a weighting scheme for the ranking of features. *A remarkable property of the energy function is that sparse solutions for the weights naturally emerge as a result of a "biased non-negativity" of a key matrix in the process*. The algorithm, called $Q - \alpha$, is iterative, very efficient and achieves remarkable performance on a variety of experiments we have conducted.

There are several benefits of our approach: First, we avoid the expensive computations associated with Embedded and Wrapper approaches, yet still make use of a predictor to guide the feature selection. Second, the framework can handle both unsupervised and supervised inference within the same framework and handle any number of classes. In other words, since the underlying inference is based on clustering class labels are not necessary, but on the other hand, when class labels are provided they can be used by the algorithm to provide better feature selections. Third, the algorithm is couched within a least-squares framework — and least-squares problems are the best understood and easiest to handle. Finally, the performance (accuracy) of the algorithm is remarkable.

2 Algebraic Definition of Relevancy

A key issue in designing a feature selection algorithm in the context of an inference is defining the notion of relevancy. Definitions of relevancy proposed in the past [2, 10] lead naturally to a explicit enumeration of feature subsets which we would like to avoid. Instead, we take an algebraic approach and measure the relevance of a subset of features against its influence on the cluster arrangement of the data points with the goal of introducing an energy function which receives its optimal value on the desired feature selection. We will consider below a measure of relevancy based on the Standard spectrum — the use of the Laplacian spectrum is detailed in [26].

Consider a data set M consisting of column vectors (data points) $\mathbf{M}_1, ..., \mathbf{M}_q$ each a vector in R^n representing n features $x_1, ..., x_n$. Let the row vectors of M be denoted by $\mathbf{m}_1^\top, ..., \mathbf{m}_n^\top$ pre-processed such that $\sum_i \mathbf{m}_i = 0$ and normalized to unit norm $\|\mathbf{m}_i\| = 1$. Let $\mathcal{S} = \{x_{i_1}, ..., x_{i_l}\}$ be a subset of (relevant) features from the set of n features and let $\alpha_i \in \{0, 1\}$ be the indicator value associated with feature x_i, i.e., $\alpha_i = 1$ if $x_i \in \mathcal{S}$ and zero otherwise. Let A_s be the corresponding *affinity* matrix whose (i, j) entries are the inner-product between the i'th and j'th data points restricted to the selected coordinate features, i.e., $A_s = \sum_{j=1}^{l} \alpha_{i_j} \mathbf{m}_{i_j} \mathbf{m}_{i_j}^\top$ where $\mathbf{m}_i \mathbf{m}_i^\top$ is the rank-1 matrix defined by the outer-product between $\mathbf{m}_i$ and itself. Finally, let Q_s be a $q \times k$ matrix whose columns are the first k eigenvectors of A_s associated with the highest eigenvalues $\lambda_1 \geq ... \geq \lambda_k$.

We define "relevancy" as directly related to the clustering quality of the data points restricted to the selected coordinates. In other words, we would like to measure the quality of the subset $\mathcal{S}$ in terms of cluster coherence of the first k clusters, i.e., we make a direct linkage between cluster coherence of the projected data points and relevance of the selected coordinates.

We measure cluster coherence by analyzing the (standard) spectral properties of the affinity matrix A_s. Considering the affinity matrix as representing weights in an undirected graph, it is known that maximizing the quadratic form $\mathbf{x}^\top A_s \mathbf{x}$ where $\mathbf{x}$ is constrained to lie on the standard simplex ($\sum x_i = 1$ and $x_i \geq 0$) provides the identification of the maximal *clique* of the (unweighted) graph [14, 6], or the maximal "dominant" subset of vertices of the weighted graph [17]. Likewise there is evidence (motivated by finding cuts in the graph) that solving the quadratic form above where $\mathbf{x}$ is restricted to the unit sphere provides cluster membership information (cf. [15, 24, 18, 20, 4, 5]). In this context, the eigenvalue (the value of the quadratic form) represents the cluster coherence. In the case of k clusters, the highest k eigenvalues of A_s represent the corresponding cluster coherences and the components of an eigenvector represent the coordinate (feature) participation in the corresponding cluster. The eigenvalues decrease as the interconnections of the points within clusters get sparser (see [19]). Therefore, we define the relevance of the subset $\mathcal{S}$ as:

$$
\begin{aligned}
&rel(\mathcal{S}) = trace(Q_s^\top A_s^\top A_s Q_s) \\
&= \sum_{r,s} \alpha_{i_r} \alpha_{i_s} (\mathbf{m}_{i_r}^\top \mathbf{m}_{i_s}) \mathbf{m}_{i_r}^\top Q_s Q_s^\top \mathbf{m}_{i_s} = \sum_{j=1}^{k} \lambda_j^2,
\end{aligned}
$$

where λ_j are the ordered eigenvalues of A_s. Note that the proposed measure of relevancy handles interactions among features up to a second order. To conclude, achieving a high score on the combined energy of the first k eigenvalues of A_s indicate (although indirectly) that the q input points projected onto the l-dimensional feature space are "well clustered" and that in turn suggests that $\mathcal{S}$ is a relevant subset of features.

Rather than enumerating all possible feature subsets $\mathcal{S}$ and ranking them according to the value of $rel(\mathcal{S})$ we consider the prior weights $\alpha_1, ..., \alpha_n$ as unknown *real numbers* and define the following optimization function:

Definition 1 (Relevant Features Optimization) *Let M be an $n \times q$ input matrix with rows $\mathbf{m}_1^\top, ..., \mathbf{m}_n^\top$. Let $A_\alpha = \sum_{i=1}^n \alpha_i \mathbf{m}_i \mathbf{m}_i^\top$ for some unknown scalars $\alpha_1, ..., \alpha_n$. The weight vector $\alpha = (\alpha_1, ..., \alpha_n)^\top$ and the orthonormal $q \times k$ matrix Q are determined at the maximal point of the following optimization problem:*

$$\max_{Q,\alpha_i} trace(Q^\top A_\alpha^\top A_\alpha Q) \qquad (1)$$

$$subject\ to \qquad \sum_{i=1}^n \alpha_i^2 = 1, \;\; Q^\top Q = I$$

Note that the optimization function does not include the inequality constraint $\alpha_i \geq 0$ and neither a term for "encouraging" a sparse solution of the weight vector α — both of which are necessary for a "feature selection". As will be shown later in Section 4, the sparsity and positivity conditions are implicitly embedded in the nature of the optimization function and therefore "emerge" naturally with the optimal solution.

Note also that it is possible to maximize the gap $\sum_{i=1}^k \lambda_i^2 - \sum_{j=k+1}^q \lambda_j^2$ by defining $Q = [Q_1|Q_2]$ where Q_1 contains the first k eigenvectors and Q_2 the remaining $q-k$ eigenvectors (sorted by decreasing eigenvalues) and the criterion function (1) would be replaced by:

$$\max_{Q=[Q_1|Q_2],\alpha_i} trace(Q_1^\top A_\alpha^\top A_\alpha Q_1) - trace(Q_2^\top A_\alpha^\top A_\alpha Q_2).$$

We will describe in Section 3 an efficient algorithm for finding a local maximum of the optimization (1) and later address the issue of sparsity and positivity of the resulting weight vector α. The algorithms are trivially modified to handle the gap maximization criterion and those will not be further elaborated here. We will describe next the problem formulation using an additive normalization (the Laplacian) of the affinity matrix.

3 An Efficient Algorithm

We wish to find an optimal solution for the non-linear problem (1). We will focus on the Standard spectrum matrix A_α and later discuss the modifications required for L_α. If the weight vector α is known, then the solution for the matrix Q is readily available by employing a Singular Value Decomposition (SVD) of the symmetric (and positive definite) matrix A_α. Conversely, if Q is known then α is readily determined as shown next. We already saw that

$$\begin{aligned} trace(Q^\top A_\alpha^\top A_\alpha Q) &= \sum_{i,j} \alpha_i \alpha_j (\mathbf{m}_i^\top \mathbf{m}_j) \mathbf{m}_i^\top Q Q^\top \mathbf{m}_j \\ &= \alpha^\top G \alpha \end{aligned}$$

where $G_{ij} = (\mathbf{m}_i^\top \mathbf{m}_j)\mathbf{m}_i^\top Q^\top Q \mathbf{m}_j$ is symmetric and positive definite. The optimal α is therefore the solution of the optimization problem:

$$\max_\alpha \alpha^\top G \alpha \qquad subject\ to\ \alpha^\top \alpha = 1,$$

which results in α being the eigenvector of G associated with its largest eigenvalue. A possible scheme, guaranteed to converge to a local maxima, is to start with some initial guess for α and iteratively interleave the computation of Q given α and the computation of α given Q until convergence. We refer to this scheme as the **Basic** $Q-\alpha$ **Method**.

A more advanced scheme with superior convergence rate and more importantly accuracy of results (based on empirical evidence) is to embed the computation of α within the "orthogonal iteration" [7] cycle for computing the largest k eigenvectors, described below:

Definition 2 (Standard Power-Embedded $Q-\alpha$ **Method)** *Let M be an $n \times q$ input matrix with rows $\mathbf{m}_1^\top, ..., \mathbf{m}_n^\top$, and some orthonormal $q \times k$ matrix $Q^{(0)}$, i.e., ${Q^{(0)}}^\top Q^{(0)} = I$. Perform the following steps through a cycle of iterations with index $r = 1, 2, ...$*

1. *Let $G^{(r)}$ be a matrix whose (i,j) components are $(\mathbf{m}_i^\top \mathbf{m}_j)\mathbf{m}_i^\top Q^{(r-1)} {Q^{(r-1)}}^\top \mathbf{m}_j$.*
2. *Let $\alpha^{(r)}$ be the largest eigenvector of $G^{(r)}$.*
3. *Let $A^{(r)} = \sum_{i=1}^n \alpha_i^{(r)} \mathbf{m}_i \mathbf{m}_i^\top$.*
4. *Let $Z^{(r)} = A^{(r)} Q^{(r-1)}$.*
5. *$Z^{(r)} \xrightarrow{QR} Q^{(r)} R^{(r)}$.*
6. *Increment index r and go to step 1.*

The method is very efficient and achieves very good performance (accuracy). Note that steps 4,5 of the algorithm consist of the "orthogonal iteration" module, i.e., if we were to repeat steps 4,5 *only* we would converge onto the eigenvectors of $A^(r)$. However, note that the algorithm does not repeat steps 4,5 in isolation and instead recomputes the weight vector α (steps 1,2,3) before applying another cycle of steps 4,5. The convergence proof, a faster converging method using the "Ritz" acceleration [7] to the basic power method and the manner in which *supervised* inference can be handled in this framework can be found in [26].

4 Sparsity and Positivity of α

The optimization criteria (1) is formulated as a least-squares problem and as such there does not seem to be any apparent guarantee that the weights $\alpha_1, ..., \alpha_n$ would come out *non-negative* (same sign condition), and in particular *sparse* when there exists a sparse solution (i.e., there is a relevant subset of features). These two conditions are critical for the compatibility of the algorithm for feature selection. The positivity is required for making the variables α_i serve as weights, and the sparsity for the feature selection itself — otherwise the scheme would produce some feature combination rather than feature selection.

Typically, these conditions should be specifically presented into the optimization criterion one way or the other. The possible means for doing so include introduction of inequality constraints, use of L_0 or L_1 norms, adding specific terms to the optimization function to "encourage" sparse solutions or use a multiplicative scheme of iterations which preserve the sign of the variables throughout the iterations (for a very partial list see [16, 12, 22]). It is therefore somewhat surprising, if not remarkable, that the least-squares formulation of the feature selection problem could consistently converge onto same-sign and sparse solutions.

The key for the emergence of a sparse and positive α has to do with the way the entries of the matrix G are defined. Recall that $G_{ij} = (\mathbf{m}_i^\top \mathbf{m}_j)\mathbf{m}_i^\top Q^\top Q \mathbf{m}_j$ and that α comes out as the largest eigenvector of G (at each iteration). If G were to be non-negative (and irreducible), then from the Perron-Frobenious theorem the first eigenvector is guaranteed to be non-negative (or same-sign). However, this is not the case and G in general has negative terms as well as positive ones. A closer look shows that each entry of G consists of a sum of products of three inner-products:

$$G_{ij} = \sum_{l=1}^{k} (\mathbf{m}_i^\top \mathbf{q}_l)(\mathbf{m}_j^\top \mathbf{q}_l)(\mathbf{m}_i^\top \mathbf{m}_j).$$

In general, a product of the form $f = (\mathbf{a}^\top \mathbf{b})(\mathbf{a}^\top \mathbf{c})(\mathbf{b}^\top \mathbf{c})$, where $\|\mathbf{a}\| = \|\mathbf{b}\| = \|\mathbf{c}\| = 1$ satisfies $-1 < f \leq 1$ where $f = 1$ when $\mathbf{a} = \mathbf{b} = \mathbf{c}$. Since $f > -1$ there is an asymmetry on the expected value of f, i.e., the expected values of the entries of G are biased towards a positive value — and we should expect a bias towards a positive first eigenvector of G. In the context of deriving the probability that the first eigenvector of G is positive we will address the following three questions:

- What is the minimal value of $f = (\mathbf{a}^\top \mathbf{b})(\mathbf{a}^\top \mathbf{c})(\mathbf{b}^\top \mathbf{c})$ when $\mathbf{a}, \mathbf{b}, \mathbf{c}$ vary over the n-dimensional unit hypersphere? We will show that the $-1/8 \leq f \leq 1$.
- Given a uniform sampling of the vectors $\mathbf{a}, \mathbf{b}, \mathbf{c}$ over the n-dimensional unit hypersphere, what is the mean μ and variance σ^2 of f? The result that $-1/8 \leq f \leq 1$ suggests that $\mu > 0$.
- Given that $G_{ij} \sim N(\mu > 0, \sigma^2)$ sampled i.i.d, what is the probability (as a function of n) that the first eigenvector of G is strictly non-negative (same sign)?

We will show that for a random matrix G, the probability of the leading eigenvector α of G to be strictly non-negative rapidly approaches 1 with n.

Proposition 1 *The minimal value of* $f = (\mathbf{a}^\top \mathbf{b})(\mathbf{a}^\top \mathbf{c})(\mathbf{b}^\top \mathbf{c})$ *where* $\mathbf{a}, \mathbf{b}, \mathbf{c} \in R^n$ *are defined over the unit hypersphere is* $-1/8$.

Proof: Let $\mathbf{e}_1, \mathbf{e}_2, \mathbf{e}_3 \in R^n$ be three units vectors $(1, 0, ..., 0)$, $(0, 1, 0, ..., 0)$ and $(0, 0, 1, 0, ..., 0)$. The parameterization of 3 points on the unit hypersphere takes the form:

$$[\mathbf{a}, \mathbf{b}, \mathbf{c}] = [\mathbf{e}_1, \mathbf{e}_2, \mathbf{e}_3] \begin{bmatrix} 1 & \cos(\beta) & \cos(\gamma_1) \\ 0 & \sin(\beta) & \sin(\gamma_1)\cos(\gamma_2) \\ 0 & 0 & sin(\gamma_1) sin(\gamma_2) \end{bmatrix} \tag{2}$$

Setting the partial derivatives of

$$f = \cos(\beta)\cos(\gamma_1)\left(\cos(\beta)\cos(\gamma_1) + \sin(\beta)\sin(\gamma_1)\cos(\gamma_2)\right) \tag{3}$$

with respect to $\beta, \gamma_1, \gamma_2$ to zero and solving for the extremum points (using a symbolic solver such as Maple) yields 36 solutions for the triplet $(\beta, \gamma_1, \gamma_2)$. When these solutions are substituted in expression (3) the values $f = \{-1/8, 0, 1\}$ appear with multiplicity $\{16, 14, 4\}$, respectively. □

Proposition 2 *The expected value of* $f = (\mathbf{a}^\top \mathbf{b})(\mathbf{a}^\top \mathbf{c})(\mathbf{b}^\top \mathbf{c})$ *where* $\mathbf{a}, \mathbf{b}, \mathbf{c} \in R^n$ *are uniformly sampled over the unit hypersphere is* $\mu = \frac{1}{6}$ *with a standard deviation (s.t.d)* $\sigma = \sqrt{2}\frac{1}{6}$.

Proof: Let the parameterization of 3 points on the unit hypersphere be described as in (2), where $0 \leq \beta \leq 2\pi$, $-1 \leq \cos(\gamma_1) \leq 1$ and $0 \leq \gamma_2 \leq \pi$ are sampled uniformly inside their respective interval domains. This parameterization guarantees a uniform sampling of all the unit direction triplets which is invariant to rotation. For instance, a uniform sampling of γ_1 would have resulted in a bias (at the poles) which can be fixed by sampling $\cos(\gamma_1)$ uniformly instead (as can be verified by deriving the Jacobian of the joint distribution). The expectation μ can be computed by the following integral:

$$\begin{aligned} \mu &= \frac{1}{4\pi}\int_0^{\pi}\int_{-1}^{1}\int_0^{2\pi} (\mathbf{a}^\top \mathbf{b})(\mathbf{a}^\top \mathbf{c})(\mathbf{b}^\top \mathbf{c}) d\gamma_2 d(cos(\gamma_1))d\beta \\ &= \frac{1}{4\pi}\int_0^{\pi}\int_{-1}^{1}\int_0^{2\pi} cos(\beta)cos(\gamma_1)(cos(\beta)cos(\gamma_1) \\ &+ sin(\beta)\sqrt{1 - cos(\gamma_1^2)}cos(\gamma_2))d\gamma_2 dcos(\gamma_1)d\beta = \frac{1}{6} \end{aligned}$$

The s.t.d of the distribution can be similarly computed with the result of $\sigma = \sqrt{2}\frac{1}{6}$. □

Each entry G_{ij} is a a sum of k such terms, each with a mean of $1/6$ and s.t.d $\sqrt{2}(1/6)$, therefore the mean of G_{ij} is $k(1/6)$ with s.t.d $\sqrt{2k}(1/6)$. In the sequel we will take the worst case where $k = 1$. Next, we address the probability that a matrix G whose entries are normally distributed $N(\mu, \sigma^2)$ will have a strictly non-negative first eigenvector. The theorem is a result of joint work with Ofer Zeitouni from U. of Minnesota and Michael Ben-Or from the Hebrew U.

Theorem 1 (Weak Version) *Let G be a symmetric $n \times n$ matrix whose entries are drawn i.i.d from a Normal distribution $G_{ij} \sim N(\mu > 0, \sigma^2)$. Let $\mathbf{v}_1$ be the leading eigenvector of G. There exists σ_0 which depends on μ such that for all $\sigma \le \sigma_0$,*

$$P(\mathbf{v}_1 \ge 0) \to_{n\to\infty} 1.$$

In other words, the probability that the entries of the leading eigenvector are non-negative approaches unity with increasing size of n.

Proof: Let $G = \mu J + \sigma S$ where $J = \mathbf{1}\mathbf{1}^\top$ and S_{ij} are i.i.d. sampled according to $N(0,1)$. Let $\mathbf{e} = \frac{1}{\sqrt{n}}\mathbf{1}$. and let $\mathbf{v}_1, ..., \mathbf{v}_n$ and $\lambda_1, ..., \lambda_n$ be the spectrum of G. It is known that $\lambda_i = O(\sqrt{n})$ for $i = 2, 3..., n$. For λ_1 we can assert the following bound:

$$\mu n - O(1) \le \lambda_1 \le \mu n + O(\sqrt{n})$$

(see [26] for derivation). Since $\mathbf{v}_i$, $i = 1, ..., n$ form an orthonormal basis, let $\mathbf{e} = \sum_i a_i \mathbf{v}_i$ and since $\mathbf{e}$ and the eigenvectors are of unit norm we have $\sum_i a_i^2 = 1$. We have therefore $\mathbf{e}^\top G \mathbf{e} = \sum_i \lambda_i a_i^2$. Since $\lambda_i = O(\sqrt{n})$ for $i = 2, ..., n$ and $\sum_i a_i^2 = 1$ we have: $\mathbf{e}^\top G \mathbf{e} \le \lambda_1 a_1^2 + O(\sqrt{n})$. Using the bound $\mathbf{e}^\top G \mathbf{e} \ge \mu n - O(1)$, we have:

$$\mu n - O(1) \le \lambda_1 a_1^2 + O(\sqrt{n})$$
$$\frac{\mu n - O(\sqrt{n})}{\mu n + O(\sqrt{n})} \le a_1^2 \le a_1$$

from which we can conclude (with further manipulation):

$$1 - \frac{1}{\mu O(\sqrt{n})} \le a_1.$$

Consider now that a_1 is the angle between $\mathbf{e}$ and $\mathbf{v}_1$:

$$\frac{1}{\sqrt{n}} \sum_i v_{1_i} = \mathbf{e}^\top \mathbf{v}_1 = a_1 \ge 1 - \frac{1}{\mu O(\sqrt{n})},$$

from which we obtain:

$$\sum_i v_{1_i} \ge \sqrt{n} - O(1).$$

Finally, assume that $v_{1_i} < 0$, then this implies:

$$\begin{aligned} 0 > \lambda v_{1_i} = (G\mathbf{v}_1)_i &= \mu \sum_i v_{1_i} + \sigma(S\mathbf{v})_i \\ &\ge \mu\sqrt{n} - O(1) + \sigma(S\mathbf{v})_i \end{aligned}$$

By concentration inequalities for the Gaussian process (e.g., Talagrand's [21]),

$$P(\exists w : \|w\|_2 = 1, \sigma|(Sw)_i| > \mu\sqrt{n}) \le e^{-Cn}$$

where the constant C depends on μ and σ and in particular we should have $\sigma < \sigma_0$ where σ_0 depends on μ. Thus, the probability for a particular entry i of the leading eigenvector decays exponentially in n and since there are n possibilities for i, the probability that there is a negative entry decays exponentially as well. □

A much stronger theorem can be proven (but not shown here) of the claim that the probability of positive leading eigenvector approaches unity regardless of the value of σ. In other words, for any fixed positive value of μ the probability increases with the value of n. For the value of $\mu = 1/6$ and $\sigma = \sqrt{2}/6$ the probability we acheive in simulations becomes very close to 1 once $n > 20$ (see Fig. 1a). On the other hand, for $\mu = 0.1$ and $\sigma = 1$ the value of n must exceed 500 in order for the probability to be close to 1.

Regarding the issue of sparsity of the weight vector α., It has been observed in the past that the key for sparsity lies in the positive combination of terms (cf. [12]) — therefore there is a strong (somewhat anecdotal) relationship between the positivity of α and the sparsity feature. In [26] we establish the relationship between the "sparsity gap" and the fraction of relevant features $0 < p \le 1$. We show there that the gap between the high and low values of α_i is inversely proportional to the value of p. In other words, the sparsity result is significant when the ratio between the number of relevant and irrelevant features is high.

In the next section we will present a number of experiments, both synthetic and with real data. Fig. 1b shows the weight vector α for a random data matrix M, and for a synthetic experiment (6 relevant features out of 202) described in the next section. One can clearly observe the positivity and sparsity of the recovered weight vector — even for a random matrix.

5 Experiments

Synthetic Data

We compared the $Q - \alpha$ algorithm with three classical filter methods (Pearson correlation coefficients, Fisher criterion score and the Kolmogorov-Smirnoff test), standard SVM and the wrapper method using SVM of [25]. The data set

(a) (b)

Figure 1: (a) Probability of positive leading eigenvector of the matrix G in simulations. The probability is very close to 1 starting from $n = 20$. (b) Positivity and sparsity demonstrated on the synthetic feature selection problem described in Section 5 (6 relevant features out of 202) and of a random data matrix. The alpha weight vector (sorted for display) comes out positive and sparse.

we used follow precisely the one described in [25] which was designed for supervised 2-class inference. In [25] two experiments were designed, one with 6 relevant features out of 202 referred to as “linear” problem, and the other experiment with 2 relevant features out of 52 designed in a more complex manner and referred to as “non-linear” problem. In the linear data the class label $y \in \{-1, 1\}$ was drawn at equal probability. The first six features were drawn as $x_i = yN(i, 1)$, $i = 1..3$, and $x_j = N(0, 1)$, $j = 4..6$ at probability 0.7, otherwise they were drawn as $x_i = N(0, 1)$, $i = 1..3$, and $x_j = yN(i - 3, 1)$, $j = 4..6$. The remaining 196 dimensions were drawn from $N(0, 20)$. The reader is referred to [25] for details of the non-linear experiment. We ran $Q - alpha$ on the two problems once with known classes (supervised version) and with unknown class labels (unsupervised version). In the supervised case the selected features were used to train an SVM and in the unsupervised case the class labels were not used for the $Q - \alpha$ feature selection but were used for the SVM training. The unsupervised test appears artificial but is important for appreciating the strength of the approach as the results of the unsupervised are only slightly inferior to the supervised test. In Fig. 2a we *overlay* the $Q - \alpha$ results (prediction error of the SVM on a testing set) on the figure obtained by [25]. The performance of the supervised $Q - \alpha$ closely agrees with the performance of the wrapper SVM feature selection of [25]. The performance of the unsupervised version does not fall much behind. Similar results were obtained for the non-linear problem but are omitted due to lack of space. Additional simulations can be found in [26].

Figure 2: Comparison of feature selection methods following [25]. Performance curves of $Q - \alpha$ were overlaid on the figure adapted from [25]. The x-axis is the number of training points and the y-axis is the test error as a fraction of test points. The thick solid lines correspond to the $Q - \alpha$ supervised and unsupervised methods (see text for details).

Real Image Unsupervised Feature Selection

The strength of the $Q - \alpha$ method is that it applies for unsupervised settings as well as supervised. An interesting unsupervised feature selection problem in the context of visual processing is the one of automatic selection of relevant features which discriminate among perceptual classes. Assume one is given a collection of images where some of them contain pictures of a certain object class (say, green frogs (*Rana clamitans* specie)) and other images contain pictures of a different class of objects (say, American toads) — see Fig. 3. We would like to automatically, in an unsupervised manner, select the relevant features such that a new picture could be classified to the correct class membership.

The features were computed by matching patches of equal size of 20×20 pixels in the following manner. Assuming that the object of interest lies in the vicinity of the image center, we defined 9 “template” patches arranged in a 3×3 block centered at the image. We had 27 images (18 from one class and 9 from the other), which in turn defines $27 * 9 = 243$ feature coordinates. Each image was sampled by 49 “candidate” patches (covering the entire image) where each of the 243 template patches was matched against the 49 patches in its respective image and the score of the best match was recorded in 243×27 data matrix. The matching between a pair of patches was based on L_1-distance between the respective color histograms in HSV space. The resulting α weight vector forms a feature selection from which we create a submatrix of data points and construct its affinity matrix and the associated matrix of eigenvectors Q. The rows of the Q matrix were clustered using k-means into two clusters. A test image was classified based on distance from the cluster centroids. Performance on test images varied between 80% to 90% correct classi-

Figure 3: Image samples of several animal classes — American toad (top row) and Green frogs (*Rana clamitans*), elephants, and sea elephants. The objects appear in various positions, illumination, context and size.

(a) (b) (c)

(d) (e) (f)

Figure 4: Unsupervised feature selection for automatic object discrimination from images. (a),(b) the first 20 features from pictures containing the American frog and the Green frog ranked by the α weight vector. (c) the (sorted) α values. (d),(e),(f) similar to the elephant and sea elephant.

fication over many experiments over several object classes (including elephants, sea elephants, and so forth). This performance was compared to spectral clustering using all the 243 features which provided a range of 55% to 65% correct classification.

Fig. 4a and Fig. 4b show the 20 most relevant templates selected for the two classes, and Fig. 4c shows the alpha values. Note that the α weights are positive as predicted from Theorem 1 and exhibit a sharp break when the relevant features begin (sparsity).

References

[1] H. Almuallim and T .G .Dietterich. Learning with many irrelevant features. *Proc. 9th Nat. Conf. on AI*, 1991.

[2] A. Blum and P. Langley. Selection of relevant features and examples in machine learning. *AI*, 97(1-2), 1997.

[3] P.S. Bradley and O.L. Mangasarian feature selection via concave minimization and support vector machines *ICML*, 1998

[4] M. Brand and K. Huang. A unifying theorem for spectral embedding and clustering In *9th Int. Conf. on AI and Statistics*, 2002.

[5] F.R.K. Chung. *spectral graph theory*. AMS, 1998.

[6] L.E. Gibbons, D.W. Hearn, P.M. Pardalos, and M.V. Ramana. Continuous characterizations of the maximum clique problem. *Math. Oper. Res.*, 22:754–768, 1997.

[7] G. Golub and C.V. Loan, *Matrix Computations*, 3rd ed., Johns Hopkins University Press, Baltimore, MD, 1996.

[8] I. Guyon and A. Elissef. An introduction to variable and feature selection. *Journal of Machine Learning Research*, 2003.

[9] K. Kira and L. Rendell. A practical approach to feature selection. *ICML*, 1992.

[10] R. Kohavi and G. John. Wrappers for feature selection. *Artificial Intelligence*, 97(1-2), 273–324, 1997.

[11] P. Langley and W. Iba. Average-case analysis of a nearest neighbor algorithm. In *Proceedings of the 13th Int. Conf. on Artificial Intelligence*, 1993.

[12] D.D. Lee and H.S. Seung. learning the parts of objects by non-negative matrix factorization. *Nature* 401(10), 1999

[13] D. D. Lewis. Feature selection and feature extraction for text categorization. In *Proceedings of Speech and Natural Language Workshop*, 1992.

[14] T.S. Motzkin and E.G. Straus. Maxima for graphs and a new proof of a theorem by turan. *Canadian Journal of Math.*, 17:533–540, 1965.

[15] A.Y. Ng, M.I. Jordan and Y. Weiss. On Spectral Clustering: Analysis and an algorithm. NIPS, 2001.

[16] B.A. Olshausen and D.J. Field. Emergence of simple-cell receptive field properties by learning a sparse code for natural images. *Nature*, 381(13), 1996.

[17] M. Pavan and M. Pelillo. A new graph-theoretic approach to clustering and segmentation. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2003.

[18] P. Perona and W. T. Freeman. A Factorization Approach to Grouping. ECCV,1998.

[19] S. Sarkar and K.L. Boyer Quantitative measures of change based on feature organization: eigenvalues and eigenvectors In *CVIU*, 71(1), pp. 110-136, 1998.

[20] J. Shi and J. Malik. Normalized Cuts and Image Segmentation. *PAMI* 22(8), 2000.

[21] Talagrand Concentration of measure and isoperimetric inequalities in product spaces. *Publ. Math. I.H.E.S. 81, 1995, 73-203.*

[22] V.N. Vapnik. *The nature of statistical learning*. Springer, 2nd edition, 1998.

[23] P. Viola and M. Jones. Robust Real-time Object Detection *IJCV*, 2002.

[24] Y. Weiss. Segmentation using eigenvectors: a unifying view. *ICCV*, 1999.

[25] Weston, J., S. Mukherjee, O. Chapelle, M. Pontil, T. Poggio and V. Vapnik. Feature Selection for SVMs. *NIPS*, 2001.

[26] L. Wolf and A. Shashua. Feature Selection for Unsupervised and Supervised Inference: the Emergence of Sparsity in a Weighted-based Approach. *Technical report 2003–58, School of Eng. and CS, June 2003.*