Informed Projections

David Cohn*
DigitalMC Corporation and
Carnegie Mellon University
Pittsburgh, PA 15213
cohn@digitalmc.com

Abstract

Low rank approximation techniques are widespread in pattern recognition research — they include LSA, PLSA, PCA, the Generative Aspect Model, and many forms of bibliometric analysis. All make use of a low-dimensional manifold onto which data are projected.

Such techniques are generally “unsupervised,” which allows them to model data in the absence of labels or categories. With many practical problems, however, some prior knowledge is available in the form of context. In this paper, I describe a principled approach to incorporating such information, and demonstrate its application to PCA-based approximations of several data sets.

1 Introduction

Many practical problems involve modeling large, high-dimensional data sets to uncover similarities or latent structure. Linear low rank approximation techniques such as PCA [11], LSA [3], PLSA [4] and generative aspect models [8] are powerful tools for approaching these tasks. They identify (relatively) low-dimensional hyperplanes that best approximate the data according to a given noise model. In doing so, they exploit and expose regularities in the data: the hyperplanes represent a latent space whose dimensions are often observed to correspond to distinct latent categories in the data set. For example, an LSA-derived low-rank approximation to a corpus of news stories may have dimensions corresponding to “politics,” “finance,” “sports,” etc. Documents with the same inferred sources (therefore “about” the same topic) generally lie close to each other in the latent space.

The broad applicability of these techniques comes from the fact that they are essentially “unsupervised” — a model is learned in the absence of training labels indicating class or category memberships. There are, however, many situations in which some prior information is available; in these cases, we would like to have some way of using that information to improve our model.

Nigam et al. [9] studied the problem of learning to classify data into pre-existing categories in the presence of labeled and unlabeled examples. Their approach augmented a traditional supervised learning algorithm with distribution information made available from the unlabeled data. In contrast, this paper considers a method for augmenting a traditional unsupervised learning problem with the addition of “semi-supervised” information.

* DRAFT: submitted to NIPS2002 — do not distribute.
1.1 Semi-supervised learning

Rather than assuming we have known classes and labels, we will assume only “semi-supervised” information — we will assume prior knowledge that some of our examples are “similar” to others, and should occupy similar places in the latent space. The similarity is indicated by set membership, where each observation in our data set may be a member of zero, one or more sets, each of which is self-similar.

Note that the sets are not required to be comprehensive — we may only have known associations between a handful of observations. Further, the sets are not required to be disjoint — we may know that members of a set are similar, but there is no implication that members of two different sets are dissimilar. As such, this information is strictly weaker than that provided by supervised data: given class-labeled data, we can define each class to be a set, but given set memberships, we do not have enough information to assign distinct classes.

Set membership is a natural concept for many real-world problems. We frequently have some reason for believing that a set of observations are similar in some sense without wanting to or being able to say why they are similar. The hope is that by indicating which observations are similar, we can bias our model focus on relevant features and to ignore differences that, while statistically significant, are not correlated with our idea of similarity in the problem at hand. This paper describes an algorithm validating the use of this approach.

1.2 Related work

The techniques most closely related to the one described in this paper are variations on linear discriminant analysis (LDA), such as that by Saon et al. [14, 7]. They begin with data that have been labeled according to known classes. LDA is an essentially supervised technique that seeks a projection minimizing the ratio of in-class variance to between-class variance, while discriminative projections attempt to maximize differences in both the means and variances of the given classes.

With LDA however, it is assumed that the classes are distinct and known in advance, and that all data belong to exactly one class. With an informed projection, in contrast, data may belong to zero, one or more sets. The model is biased to minimize the variance of members of the same set, but the bias is independent of any other sets.

For illustration, consider Platt’s playlist selection problem [10]. The tracks on a given album may be seen as a playlist — the artist who created the album believed that, in some sense, the tracks “belonged together.” As such, we would like our model to capture the similarity of tracks from the same album. But a given track may appear on multiple albums. By treating each album as a separate class, LDA and discriminative projection would be forced to find a projection that separates the two albums in spite of their common member. An informed projection, however, would minimize the variances independently, and by transitivity would match the intuition that albums with common tracks were in some sense similar.

1.3 Cross-language retrieval

Another closely-related technique is CI-LSI [5], which uses latent semantic analysis for cross-language retrieval. The technique involves training on text documents from a parallel corpus for two or more languages (e.g. French and English), such that each document exists as both an English and French version. In CI-LSI, each document is merged with its twin, and the hyperplane is fit to the set of paired documents.

The goal of CI-LSI matches the goal of this paper, and the technique can in fact be seen as a special case of the informed projections discussed here. By using the “mean” of a pair of
Figure 1: PCA maximizes the variance of the observations (on left), while an informed projection minimizes variance of projections from observations belonging to the same set.

documents as a proxy for the documents themselves, we assert that the two come from a common source; fitting a model to a collection of such means finds a maximum likelihood solution subject to the constraint that each member of a pair comes from a common source.

2 Informed and uninformed projections

To introduce informed projections, I will first briefly review principal components analysis (PCA) and an algorithm for efficiently computing the principal components of a data set.

2.1 PCA and EMPCA

Given a finite data set $X \subset R^n$, where each column of $X$ corresponds to one observation, PCA can be used to find a rank $m$ approximation $\hat{X}$ (where $m < n$) which minimizes the sum squared distortion with respect to $X$. It does this by identifying the $m$ orthogonal directions in which $X$ exhibits the greatest variance, corresponding to the $m$ largest eigenvectors $C = [C_1, C_2, \ldots, C_m]$. $X$ can then be projected onto the hyperplane defined by $C$ as

$$\hat{X} = C(C^T C)^{-1} C^T X.$$  

Although not strictly a generative model, PCA offers a probabilistic interpretation: $C$ represents a maximum likelihood model of the data under the assumption that $X$ consists of (Gaussian) noise-corrupted observations taken from linear combinations of $m$ sources in an $n$-dimensional space. The values for $\hat{X}$ then represent maximum likelihood estimates of the mixtures responsible for the corresponding values in $X$.

Roweis [12] described an efficient iterative technique for identifying $C$ using an EM procedure. Beginning with an arbitrary guess for $C$, the latent representation of $X$ is computed

$$Y = (C^T C)^{-1} C^T X$$  

after which $C$ is updated to maximize the estimated likelihoods

$$C = X Y^T (Y^T Y)^{-1}.$$  

Equations 2 and 3 are iterated until convergence (typically less than 10 iterations), at which time the sum squared error of $\hat{X}$’s approximation to $X$ will have been minimized.

2.2 Informed projections

PCA only penalizes according to squared distance of an observation $x_i$ from its projection $\hat{x}_i$. Given a Gaussian noise model, $\hat{x}_i$ is the maximum likelihood estimate of $x_i$’s “source,” which is the only constraint with which PCA is concerned.
If we believe that a set of observations \( S_i = \{x_1, x_2, \ldots, x_n\} \) have a common cause, then they should share a common source. For a hyperplane defined by eigenvectors \( C \), the maximum likelihood source is the mean of \( S_i \)'s projections onto \( C \), denoted \( \overline{S_i} \). As such, the likelihood should be penalized not only on the basis of the variance of observations around their projections \( \left( \sum_j ||x_j - \hat{x}_j||^2 \right) \), but also the variance of the projections around their set means \( \left( \sum_i \sum_{x_j \in S_i} ||\hat{x}_j - \overline{S_i}||^2 \right) \).

These two penalty terms may be at odds with each other, so we must introduce a hyperparameter \( \beta \) representing how much weight to place on accurately reproducing the original observations and how much to place on preserving the integrity of the known sets:

\[
E_\beta = (1 - \beta) \sum_j ||x_j - \hat{x}_j||^2 + \beta \sum_i \sum_{x_j \in S_i} ||\hat{x}_j - \overline{S_i}||^2.
\] (4)

Assigning equal weight (\( \beta = 0.5 \)) to the two penalties in Equation 4 is equivalent to minimizing \( \sum_i \sum_{x_j \in S_i} ||x_j - \overline{S_i}||^2 \) under the assumption that all otherwise unaffiliated \( x_i \) are members of their own singleton sets. This is simply the squared distance from each observation to its projected cluster mean, which appears to be the criterion CI-LSI minimizes by averaging documents.

Our experiments in the next section suggest that while \( \beta = 0.5 \) appears to be an acceptable weighting in most cases, it is not always optimal. For many applications, \( \beta \) could clearly be optimized by some form of cross-validation over known sets; a careful study of this process will be the subject of future work.

### 2.3 Finding an informed projection

The error criterion in 4 may be efficiently optimized with an expectation-maximization (EM) procedure based on Roweis’ EMPCA [12], alternately computing estimated sources \( \hat{x} \) and maximizing the likelihoods of the observed data given those sources.

The likelihood with respect to a set is maximized by minimizing the variance of projections from members of a set around their mean. This is at odds with the efforts of PCA to maximize likelihood by maximizing the variance of projections from the data set at large. We can make these forces work together by replacing the members of set \( S_i \) by a “complement set” \( \overline{S_i} \) such that the variance of \( S_i \)'s projections is minimized by maximizing the variance of \( \overline{S_i} \)'s projections.

The complement set may be determined analytically, but can also be computed efficiently as an extra step between the “E” and “M” steps of the EM iteration. Given an observation \( x_j \in S_i \), the complement for \( x_j \) may be computed in terms of its projection \( \hat{x}_j \) onto the hyperplane and \( \overline{S_i} \), the mean of the set.

In order to “pull” the current hyperplane in the direction that will minimize \( x_j \)'s distance from the set mean, \( \hat{x}_j \)'s projection must lie along line from \( \overline{S_i} \) to \( \hat{x}_j \) at a distance from the hyperplane equal to \( ||x_j - \hat{x}_j|| \). With some geometric manipulation (Figure 2), it can be seen that

\[
\hat{x}_j = \overline{S_i} + (\hat{x}_j - \overline{S_i}) \frac{||x_j - \hat{x}_j||}{||\hat{x}_j - \overline{S_i}||} + (\hat{x}_j - x) \frac{||\hat{x}_j - \overline{S_i}||}{||\hat{x}_j - \overline{S_i}||}.
\]

For efficiency, it is worth noting that by subtracting each set's mean from its constituent observations, all sets may be combined into a single zero-mean “superset” \( \overline{S} \) from which complements are computed.
Once the complement set has been computed, it can be appended to the original observations to create a joint data set, denoted \( X^+ = [X | \bar{S}_i] \), and the “M” step of the EM procedure is continued as before:

\[
Y = (C^T C)^{-1} C^T X^+, \quad C = X^+ Y^T (YY^T)^{-1}.
\] (5)

Applying \( \beta \) to the optimization is straightforward – if we preprocess the data by subtracting the mean of the observations (as is standard for PCA), the effect of each observation is to apply a “torque” to the current hyperplane around the origin. By multiplying all coordinates of an observation by the same scalar, we scale the torque applied by the same amount. As such, we can trade off the weight attached to enforcing the sets against the weight attached to reconstructing the original data by multiplying \( \bar{S}_i \) and \( X \) by \( \beta \) and \( 1 - \beta \) respectively:

\[
X_{ij}^+ = [(1 - \beta)X | \beta \cdot \bar{S}_i]
\]

### 3 Experiments

I examined the effect of “informing” projections on three data sets from two domains. The first two were text data sets taken from the WebKB project and the “20 newsgroups” data set. The third data set consisted of acoustic features from recorded music.

#### 3.1 WebKB data

The first set of experiments began with a vocabulary-restricted subset of the WebKB data set [2]. Using Rainbow [6], I tokenized 1000 randomly-selected documents, stripping out HTML and digits, and kept the 1000 terms with highest class-dependent information gain (the reduced vocabulary greatly decreased processing times). The result was 1000 documents with 1000 features, where feature \( f_{i,j} \) represented the frequency with which term \( j \) occurred in document \( x_i \). Sets were constructed from the categories provided with each document.

The experiments varied both the fraction of the training data for which set associations were provided (0-1) and the weight given to preserving those sets (also 0-1). For each combination, I ran 40 trials, each using a randomized split of 200 training documents and 100 test documents. Accuracy was evaluated based on leave-one-out nearest neighbor classification over the test set.\(^2\)

---

1. Since \( \bar{S}_i \) depends on the projections, and therefore the position of the hyperplane, it must be recomputed with each iteration.

2. Obviously, simple nearest neighbor is far from the most effective classification technique for this domain. But the point of the experiment is to evaluate to what degree informing a projection preserves or improves topic locality, which nearest neighbor classifiers are well-suited to measure.
Figure 3 summarizes the results of these experiments. As expected, the more documents that had set associations, the greater the improvement in classification accuracy, but this improvement was only evident for $0.3 \leq \beta \leq 0.7$; below 0.3, the sets were not given enough weight to make a difference, while above 0.7 there is a rapid deterioration in accuracy.

### 3.2 20 Newsgroups

The second set of experiments also used a standard text classification corpus, but with an unrestricted vocabulary. Beginning with the documents of the 20 newsgroups data set, I again preprocessed the documents as above with Rainbow, but this time kept the entire vocabulary (27214 unique terms), instead of preselecting maximally informative terms.

Because of the additional running time required to handle the complete vocabularies, the experiments used all set labels and only varied the weighting. Thirty independent training and test sets of 100 documents each were run for $0 \leq \beta \leq 1$, and as before, accuracy was evaluated in terms of leave-one-out classification error on the test set.
Figure 4 summarizes the results of these experiments. The characteristic learning curve is very similar to that for the WebKB data — an intermediate set weighting yields significantly better performance than the purely supervised or unsupervised cases. There is, however, one notable distinction: in these experiments, there is much less variation in accuracy for large values of $\beta$ — it almost appears that there are three stable regions of performance. Future work will examine this difference by taking the (now obvious) step of running unrestricted-vocabulary experiments on the WebKB data.

### 3.3 Album recognition from acoustic features

The final test used a proprietary data set of acoustic properties of recorded music. The data set contained 11252 recorded music tracks from 939 albums. Each observation consisted of 85 highly-processed acoustic features extracted automatically via digital signal processing.

The goal of this experiment was to determine whether informing a projected model could improve the accuracy with which it could identify tracks from the same album. Recalling Platt’s playlist selection problem [10], this can serve as a proxy for estimating how well the model can predict whether two tracks “belong together” by some subjective measure.

For these experiments, I selected the first 8439 tracks (3/4 of the data) for training, assigning each track to be a member of the set defined by the album it came from. Many tracks appeared on multiple albums (“Best of...” and soundtrack collections). The remaining 2813 tracks were used as test data.

The 85 dimensional features were projected down into a 10 dimensional space and a 40 dimensional space, informing the projection with sets defined by tracks from the same album. As above, I measured the frequency with which each test track had another track from the same album as its nearest neighbor when projected down into this same space.

<table>
<thead>
<tr>
<th>weight</th>
<th>$\beta = 0.0$</th>
<th>$\beta = 0.5$</th>
<th>$\beta = 1.0$</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>acc</td>
<td>ratio</td>
<td>acc</td>
</tr>
<tr>
<td>10 dim</td>
<td>0.1056</td>
<td>0.3822</td>
<td>0.1141</td>
</tr>
<tr>
<td>40 dim</td>
<td>0.2179</td>
<td>0.5736</td>
<td>0.2190</td>
</tr>
</tbody>
</table>

Table 1: Album recognition results using 2813 test tracks from 316 albums. For each weighting $\beta$, “acc” indicates accuracy - the fraction of times which the closest track to a test track came from the same album; “ratio” indicates the average ratio of intra-album distances to inter-album distances in the test set. In all cases, informing the projection with a weight of $\beta = 0.5$ increases the accuracy and decreases the ratio of the model.

Table 1 summarizes the results of these experiments. While the improvements in performance are not as striking as those from the previous experiments, they are nonetheless significant. One reason for the meager improvement may be that the features from which the projections were computed had already been manually optimized for classification accuracy. As time permits, I will be expanding these experiments to consider projections from raw audio feature sets.

### 4 Discussion and future work

The experiments so far indicate that adding set information to a low rank approximation does improve the quality of a model, but only to the extent that the information is used in conjunction with the unsupervised information already present in the data set. Too much reliance on set information (using this particular formalism) is in fact detrimental.

What weight is best remains an open question. Giving equal weight to sets and unsuper-
vised data ($\beta = 0.5$) seems to be robust, but not quite optimal; one subject of future work will be exploring the effect of $\beta$ and techniques for setting it automatically.

4.1 Future work

Beyond experiments that could clarify the effect of $\beta$ on model fitness, there are a number of obvious directions for future work. The first is further exploration on the relationship between informed PCA and and the variants of LDA discussed in Section 1.2. Although the differences are mathematically straightforward, the effect of enforcing set separations bears further study.

A second broad area for future work is the application of the techniques described here to richer low rank approximation models. While this paper considered the effect of informing PCA, it would be fruitful to examine both the process and effect of informing multinomial-based models [4, 1], fully-generative models [8] and local linear embeddings [13].

References


