Query Expansion and Dimensionality Reduction: Notions of Optimality in Rocchio Relevance Feedback and Latent Semantic Indexing

Miles Efron*
School of Information, University of Texas
Sanchez Building 564, 1 University Station D7000
Austin, TX 78712-0390
miles@ischool.utexas.edu

December 7, 2006

Abstract

Rocchio relevance feedback and latent semantic indexing (LSI) are well-known extensions of the vector space model for information retrieval (IR). This paper analyzes the statistical relationship between these extensions. The analysis focuses on each method's basis in least-squares optimization. Noting that LSI and Rocchio relevance feedback both alter the vector space model in a way that is in some sense least-squares optimal, we ask: what is the relationship between LSI's and Rocchio's notions of optimality? What does this relationship imply for IR? Using an analytical approach, we argue that Rocchio relevance feedback is optimal if we understand retrieval as a simplified classification problem. On the other hand, LSI's motivation comes to the fore if we understand it as a biased regression technique, where projection onto a low-dimensional orthogonal subspace of the documents reduces model variance.

1 Introduction

This paper examines the relationship between two extensions to Salton's vector space model (VSM) of information retrieval (Salton, Wong, & Yang, 1975), Rocchio's method for rele-

*The author wishes to thank Robert M. Losee and Jonathan Elsas, who provided helpful comments on an early draft of this paper. Additionally, the anonymous reviewers of this paper offered excellent advice during revision, for which I was very grateful.

vance feedback (Rocchio, 1971) and latent semantic indexing (LSI) (Deerwester, Dumais, Landauer, Furnas, & Harshman, 1990). Though both of these methods have been well studied individually, the literature lacks a theoretical analysis of their relationship. This paper speaks to that omission. Both techniques use least-squares optimization to define a projection of documents onto a low-dimensional subspace of the corpus. Rocchio's model seeks the direction in term space that is maximally discriminative in the least-squares sense, while LSI projects documents onto the dimensions that capture the maximal amount of the original variance. This paper explicates the mathematical relationship between these projections and treats the implications of this relationship for retrieval.

Due to the complexity of information retrieval (IR), we bring a simplified analytical approach to this study. We ask: subject to admittedly narrow assumptions, under which conditions can LSI's variance-based projection be said to perform better than Rocchio's discriminant-based projection, and vice versa? The relationship between these projection methods is complex. But a clear pattern emerges from the research presented here: the conditions under which one projection outperforms the other depends on how we understand the retrieval problem itself. Projecting documents onto dimensions that are maximally discriminative yields superior performance if we understand retrieval as a classification problem where our goal is to separate relevant and non-relevant documents. But if we imagine retrieval as a regression problem, where our goal is to estimate the degree of relevance of a document, discovering a low-dimensional orthogonal subspace is desirable.

While analytical models cannot match experimental data in richness, they compensate for this in their clarity (Losee, 1998). Working in highly simplified terms allows our discussion to progress with greater lucidity than it could if we relied on inference and observation. The following sections, then, do not aim to describe retrieval in detail. Rather, the argument developed here is intended to further our understanding of the idea of optimality as it pertains to the problem of data projection for IR in a theoretical sense.

2 Projection Methods in IR

Under the vector space model, documents and queries are represented as vectors in the vector space spanned by the indexing terms of their collection of documents. Each document's location in this term space is defined by its score (term count, weight, etc.) on each of the p terms that comprise the indexing vocabulary for the collection's n documents. Similarity between a query vector $\mathbf{q}$ and a document vector $\mathbf{d}$ is modeled by the cosine of the angle between these vectors:

$$s(\mathbf{q}, \mathbf{d}) = \frac{\mathbf{q}\dot{\mathbf{d}}}{\|\mathbf{q}\|\|\mathbf{d}\|} \tag{1}$$

If we assume that queries and documents have been normalized to unit length before analysis, then equation 1 is simply the inner product between $\mathbf{q}$ and $\mathbf{d}$. Representing our corpus as the $n \times p$ document-term matrix $\mathbf{X}$ with documents on the rows and terms on the columns, we have the basic vector space model of relevance for the query q represented by query vector $\mathbf{q}$:

$$\hat{r}(q, \mathbf{X}) = \mathbf{s} = \mathbf{X}\mathbf{q} \tag{2}$$

where $\mathbf{s}$ is a vector of query-document similarity scores. Under Salton's model, we estimate the relevance of each document to a query by projecting the document from the p-dimensional term space to a single dimension. The projection in the classic VSM is defined by the query vector $\mathbf{q}$.

Since its inception, the VSM has seen criticism. Documents and queries, a common complaint runs, are underdetermined in a way that the VSM can't account for (Wong, Ziarko, & Wong, 1985). Although a query may lack a particular term, that does not imply that the term is not associated with the information need that generated the query. Thus queries about *cars* might be satisfied by documents about *automobiles*, despite the VSM's failure to recognize these terms' latent similarity.

Numerous extensions to the VSM have been proposed to remedy this problem. While the details of the extensions vary, many share a common strategy: to improve retrieval these methods alter the projection used to compute similarity relationships. Two of the most widely studied VSM extensions are Rocchio relevance feedback and latent semantic indexing. This study focuses on these models. Though neither Rocchio relevance feedback nor LSI represent the current state of the art in IR, we focus on them for several reasons. Most practically, both relevance feedback and LSI have seen wide use in IR literature and practice; they are mainstays of retrieval (Dumais, 1992, 2004; Salton & Buckley, 1997; Buckley, Salton, Allan, & Singhal, 1994; Ruthven & Lalmas, 2003). Secondly, Rocchio's model and LSI both spawned further elaborations on Salton's original model. Thus by studying these models rigorously, we may gain insight into a wide body of research.

Relevance feedback involves query expansion. That is, Rocchio's model estimates an optimal query in efforts to organize the projected documents accurately. On the other hand LSI is essentially a document expansion method (Bast & Majumdar, 2005). LSI alters the VSM by deriving a low-dimensional, uncorrelated representation of the corpus. Projecting documents onto this subspace, it is argued, mitigates overfitting in the basic vector space model.

Crucial for this paper is the fact that both relevance feedback and LSI alter the VSM by defining a projection that is optimal in the least-squares sense. Rocchio's optimal query is a special case of the Fisher linear discriminant, while LSI derives much of its motivation and methodology from principal component analysis. Thus both methods are firmly rooted in statistical theory. The remainder of this paper clarifies the statistical foundations of each

method with an eye toward articulating their relative merits with respect to information retrieval. In particular the discussion will concern the question, in what sense is Rocchio's model optimal and how does this differ from LSI's optimality?

3 Least-Squares Projections for IR

This section reviews Rocchio relevance feedback and latent semantic indexing. The discussion concerns each model's basis in least-squares optimization. The method of least-squares is an optimization technique for estimating a function $y = f(x)$ (Plackett, 1972). Given a sample of data $x_1 \dots x_n$ and corresponding $y_1 \dots y_n$, the least-squares estimate $\hat{y} = \hat{f}(x)$ is the function that minimizes the so-called sum of squared error $\sum_{i=1}^{n} (\hat{y}_i - y)^2$.

Models based on least-squares optimization are among the most studied in the statistical literature. Understanding precisely how each approach to IR—relevance feedback and LSI—relates to the method of least-squares will lend us a strong theoretical basis for understanding their interrelationship.

3.1 Rocchio's Model of Relevance Feedback

In his work on relevance feedback (Rocchio, 1971), Rocchio posited a framework for improving queries. To guide his research, Rocchio proposed the notion of the "optimal" query. That is, given a corpus D and some query q, for which a subset D_r of the corpus is relevant and the remainder D_n is non-relevant, Rocchio defines the optimal query $\mathbf{q}_{opt}$:

$$\mathbf{q}_{opt} = \boldsymbol{\mu}_r - \boldsymbol{\mu}_n \tag{3}$$

where $\boldsymbol{\mu}_r$ is the mean vector of D_r and $\boldsymbol{\mu}_n$ is the mean vector of D_n.

Equation 3 is equivalent to the Fisher linear discriminant under the assumption that the covariance matrix of the terms is $\mathbf{I}_p$. Fisher's linear discriminant forms the basis of discriminant analysis, a well-known method of statistical classification. Informally, the Fisher linear discriminant is the projection that best separates two (normally distributed) classes. More formally, the Fisher linear discriminant is the direction in the observed space that maximizes the squared distance between the projections of the class means $\boldsymbol{\mu}_r$ and $\boldsymbol{\mu}_n$ onto itself while minimizing the variance within classes.

We now show two results. First we will show that the linear discriminant is in fact an optimal projection in the least-squares sense. Second, we demonstrate that Rocchio's optimal query is a special case of the linear discriminant.

To show the least-squares optimality of the linear discriminant, our discussion follows Anderson (Anderson, 2003, p. 218). Fisher's linear discriminant is the projection of $\mathbf{X}$ that maximizes the squared distance between class means, relative to their shared covariance matrix $\mathbf{\Sigma}$. That is, the linear discriminant is the vector $\mathbf{w}$ that maximizes

$$\frac{[\boldsymbol{\mu}_r^T\mathbf{w} - \boldsymbol{\mu}_n^T\mathbf{w}]^2}{\mathbf{w}^T\mathbf{\Sigma}\mathbf{w}} \tag{4}$$

Expanding the numerator we have

$$[\boldsymbol{\mu}_r^T\mathbf{w} - \boldsymbol{\mu}_n^T\mathbf{w}]^2 = \mathbf{w}^T[(\boldsymbol{\mu}_r - \boldsymbol{\mu}_n)(\boldsymbol{\mu}_r - \boldsymbol{\mu}_n)^T]\mathbf{w}.$$

To maximize the numerator, holding the denominator constant, let λ be a Lagrange multiplier. We thus need to maximize

$$\mathbf{w}^T[(\boldsymbol{\mu}_r - \boldsymbol{\mu}_n)(\boldsymbol{\mu}_r - \boldsymbol{\mu}_n)^T]\mathbf{w} - \lambda(\mathbf{w}^T\mathbf{\Sigma}\mathbf{w} - 1). \tag{5}$$

Setting the derivatives with respect to $\mathbf{w}$ equal to zero, we obtain

$$2[(\boldsymbol{\mu}_r - \boldsymbol{\mu}_n)(\boldsymbol{\mu}_r - \boldsymbol{\mu}_n)^T]\mathbf{w} = 2\lambda\mathbf{\Sigma}\mathbf{w} \tag{6}$$

where $(\boldsymbol{\mu}_r - \boldsymbol{\mu}_n)\mathbf{w}$ is some scalar, say s. We then have

$$\boldsymbol{\mu}_r - \boldsymbol{\mu}_n = \frac{\lambda}{s}\mathbf{\Sigma}\mathbf{w}. \tag{7}$$

Thus $\frac{\lambda}{s}$ is an arbitrary constant. Setting $s = \lambda$ and rearranging terms, we have

$$\mathbf{w} = \mathbf{\Sigma}^{-1}(\boldsymbol{\mu}_r - \boldsymbol{\mu}_r) \tag{8}$$

which is the Fisher linear discriminant. From here it is easy to see that the Rocchio optimal query given by Equation 3 is equal to the linear discriminant under the condition of identity covariance.

Under Rocchio's formulation, then, retrieval is optimized when we replace $\mathbf{q}$ with $\mathbf{q}_{opt}$ to arrive at:

$$\hat{r}_{Rocchio}(q, \mathbf{X}) = \mathbf{s}_{Rocchio} = \mathbf{X}\mathbf{q}_{opt}. \tag{9}$$

This amounts to a projection of each document onto the vector that maximally separates (in the least-squares sense) the means of relevant and non-relevant documents. If we assume that documents are generated by multivariate Gaussian distributions (one for relevant and one for non-relevant documents), Equation 9 projects the documents along the vector that minimizes the probability of misclassifying a relevant document as non-relevant or vice versa (Duda, Hart, & Stork, 2001). The Rocchio optimal query, then, is optimal insofar as we understand retrieval as a two-class classification problem.

3.2 Latent Semantic Indexing

Whereas Rocchio relevance feedback is concerned with a projection that is optimal with respect to mean vectors, LSI derives a projection that is optimal with respect to variance. Several recent studies have argued that LSI aims to improve retrieval by representing the covariance structure of the indexing terms (Bast & Majumdar, 2005; Kontostathis & Pottenger, 2006; Efron, 2005). The current study pursues this interpretation, arguing that we can understand LSI as an effort to reduce the impact of overfitting term covariance models in the vector space model.

The latent semantic indexing model derives a projection matrix for documents by the singular value decomposition (SVD) of the document-term matrix $\mathbf{X}$:

$$\mathbf{X} = \mathbf{U}\mathbf{\Sigma}\mathbf{V}^T \tag{10}$$

where $\mathbf{U}$ and $\mathbf{V}$ are orthonormal matrices containing the eigenvectors of $\mathbf{X}\mathbf{X}^T$ and $\mathbf{X}^T\mathbf{X}$ respectively, while $\mathbf{\Sigma}$ is diagonal, with σ_{ii} the positive square root of the i^{th} eigenvalue. Further properties and proofs regarding SVD are given in (Golub & Van Loan, 1996).

Assuming that those eigenvectors corresponding to small eigenvalues constitute random error, LSI operates by defining the projection matrix $\mathbf{P}_{LSI}$:

$$\mathbf{P}_{LSI} = \mathbf{V}_k\mathbf{\Sigma}_k^{-1} \tag{11}$$

where $\mathbf{V}_k$ contains the first k columns of $\mathbf{V}$ and the diagonal $\mathbf{\Sigma}_k^{-1}$ contains the inverses of the largest k singular values. Prior to retrieval, a reduced rank approximation of the original matrix is obtained by

$$\mathbf{X}_p = \mathbf{X}\mathbf{P}_{LSI} \tag{12}$$

and each query is projected into the reduced space by

$$\mathbf{q}_p^T = \mathbf{q}^T\mathbf{P}_{LSI} \tag{13}$$

and query-document similarity is simply the inner product between the projected documents and queries:

$$\hat{r}_{LSI}(q, \mathbf{X}) = \mathbf{s}_{LSI} = \mathbf{X}_p\mathbf{q}_p. \tag{14}$$

The appeal of LSI lies in the notion that our observed data (term occurrences in documents) are subject to sampling error. We would like to gauge inter-document similarity not in this overfitted space, but in a space that more accurately reflects the covariance structure of the probability distribution that generated the documents. To accomplish this, LSI projects queries and documents onto the first k eigenvectors of the data. In this sense, LSI is closely related to principal component analysis (PCA). Showing the relationship between LSI and PCA will frame our discussion of LSI's basis in least-squares optimization.

3.2.1 Principal Components and Least-Squares

Principal component analysis can be described as a solution to several optimization problems. In the context of this paper, we define PCA as a least-squares problem. Our discussion follows the derivation in (Duda et al., 2001, Sec. 3.8.1). Given our $n \times p$ matrix $\mathbf{X}$ of rank p, we wish to find a k-dimensional approximation $\mathbf{X}_k$ that minimizes

$$\Delta = ||\mathbf{X} - \mathbf{X}_k||^2 \tag{15}$$

which is simply the squared error $\sum(x - x_k)^2$. We begin with the simplest case, where $k = 1$. That is, we wish to find a single p-vector $\mathbf{x}_0$ that minimizes the sum of squared distance between itself and the n row vectors of $\mathbf{X}$. Intuitively (refer to (Duda et al., 2001) for the proof) we suspect that $\bar{\mathbf{x}}$ the mean vector of $\mathbf{X}$ gives the best zero-dimensional representation of the data. To advance to one dimension we write $\mathbf{x}_i$ as line through the mean

$$\mathbf{x}_i = \bar{\mathbf{x}} + a_i\mathbf{e} \tag{16}$$

where a_i gives the distance of the point from the mean. We find an optimal set of coefficients a_i by minimizing

$$\begin{aligned}
\Delta_1(a_1, \ldots, a_n, \mathbf{e}) &= \sum_{i=1}^{n} ||(\bar{x} + a_i\mathbf{e}) - \mathbf{x}_i||^2 \\
&= \sum_{i=1}^{n} ||a_i\mathbf{e} - (\mathbf{x}_i - \bar{x})||^2 \\
&= \sum_{i=1}^{n} a_i^2||\mathbf{e}||^2 - 2\sum_{i=1}^{n} a_i\mathbf{e}^T(\mathbf{x} - \bar{\mathbf{x}}) + \sum_{i=1}^{n} ||\mathbf{x}_i - \bar{\mathbf{x}}||^2. \qquad (17)
\end{aligned}$$

Setting $||\mathbf{e}|| = 1$, we partially differentiate with respect to a_i and set the derivative to zero to obtain

$$a_i = \mathbf{e}^T(\mathbf{x}_i - \bar{\mathbf{x}}). \qquad (18)$$

Expressing a_i in these terms allows us to re-write Equation 17 as a function of $\mathbf{e}$ to find the optimal direction for our projection:

$$\begin{aligned}
\Delta_1(\mathbf{e}) &= = \sum_{i=1}^{n} a_i^2 - 2\sum_{i=1}^{n} a_i^2 + \sum_{i=1}^{n} ||\mathbf{x}_i - \bar{\mathbf{x}}||^2 \\
&= -\sum_{i=1}^{n} [\mathbf{e}^T(\mathbf{x}_i - \bar{\mathbf{x}})]^2 + \sum_{i=1}^{n} ||\mathbf{x}_i - \bar{\mathbf{x}}||^2 \\
&= -\sum_{i=1}^{n} \mathbf{e}^T(\mathbf{x}_i - \bar{\mathbf{x}})(\mathbf{x}_i - \bar{\mathbf{x}})^T\mathbf{e} + \sum_{i=1}^{n} ||\mathbf{x}_i - \bar{\mathbf{x}}^2|| \\
&= -\mathbf{e}^T\mathbf{S}\mathbf{e} + \sum_{i=1}^{n} ||\mathbf{x}_i - \bar{\mathbf{x}}^2|| \qquad (19)
\end{aligned}$$

where $\mathbf{S}$ is proportional to the covariance matrix of $\mathbf{X}$. Maximizing $\mathbf{e}^T\mathbf{S}\mathbf{e}$ will of course minimize Equation 19, as we desire. Letting λ be a Lagrange multiplier subject to $||\mathbf{e}|| = 1$ we differentiate

$$\mathbf{e}^T\mathbf{S}\mathbf{e} - \lambda(\mathbf{e}^T\mathbf{e} - 1)$$

with respect to $\mathbf{e}$ and set the derivative to zero, obtaining

$$\mathbf{Se} = \lambda\mathbf{e}$$

to see that $\mathbf{e}$ is an eigenvector of $\mathbf{S}$ corresponding to the eigenvalue λ. To maximize $\mathbf{e}^T\mathbf{Se}$ we choose the eigenvector corresponding to the largest eigenvalue. In other words, the best 1-dimensional projection is a line passing through the mean of $\mathbf{X}$ in the direction of the principal eigenvector. This projection, $\mathbf{Xe}$, is the first principal component of $\mathbf{X}$.

We can easily extend this result from one dimension to k dimensions. As we would expect, the best k-dimensional representation in the least-squares sense is the first k principal components, which we obtain by projecting $\mathbf{X}$ onto the first k eigenvectors of its covariance matrix.

Recall that LSI is based on the singular value decomposition of $\mathbf{X}$. In particular, in Equation 11 we defined LSI's projection matrix in terms of the first k eigenvectors of $\mathbf{X}^T\mathbf{X}$. If the columns of $\mathbf{X}$ have been centered around their means, then LSI's projection gives the first k principal components; LSI and PCA are equivalent in this case. However, in IR column centering is not the norm, so the LSI and PCA projections will differ slightly. The more general fact that the singular value decomposition allows us to find the best rank-k approximation of a rank-p matrix where $k < p$ is known as the Eckart Young theorem and is proved in (Eckart & Young, 1936).

Two crucial results emerge from the definition of the singular value decomposition (Equation 10) and our discussion so far. The singular vectors of $\mathbf{X}$ provide an orthonormal basis for the row- and column-space of $\mathbf{X}$. In other words, they are uncorrelated. Second, the LSI projection matrix $\mathbf{P}_{LSI} = \mathbf{V}_k\mathbf{\Sigma}_k^{-1}$ is the best rank-k approximation of $\mathbf{X}^T\mathbf{X}$, the term-term co-occurence matrix.

From a practical standpoint, LSI provides our model with an approximation of the term relationships. If it works, our hope is that this approximation might be better than the original model, having removed from its consideration spurious relationships. The assumption in LSI is that the least-squares optimal projection gives the best approximation in this context.

4 Comparing Projection Methods

We have shown that Rocchio relevance feedback and LSI alter the standard vector space model in a way that is in some sense least-squares optimal. Rocchio's method finds the direction in term space that maximally separates the mean vectors of relevant and non-relevant documents. On the other hand, LSI finds the k-dimensional orthogonal subspace of the documents that minimizes the distance between itself and the full-rank document-term

matrix. In this section we pursue the question: what are the implications of these different notions of optimality with respect to retrieval?

To begin our comparison, assume that we have two multivariate normal distributions: $\mathcal{D}_r$ and $\mathcal{D}_n$, which generate relevant and non-relevant documents, respectively. For the sake of simplicity, we assume that each of these distributions is bivariate normal with known mean vector and covariance matrix. Thus the density for relevant documents is $p(\mathbf{d}_r) \propto N(\boldsymbol{\mu}_r, \boldsymbol{\Sigma}_r)$ and the density for non-relevant documents is $p(\mathbf{d}_n) \propto N(\boldsymbol{\mu}_n, \boldsymbol{\Sigma}_n)$. For further simplification (but without loss of generality) we assume that the prior probabilities of observing relevant or non-relevant documents are equal.

Consider Figure 1, which shows 95% confidence regions for different arrangements of two bivariate normal distributions; $\mathcal{D}_r$ appears in black, while $\mathcal{D}_n$ is gray. In all four cases shown in Figure 1 we assume that the non-relevant documents are centered around the origin with $\boldsymbol{\Sigma} = \mathbf{I}_2$. In the upper left panel, the relevant documents also have identity covariance, while their mean vector is $\boldsymbol{\mu}^T = \begin{pmatrix} 2 & 0 \end{pmatrix}$. The upper right panel moves the relevant documents' density closer to the non-relevant documents. Finally, the lower panels show the effect of altering the covariance structure of the relevant documents' distribution. In the lower left and right panels, the covariance between features for relevant documents is 0.5 and 0.9, respectively.

Altering the orientation of the distributions for relevant and non-relevant documents affects the relative ease or difficulty of retrieval. We may assume that if the distributions are well separated, relevant and non-relevant documents will be easy to recognize. Conversely, if the distributions are similar, discrimination between relevant and non-relevant documents will be difficult. In the discussion that follows we hold constant the distribution of non-relevant documents, assuming that they are are centered around the origin with identity covariance. Further, we assume that for any orientation of relevant and non-relevant distributions, we may define a pooled covariance matrix $\boldsymbol{\Sigma}$, whose elements are simply the average between the covariance matrices of relevant and non-relevant documents. We examine the effect on each projection method introduced above as the orientation of relevant documents shifts.

4.1 IR as Classification

The upper left panel of Figure 2 shows 95% confidence regions where relevant documents have $\boldsymbol{\mu}_r^T = \begin{pmatrix} 2 & 0 \end{pmatrix}$ and the term covariance is 0.9. In the upper right panel, two vectors are superimposed on the confidence regions. The solid line corresponds to the dominant eigenvector of the pooled covariance matrix—the first LSI dimension. In contrast the dotted line points in the direction of the Rocchio optimal query.

To derive the projections shown in Figure 2, we begin by calculating the Rocchio optimal query for the data:

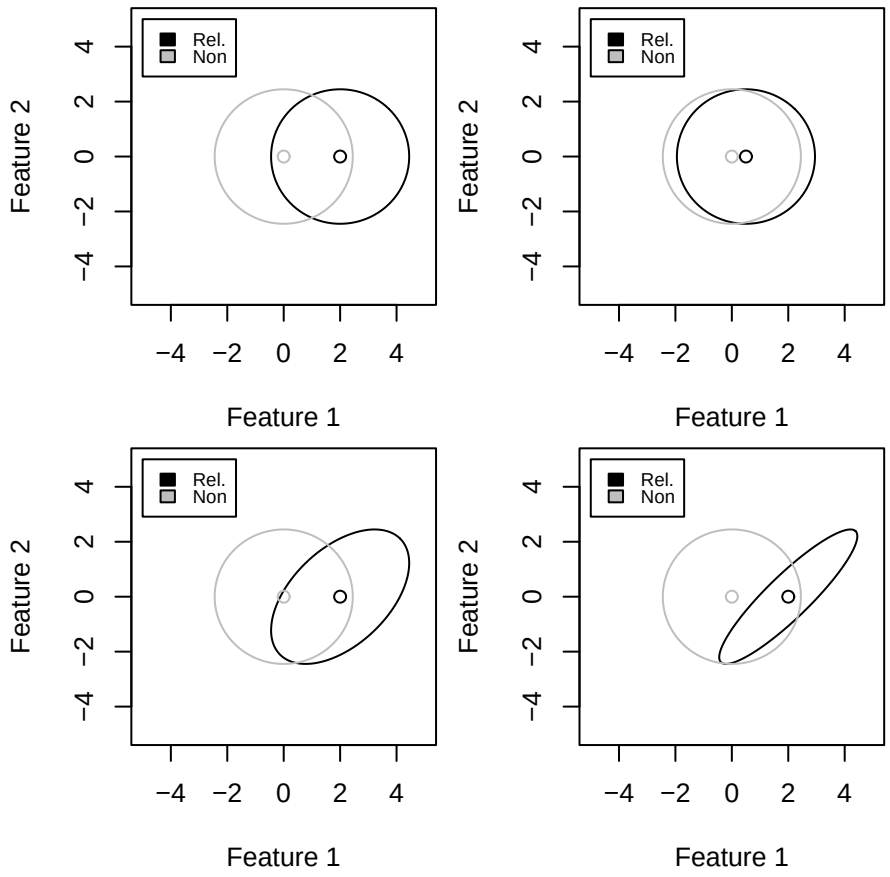

Figure 1: Sample Arrangements of Two Bivariate Normal Densities

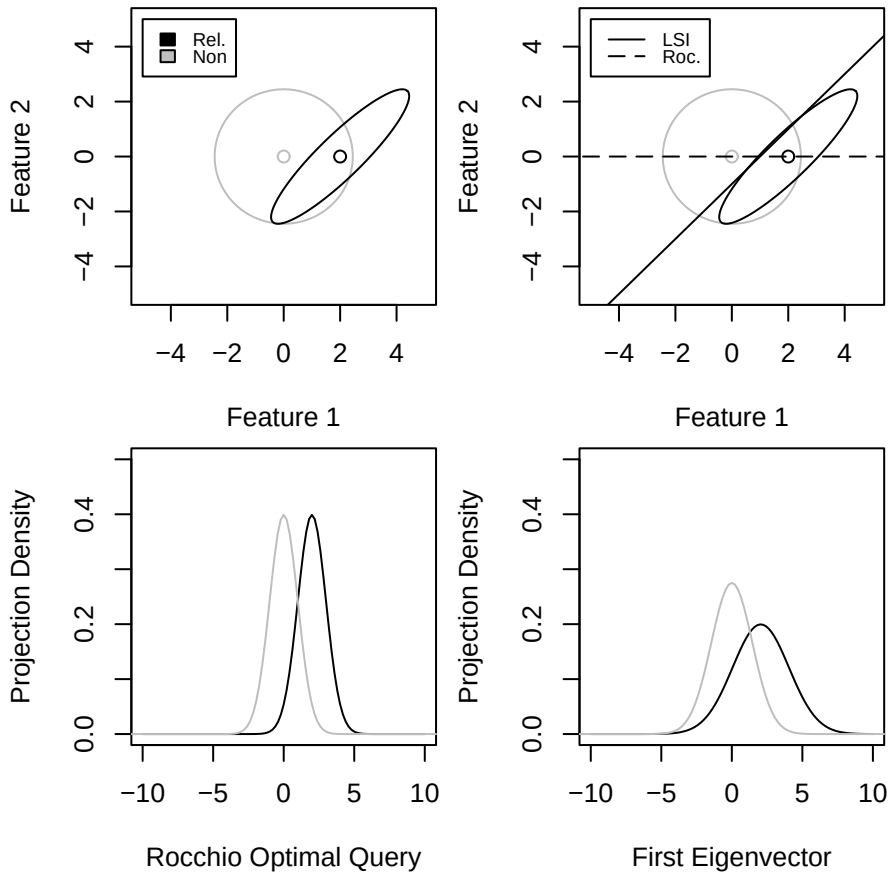

Figure 2: Projections of Two Normal Distributions

$$\begin{aligned}\mathbf{q}_{opt}^T &= \boldsymbol{\mu}_r^T - \boldsymbol{\mu}_n^T \\ &= (\ 2 \quad 0\ ) - (\ 0 \quad 0\ ) \\ &= (\ 2 \quad 0\ )\end{aligned}$$

Thus the Rocchio optimal query is the vector that passes through the mean vectors of each population. Normalizing this to unit length we have $\mathbf{u}_{opt}^T = (\ 1 \quad 0\ )$, which defines the projection of each mean vector:

$$m_i = \boldsymbol{\mu}_i^T\ \mathbf{u}_{opt} \tag{20}$$

for $i \in (\boldsymbol{\mu}_{rel},\ \boldsymbol{\mu}_{non})$. Because a linear function taken on a normal distribution is also normal, the projections are distributed $N(m_i,\ s_i)$ where m_i is defined in equation 20 and s_i, the standard deviation, is the positive square root of

$$s_i^2 = \mathbf{u}_{opt}^T\ \boldsymbol{\Sigma}_i\ \mathbf{u}_{opt}. \tag{21}$$

The bivariate normal distributions shown in Figure 2's upper left panel project onto the Rocchio optimal query as in the lower left panel of the figure. Thus the non-relevant documents are assumed to be normal along the linear discriminant with $m_{non} = \boldsymbol{\mu}_{non}^T\ \mathbf{u}_{opt} = 0$ and (by Equation 21) $s^2 = 1$. Likewise, the relevant documents are normally distributed with $m_{non} = \boldsymbol{\mu}_{rel}^T\ \mathbf{u}_{opt} = 4$ and $s^2 = 1$.

On the other hand, the projection defined by LSI is based on the pooled covariance matrix of the documents:

$$\boldsymbol{\Sigma} = \begin{pmatrix} 1 & 0.45 \\ 0.45 & 1 \end{pmatrix}$$

To find the direction in term space that captures maximal variance among the relevant documents (as is the definition of LSI) we compute the eigenvalue-eigenvector decomposition of $\boldsymbol{\Sigma}$, the covariance matrix for the relevant documents. We find the eigenvalues and eigenvectors of $\boldsymbol{\Sigma}$ by:

$$\begin{aligned}\boldsymbol{\Sigma}\mathbf{x} &= \lambda\,\mathbf{x} \\ \boldsymbol{\Sigma}\mathbf{x} - \lambda\,\mathbf{x} &= 0\end{aligned}$$

$$
\begin{aligned}
(\boldsymbol{\Sigma} - \lambda\, \mathbf{I}_2)\, \mathbf{x} &= \mathbf{0} \\
\begin{pmatrix} 1 & 0.45 \\ 0.45 & 1 \end{pmatrix} - \begin{pmatrix} \lambda & 0 \\ 0 & \lambda \end{pmatrix} &= \mathbf{0} \\
\begin{pmatrix} 1-\lambda & 0.45 \\ 0.45 & 1-\lambda \end{pmatrix} &= \mathbf{0}
\end{aligned}
$$

We find the characteristic polynomial of $\boldsymbol{\Sigma}_r$ by:

$$|\boldsymbol{\Sigma} - \lambda \mathbf{I}_2| = (1-\lambda)(1-\lambda) - 0.2025.$$

Solving the characteristic equation gives the principal eigenvalue $\lambda_1 = 1.45$, corresponding to the eigenvector $\mathbf{v}_1^T = (\ 0.707 \quad 0.707\ )$.

Recalling the projection matrix used by LSI (Equation 11), we find the mean vector for relevant documents in the space spanned by the first eigenvector:

$$m_r = \boldsymbol{\mu}_r \mathbf{P}_{LSI} \tag{22}$$

with an analogous projection for the non-relevant mean vector. As above we find the variance of each distribution (relevant and non-relevant) by

$$s_i^2 = \mathbf{P}_{LSI}^T\, \boldsymbol{\Sigma}_i\, \mathbf{P}_{LSI}. \tag{23}$$

This gives variance 0.904 for the relevant documents in the 1-space of their principal eigenvector. For non-relevant documents, we have variance of 0.476.

Having shown how we arrived that the distributions it depicts, let us return to our consideration of Figure 2. How can we characterize the projections entailed by Rocchio's model and local LSI's?

Because the two states of nature (relevance versus non-relevance) are for our purposes mutually exclusive, the probability of error is the area of overlap between the two normal distributions along a given dimension. That is, the probability of error is the probability of predicting *relevant* when a document is non-relevant plus the probability of predicting *non-relevant* when a document is relevant. More formally we have

$$P(err) = \int_{rel} p(x|non)P(non)dx + \int_{non} p(x|rel)P(rel)dx. \tag{24}$$

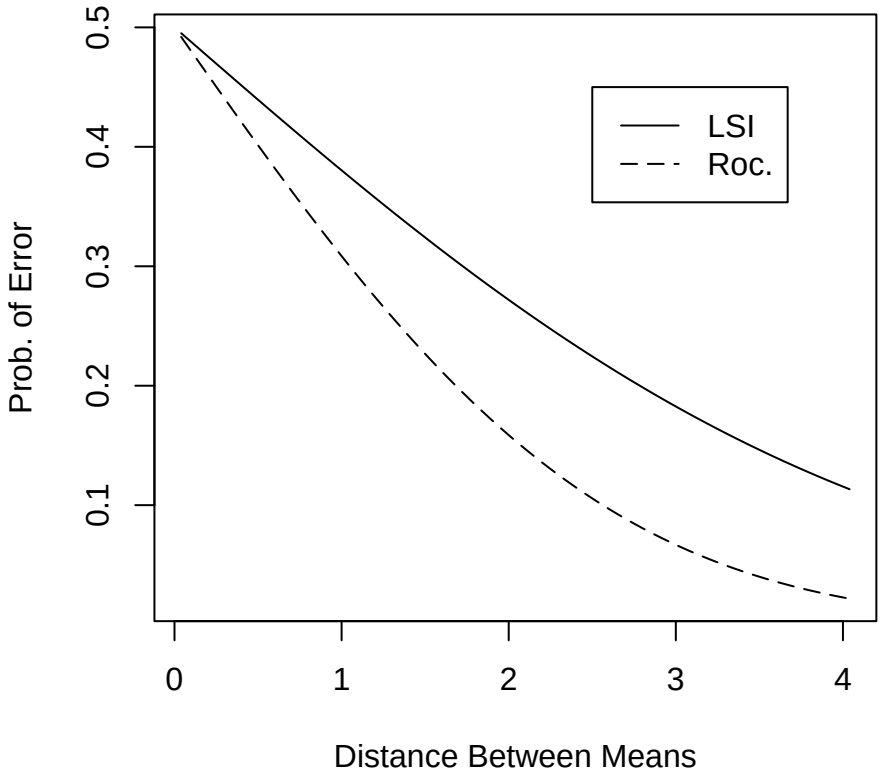

Figure 3: Relative Error of two Projection Methods

Given this model, what is the effect of altering the orientation of the document distributions in term space on the probability of error under our two projections? Figure 3 shows the relationship between the probability of error when we project onto the Rocchio optimal query and the probability of error when we project onto the principal eigenvector of the documents. The x-axis shows the distance between the mean vectors of relevant and non-relevant documents. Thus we begin (at the left side of the plot) with both populations centered at the origin. Moving to the right on the x-axis shows how far Feature 1 deviates from the origin for relevant documents. Thus at the extreme right of the plot, non-relevant documents are centered at the origin, while relevant documents are centered at $\boldsymbol{\mu}_r = (\ 4 \quad 0\ )$. The y-axis gives the probability of error under the Rocchio projection and the probability of error under LSI's projection.

When the mean vectors of both document populations are equal, neither projection is successful at separating relevant and non-relevant documents. Thus the left side of Figure 3 shows the probability of error under each projection to be equal as $\boldsymbol{\mu}_r$ approaches $\boldsymbol{\mu}_n$. With the mean vectors almost indistinguishable, both projections yield probability of error near

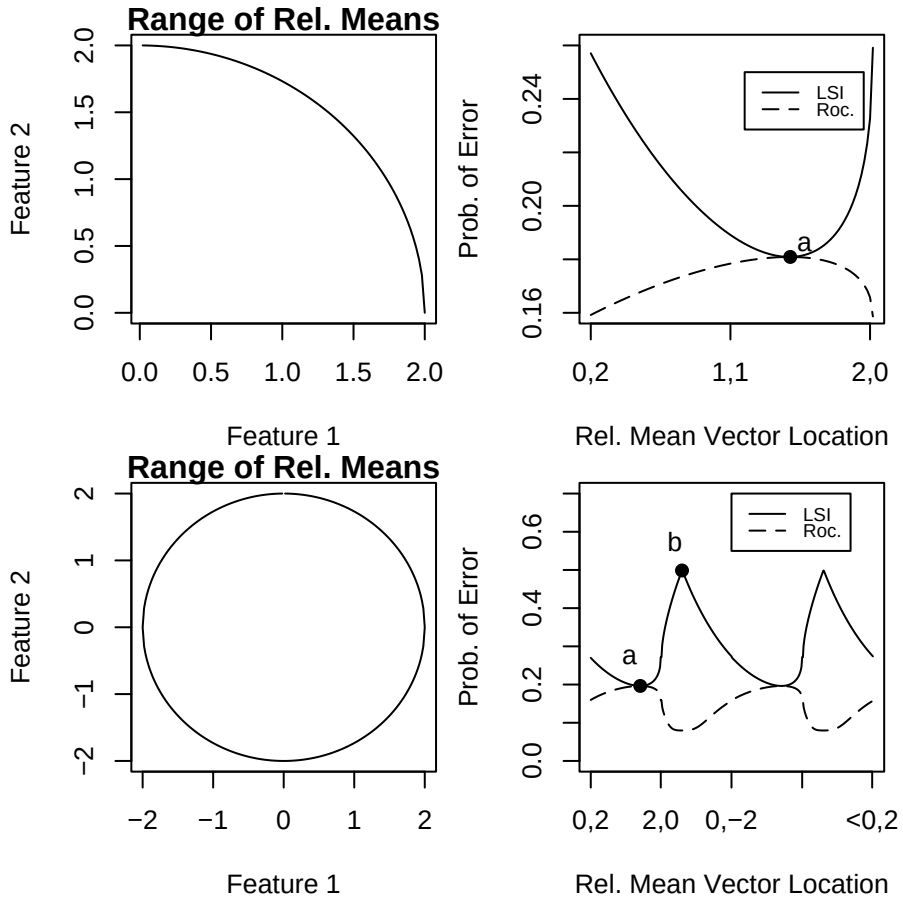

Figure 4: Error of Projections Under Different Document Orientations

0.5 (equivalent to a random guess).

However, as the mean vector of relevant documents tends towards a higher score on Feature 1, discriminating between relevant and non-relevant documents becomes more successful. As the means become more distant along Feature 1, both projection methods' probability of error decreases. However, the probability of error for the projection along the Rocchio optimal query decreases faster than it does if we project along the principal eigenvector.

A more complex picture arises if we alter the mean of relevant documents with respect to both features simultaneously. The right panels of Figure 4 show the probability of misclassification for each projection on the y-axes. The left panels show the locations of the relevant mean vector that are measured in the right panels. In the top panels we move the relevant mean vector from (0, 2) to (2, 0) along the circle with radius 2. In the bottom panels, we make an entire orbit of the circle, beginning at (0, 2) and proceeding clockwise 360 degrees. Thus the top panels deal only with mean vectors that have positive coordinates (as is most common in IR), while the bottom panels show the entire range of relationships between mean vector

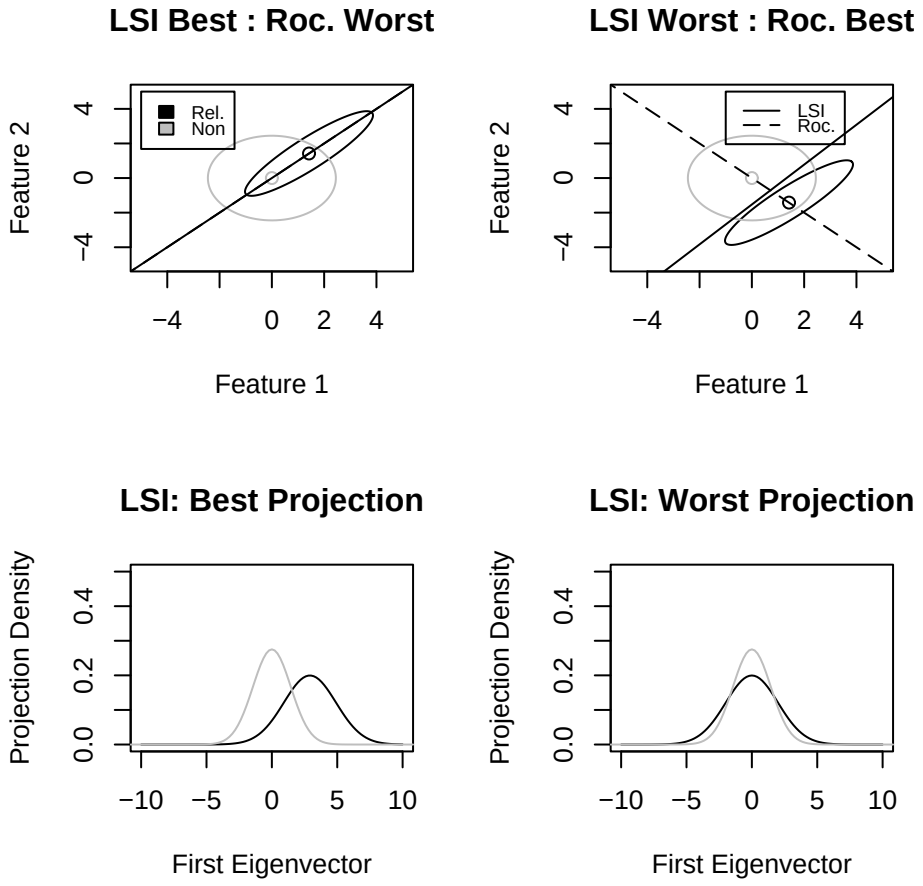

Figure 5: Best and Worst Projections

orientation and projection error in the context of this model.

A number of interesting results are evident from Figure 4. First, as we would expect, due to its relation to linear discriminant analysis, Rocchio's optimal query never gives higher risk of error than the LSI projection. In fact Rocchio's worst performance occurs with the document orientation that gives LSI's best performance (point a on the plot). On the other hand, LSI is at its worst when the documents minimize Rocchio's error (point b).

We can understand the inverse relationship between LSI's accuracy and Rocchio's by consulting Figure 5. The upper panels show the orientation of relevant and non-relevant populations in the cases that correspond to points a and b on Figure 4. The figure's lower panels show LSI's projections under each orientation. LSI's optimal document arrangement, we see, is the one in which the Rocchio optimal query and the principal eigenvector point in the same direction. On the other hand, LSI does worst when the dominant eigenvector and the Rocchio optimal query are orthogonal. Under this arrangement, we can see, both distributions–relevant and non-relevant–are collapsed onto each other when we perform the

LSI projection. Meanwhile, under this orientation, the Rocchio optimal query is perpendicular to the main axis of the relevant documents, leading them to have very low variance on its projection (and concomitantly high accuracy).

It is worth noting parenthetically that the minima of Rocchio's error probability in Figure 4 correspond to those arrangements in which the optimal query is identical to the true Fisher discriminant, as opposed to the simplified one used in the common Rocchio method.

4.2 IR as Regression

If the Rocchio optimal query always provides a lower risk of misclassification, what use is the projection offered by LSI? To understand why LSI's projection is useful, it helps to consider retrieval not as a classification problem, but rather as a regression problem. That is we can imagine our goal as estimating the degree of relevance of a particular document. Roger Story interprets IR as a regression in (Story, 1996) to demonstrate a Bayesian interpretation of LSI. If we think of retrieval as a regression problem, LSI's projection proves valuable insofar as it reduces the mean squared error (MSE) of the estimated regression model.

The standard linear regression model describes a real-valued response variable Y as a linear function of p predictor variables X_j for $j = 1 \ldots p$:

$$\mathbf{Y} = \mathbf{X}\boldsymbol{\beta} + \boldsymbol{\epsilon} \tag{25}$$

where $E\{\boldsymbol{\epsilon}\} = \mathbf{0}$ and $cov(\boldsymbol{\epsilon}) = \sigma^2\mathbf{I}$. Further, $\mathbf{E}\{\mathbf{Y}\} = \mathbf{X}\boldsymbol{\beta}$. The p-dimensional vector $\boldsymbol{\beta}$ describes the contribution that each predictor variable x_j makes to the response Y[1].

In regression analysis, the parameters comprising $\boldsymbol{\beta}$ are usually unknown. The task is to derive an estimator $\hat{\boldsymbol{\beta}}$ for these unknown parameters. The standard approach uses the method of least-squares to find the estimator $\hat{\boldsymbol{\beta}}$ that minimizes the squared error of the model:

$$SSE = \sum_{i=1}^{n} (y_i - \hat{y}_i)^2 \tag{26}$$

where $\hat{y}_i = \mathbf{x}_i^T\hat{\boldsymbol{\beta}}$. It is easily shown that we minimize Equation 26 by finding

$$\hat{\boldsymbol{\beta}} = (\mathbf{X}^T\mathbf{X})^{-1}\mathbf{X}^T\mathbf{Y}. \tag{27}$$

[1]If the predictors have not been centered around their mean then $\boldsymbol{\beta}$ has $p + 1$ dimensions, an extra parameter for the y-intercept

It can also be shown (cf. (Neter, Kutner, Nachtsheim, & Wasserman, 1996)) that this estimate yields the unbiased model with lowest variance among all linear, unbiased models. Thus the fitted regression model takes the form:

$$\hat{\mathbf{Y}} = \mathbf{X}\hat{\boldsymbol{\beta}}. \tag{28}$$

Vector space IR can thus be thought of as a regression problem where the response variable Y is the degree of relevance. Under this formulation, we may consider the vector $\boldsymbol{\beta}$ as fulfilling the same function as Rocchio's optimal query. Altering the notation of Equation 2 we have

$$\hat{r}(q, \mathbf{X}) = \mathbf{X}\hat{\boldsymbol{\beta}}.$$

Thus the parameter $\boldsymbol{\beta}$ is unknown and query-specific.

LSI's contribution to retrieval–projecting documents onto k maximally expressive dimensions–can be understood as an effort to improve the VSM's regression model. As argued in (Efron, 2005), LSI reduces errors in the VSM by accounting for inter-term correlations. In the context of this paper, LSI provides an optimal projection in the sense that the features of the projected documents are uncorrelated. The lack of correlation in the derived feature space improves the VSM similarity function by reducing the problem of colinearity in the predictors.

4.3 Colinearity in Linear Regression

Though independence among the predictor variables x_j is not requisite in the standard regression model, problems in estimation arise if the predictors show marked covariance. The general term for this situation is colinearity. A regression model fitted from collinear data suffers from high variance in the estimated $\hat{\boldsymbol{\beta}}$.

Inflated variance in the estimated regression function is problematic insofar as it suggests that the model has overfitted the training data. That is, a highly variable $\hat{\boldsymbol{\beta}}$ may perform well on the training data, yielding a low sum of squared error. But faced with new data, the same model is likely to perform poorly.

A clear way to see the relation between estimator variance and overfitting arises in the context of model selection by analysis of the mean squared error (MSE). Given a parameter θ and an estimator for it $\hat{\theta}$ we have MSE:

$$MSE(\hat{\theta}) = E((\hat{\theta} - \theta)^2). \tag{29}$$

In other words, the mean squared error of a regression model is simply the expected squared difference between y_i and $\hat{y}_i$.

It can be shown (cf. (Hastie, Tibshirani, & Friedman, 2001, P. 196)) that a model's MSE can be broken into two parts: the bias and the variance of the estimator as in Equation 30.

$$MSE(\hat{\theta}) = Bias^2(\hat{\theta}) + Variance(\hat{\theta}) \tag{30}$$

To shrink the MSE (i.e. to lower our expected rate of error), it is important to control both the bias and the variance of a model, a dynamic known as the bias-variance tradeoff. For example, a model that always gives a constant result regardless of its input will have zero variance, but is apt to show a high average error due to its heavy bias. On the other hand, a model that fits its training data perfectly has very low bias, but very high variance: given a slightly different training set, it's fitted $\hat{\theta}$ is likely to be different.

Colinear predictors lead to high model variance in regression, with concomitantly high MSE. This fact is evident from the definition of the covariance matrix for $\hat{\boldsymbol{\beta}}$:

$$\boldsymbol{\sigma}^2(\hat{\boldsymbol{\beta}}) = \sigma^2(\mathbf{X}^T\mathbf{X})^{-1}. \tag{31}$$

To derive Equation 31 recall the definition of $\hat{\boldsymbol{\beta}}$ from Equation 27. Further, let $\mathbf{A}$ be the matrix

$$\mathbf{A} = (\mathbf{X}^T\mathbf{X})^{-1}\mathbf{X}^T.$$

Substituting $\mathbf{A}$ we have

$$\begin{aligned}
\boldsymbol{\sigma}^2(\hat{\boldsymbol{\beta}}) &= \mathbf{A}\boldsymbol{\sigma}^2\{\mathbf{Y}\}\mathbf{A}^T \\
&= \mathbf{A}\sigma^2\mathbf{I}\mathbf{A}^T \\
&= (\mathbf{X}^T\mathbf{X})^{-1}\mathbf{X}^T\sigma^2\mathbf{I}\mathbf{X}(\mathbf{X}^T\mathbf{X})^{-1} \\
&= \sigma^2(\mathbf{X}^T\mathbf{X})^{-1}\mathbf{I} \\
&= \sigma^2(\mathbf{X}^T\mathbf{X})^{-1}.
\end{aligned}$$

Details of this derivation are available in (Neter et al., 1996, Chap. 5). To simplify explication (but without loss of generality) let us assume that the columns of $\mathbf{X}$ have been centered, standardized, and scaled such that the matrix $\mathbf{X}^T\mathbf{X} = \mathbf{R}$, where $\mathbf{R}$ is the $p \times p$ correlation

matrix of the predictors (thus we are working with the so-called standardized regression model). If we assume for now that $p = 2$ then we have

$$(\mathbf{X}^T\mathbf{X})^{-1} = \mathbf{R}^{-1} = \frac{1}{1 - r_{12}^2} \begin{pmatrix} 1 & -r_{12} \\ -r_{12} & 1 \end{pmatrix}$$

Thus in the two-predictor case, the variances of the regression coefficients (the main diagonal of the correlation matrix inverse) are proportional to $\frac{1}{1-r_{12}}$. As r_{12}, the predictor correlation, tends toward 1, the variance approaches infinity.

To generalize for $p > 2$ it helps to re-write the regression model in terms of the eigenvectors and eigenvalues of the predictors. Using the singular value decomposition we have

$$\mathbf{Z} = \mathbf{XV}$$

where $\mathbf{V}$ is the $p \times p$ matrix containing the eigenvectors of $\mathbf{X}^T\mathbf{X}$. We may then re-write Equation 28 as

$$\begin{aligned} \hat{\boldsymbol{\beta}} &= \mathbf{V}(\mathbf{Z}^T\mathbf{Z})^{-1}\mathbf{Z}^T\mathbf{Y} \\ &= \mathbf{V}\boldsymbol{\Sigma}^{-2}\mathbf{Z}^T\mathbf{Y} \end{aligned}$$

where $\boldsymbol{\Sigma}$ is the diagonal matrix of singular values. As shown in (Jolliffe, 2002, P. 171), the variance of the regression parameters depends directly on the magnitude of the corresponding eigenvalues of the predictor variables:

$$\begin{aligned} \sigma^2\hat{\boldsymbol{\beta}} &= \sigma^2\mathbf{V}(\mathbf{Z}^T\mathbf{Z})^{-1}\mathbf{V}^T \\ &= \sigma^2\mathbf{V}\boldsymbol{\Sigma}^{-2}\mathbf{A}^T \\ &= \sigma^2\sum_{k=1}^{p}\lambda_k^{-1}\mathbf{v}_k\mathbf{v}_k \end{aligned}$$

where λ_k is the k^{th} eigenvalue and $\mathbf{v}_k$ is the k^{th} column of $\mathbf{V}$. High degrees of inter-predictor correlation lead to very small eigenvalues. Because the variance of the regression parameters depends inversely on the magnitude of the eigenvalues, high colinearity leads to high model variance, and by extension, high MSE.

The standard vector space model uses the complete term-document matrix to inform similarity measurements. This is equivalent to an LSI model using all p-dimensions (i.e. no

dimensionality reduction). The crucial point of this section is that because the classic vector space model includes all eigenvalues it is prone to high model variance. The extent of this risk rests on the degree to which the terms are correlated. In other words, using the classic vector model we pay a price for assuming term independence in the form of increased MSE.

4.4 Biased Regression

Due to inter-term correlations, the standard vector space model incurs the high variance that accompanies colinearity. However, LSI projects the data onto a basis in which the predictors become uncorrelated. Thus we may understand the optimality of LSI's projection insofar as it lowers the variance of the regression model by minimizing predictor variable correlation.

In this context, LSI's motivation is the same as so-called principal component regression (PCR), which is a member of a family of regression models that attempt to reduce model MSE by introducing a small amount of bias. The intuition behind principal component regression is simple. If we observe in our matrix $\mathbf{X}$ p predictor variables that evince problematic colinearity we may replace the observed predictors with their principal components. If we truncate the model, using only k components where $k < p$, our estimated $\hat{\boldsymbol{\beta}}$ will be slightly biased, but is likely to have far lower variance than a model fitted from the untransformed data.

4.4.1 The Standardized Regression Model

It is convenient to describe PCR in the context of the standardized regression model referred to above. Under the standardized model we transform both the predictors and the response variable.

$$\begin{aligned} Y_i' &= \frac{1}{\sqrt{n-1}}\left(\frac{Y_i - \bar{Y}}{\sigma_y}\right) \\ X_{ik}' &= \frac{1}{\sqrt{n-1}}\left(\frac{X_{ik} - \bar{X}}{\sigma_k}\right) \quad (k = 1 \ldots p) \end{aligned}$$

Here σ_y and σ_k are the standard deviations of Y and X_k, respectively.

Having transformed our variables, the standardized model is fit using the ordinary least-squares estimates, yielding a p-dimensional vector $\hat{\boldsymbol{\beta}}'$. We can easily undo the standardization to find the regression coefficients of the original data by the multiplication $\frac{\sigma_y}{\sigma_k}\hat{\boldsymbol{\beta}}_k'$.

4.4.2 Principal Component Regression

The following explication draws from (Jolliffe, 2002, Sec. 8.1). If $\mathbf{X}$ is the $n \times p$ matrix of standardized predictors, we find the values of the principal components for each observation:

$$\mathbf{Z} = \mathbf{XV}$$

where $\mathbf{V}$ contains the right singular vectors of $\mathbf{X}$ (i.e. the eigenvectors of $\mathbf{X}^T\mathbf{X}$). Using $\mathbf{Z}$ as our predictors, we define the regression model by

$$\mathbf{Y} = \mathbf{Z}\boldsymbol{\gamma} + \boldsymbol{\epsilon}. \tag{32}$$

Because $\mathbf{V}$ is orthonormal we can rewrite $\mathbf{X}\boldsymbol{\beta}$ as $\mathbf{XVV}^T\boldsymbol{\beta} = \mathbf{Z}\boldsymbol{\gamma}$ where

$$\hat{\boldsymbol{\beta}} = \mathbf{V}\hat{\boldsymbol{\gamma}}.$$

Though we may calculate $\hat{\boldsymbol{\gamma}}$ using the usual least squares estimation technique, we may also find it directly. Since $\mathbf{Z}$ is column orthogonal

$$\hat{\boldsymbol{\gamma}} = (\mathbf{Z}^T\mathbf{Z})^{-1}\mathbf{Z}^T\mathbf{Y} = \boldsymbol{\Sigma}^{-2}\mathbf{Z}^T\mathbf{Y} \tag{33}$$

where $\boldsymbol{\Sigma}$ is the diagonal matrix of singular values.

The advantage of using Equation 33 to estimate our model parameters lies in the ability to produce a model of reduced dimensionality. Above we noted that small eigenvalues contribute large amounts of variance to the final model. By truncating the model, discarding principal components with small corresponding eigenvalues, we reduce model variance, often with only a small penalty in terms of error. Thus we define the reduced rank model:

$$\hat{\boldsymbol{\gamma}}_k = (\mathbf{Z}_k^T\mathbf{Z}_k)^{-1}\mathbf{Z}_k^T\mathbf{Y} = \boldsymbol{\Sigma}_k^{-2}\mathbf{Z}_k^T\mathbf{Y} \tag{34}$$

where $\mathbf{Z}_k$ contains the first k columns of $\mathbf{Z}$ and $\boldsymbol{\Sigma}_k$ is the diagonal matrix containing the first k singular values of $\mathbf{X}$.

The decrease in variance comes at the expense of introducing bias into our model. This is the case because

$$\hat{\boldsymbol{\beta}}_k = \hat{\boldsymbol{\beta}} - \sum_{j=k+1}^{p} \lambda_k^{-1} \mathbf{v}_k \mathbf{v}_k^T \mathbf{X}^T \mathbf{Y} \tag{35}$$

where $\mathbf{v}_k$ is the k^{th} column of $\mathbf{V}$. Since the expected value of $\hat{\boldsymbol{\beta}} = \boldsymbol{\beta}$, $\hat{\boldsymbol{\beta}}_k$ is biased by the subtracted term in Equation 35. However, the bias in $\hat{\boldsymbol{\beta}}_k$ is often small in comparison to the reduction in variance achieved by projecting onto the k uncorrelated components with maximum variance.

While Rocchio's optimal query defines the one-dimensional projection in term space that minimizes the probability of document misclassification, LSI offers a k-dimensional basis that lowers the MSE of the IR problem's implicit regression function. Thus the LSI projection is optimal in the sense that is delivers an uncorrelated (and due to dimensionality reduction, statistically stable) basis for estimation.

4.5 An Example

This section introduces a simple example to ground our discussion of biased regression. We construct a scenario where strong colinearity is present in a regression model, next showing how dimensionality reduction reduces the expected error of this model. Our goal here is to demonstrate how it is that discarding information can improve a model, as this is at the heart of latent semantic indexing.

Let us assume that we have data generated by a multivariate normal distribution with 3-dimensional mean vector $\boldsymbol{\mu} = \mathbf{0}$ and 3×3 covariance matrix $\boldsymbol{\Sigma}$ containing 1 on the main diagonal, with 0.9 on all off-diagonals. Finally, we assume that there is a response variable Y that has the form

$$y_i = 5x_{i1} + 2x_{i2} - 3x_{i3} + \epsilon \tag{36}$$

where $\epsilon \sim N(0, 1)$. Thus we know in advance that our true regression parameters are

$$\boldsymbol{\beta} = \begin{pmatrix} 5 \\ 2 \\ -3 \end{pmatrix}$$

and the error factor $\epsilon = \sigma^2 = 1$.

In the context of retrieval we may think of each variable as a term, with Y being relevance to a particular query. Thus, terms one and two contribute positively to the relevance of a

True	Full Rank	Reduced PCR
5	5.533	2.353
2	1.672	4.101
-3	-3.363	-2.591

Table 1: Regression Coefficients

Model	$var(\beta_1)$	$var(\beta_2)$	$var(\beta_3)$	Total var
Full Rank	59.733	39.085	47.661	146.479
Reduced PCR	1.738	33.373	20.234	55.346

Table 2: Regression Coefficient Variances

document, while term 3 contributes negatively. But the strong correlation between terms makes the β's difficult to interpret (this is the logical problem with colinearity in regression).

If we draw N, say 100, documents from this distribution, along with 100 relevance scores Y we can use Equation 27 to calculate $\hat{\boldsymbol{\beta}}$, the least-squares estimate of $\boldsymbol{\beta}$.

From a statistical standpoint we would like to know, how good our estimate of the regression function is. Given a new document x_i, how close to the true relevance score y_i is $\hat{y}_i$ likely to be? The results of Section 4.3 give us to the tools to answer this question and to address the limitations of our model.

To inform this discussion, I generated a 100×3 matrix $\mathbf{X}$ from the distribution described above. I also generated 100 corresponding y's. Using these data, I fit a regression model, M_3 using the standard method. I also generated a second model, M_2, by using only the first two principal components to perform PCR.

The model coefficients — the true parameters and our estimates — appear in Table 4.5. Neither model found the "true" coefficients, but both were close. To see how close they were, I generated a second predictor matrix and response vector and applied each model to the unseen data. The sum of squared error on unseen data for the full rank model was 1934.076, while the reduced rank model had $SSE = 1931.148$, a modest difference in model fit. Yet there was a large difference in the variance of the models. The variance for each regression parameter is shown in Table 4.5. The full model had over twice the variance of the reduced model. Meanwhile, the bias introduced by discarding the weakest principal component was small, only -0.234. Thus we have $MSE(full) = 0^2 + 146.479 = 146.479$ for the full model, with the reduced rank model at $MSE(PCR) = -0.234^2 + 55.346 = 55.404$.

To gain a more thorough understanding of the behavior of each model – the full regression and the reduced rank PCR – I repeated the process described above, generating 1000 training matrices (each 100×3) and 1000 corresponding response vectors. For each sample, I trained

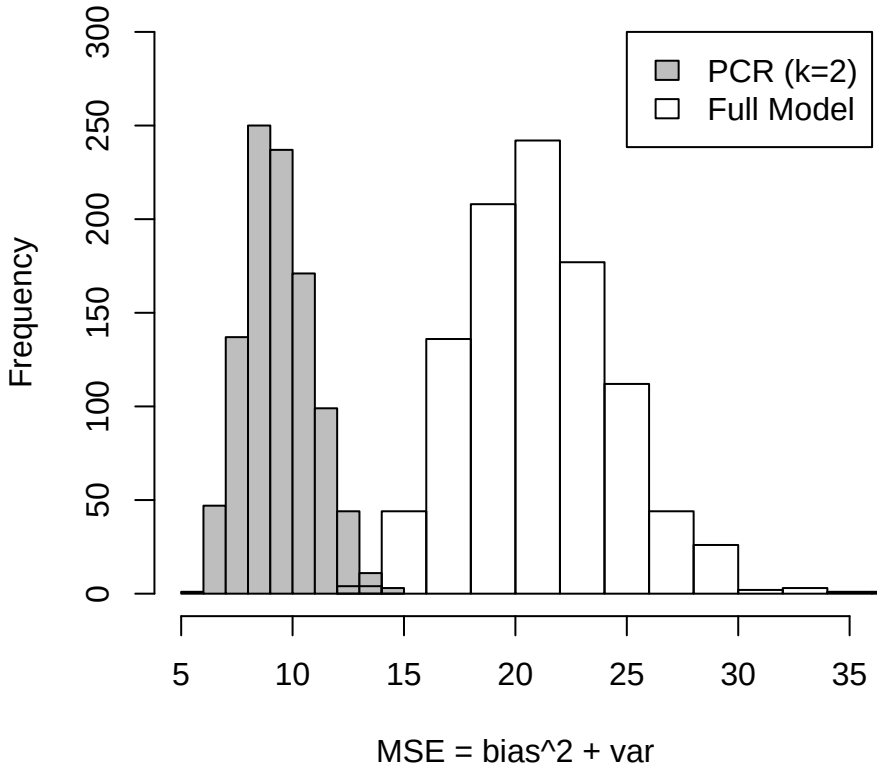

Figure 6: Mean-Squared Error on 1000 Simulated Data Sets

a full regression model and a reduced rank principal component regression model. I also generated a second set of data for each model (predictor matrix and response vector) to estimate accuracy on new data.

Results of these simulations suggest that the reduced-rank model was preferable to the full regression. With respect to prediction accuracy the full model delivered an average (across the 1000 trials) SSE of 1941.055 while the reduced rank model's SSE was 1939.767. This is not an overwhelming reduction in error. But a paired t-test on the equality of mean SSE across models yielded $p = 0.0001$. On average, the prediction error of the reduced model is lower than that of the full-rank model.

Figure 6 can help us understand why a truncated model can outperform a model trained with more information. The figure shows the MSEs under each model. MSEs for the reduced rank PCR model appear in gray, with the full model in white. The PCR models tend to have lower MSE than the standard models (a t-test on the equality of means for the MSE under each model yielded $p \ll 0.01$). While the full model is unbiased, it tends to have

an inflated variance due to the colinearity among the predictors. This variance makes itself felt when we attempt to predict the response for unseen data. That is, the full-rank model has overfitted the training data. By projecting the predictors onto an uncorrelated space of reduced dimensionality, the reduced-rank model admits a small amount of bias. But it does so with a large reduction in undesirable variance.

Admittedly, this example operates at a high level of abstraction. What does it show us about LSI in particular, and more generally, about retrieval? In the context of retrieval, predictor variables are usually terms that occur in documents. When we use the classic vector space model we assume that these terms are uncorrelated. Understanding LSI in the context of biased regression can help us understand how this oversimplification can impede prediction.

LSI replaces documents' observed features with a new set of features that are, by definition, uncorrelated. Mathematically this implies that the variance of our predictions will drop if we use LSI's low-dimensional approximation. More heuristically, it means that LSI assuages some of the brittleness of keyword matching. Thus a query for *cars* might return documents about *automobiles*. The classic vector space model suffers from high variance in the sense that numerous words can have many meanings and concepts can be expressed by different words. Just as we reduced the variance of the models reported in this section, LSI reduces the variance of the vector space model by discarding what is hopefully redundant information.

5 Discussion

In this study the Rocchio optimal query never gave a probability of misclassification higher than a projection onto the first LSI dimension. But LSI's dimensionality reduction was able to reduce the mean squared error of a regression model dramatically. Insofar as they are both solutions to least-squares problems, Rocchio relevance feedback and LSI are obviously both "optimal." Yet they are optimal with respect to different problems. This study has pursued this difference using as a foil the distinction between classification and regression. But the relationship between Rocchio and LSI is not so simple. We could, for example, reduce dimensionality via the SVD and find the Rocchio-optimal direction in the resulting k-space (in fact this is common practice).

While the difference between LSI and Rocchio appears clearly when we bring them to bear on classification and regression, their respective notions of optimality are more fundamentally distinct. Relevance feedback is essentially a supervised learning task, while LSI is unsupervised. But even this distinction is not satisfactory. David Hull's notion of local LSI (Hull, 1995) is predicated on a putative supervisory structure. Meanwhile, real implementations of relevance feedback are at best semi-supervised.

Perhaps the clearest way to articulate their relationship is to discuss each model's relationship to the data. The Rocchio optimal query is determined completely by the mean vectors

of the relevant and non-relevant documents. On the other hand, LSI derives dimensions that reflect the covariance structure of the corpus. The Rocchio optimal query separates relevant and non-relevant mean vectors as widely as possible. The matrix $\mathbf{V}_k$ in LSI gives an estimate of the term co-occurrence matrix $\mathbf{X}^T\mathbf{X}$ that we hope suffers less variance than the overdetermined sample.

Despite these differences, though, LSI and Rocchio relevance feedback share a great deal of underlying theory and motivations. Both methods assume that artifacts of human language (e.g. queries and documents) are parsimonious. That is, even at the most basic level documents' semantics exceed their vocabulary. Language is redundant and ambiguous. To remedy this, both methods attempt to learn from the corpus of documents in efforts to improve the vector space model's basic relevance estimate. In the case of Rocchio relevance feedback and LSI, this learning is framed as a least-squares optimization problem.

6 Conclusion

This study has pursued the notions of optimality that inform latent semantic indexing and Rocchio relevance feedback. Both techniques are extensions of the classic vector space IR model, and both are based on the method of least-squares. While their similarities are well-known, this paper has articulated the relationship between relevance feedback and LSI in a new light. Our argument showed that the methods' difference is not simply one of query expansion versus document expansion. We may understand Rocchio relevance feedback in the context of linear discriminant analysis, while LSI is a close relative of principal component analysis. This distinction leads each technique to excel at different problems. Relevance feedback provides a better 1-dimensional projection for document classification than LSI does (under the narrow constraints set forth here). On the other hand, LSI provides a k-dimensional uncorrelated basis that represents documents with minimal error. Thus, in the presence of high inter-term covariance, LSI's dimensionality reduction allows us to construct a linear model such as a regression with lower variance than we can in the observed space.

The notions of optimality that inform LSI and Rocchio's model are well studied. But they are not without flaws. In particular, least-squares methods have close ties to the normal distribution. Since linguistic data often defies normality, projections based on alternative optimization strategies are of great interest to IR researchers. How do projection methods such as probabilistic LSA (Hofmann, 1999) or independent component analysis (Hyvarinen, Karhunen, & Oja, 2001) relate to LSI? These are questions I hope to address in future research.

Additionally, in future work I hope to complement the theoretical approach taken in this paper with experimental data. Of particular interest will be the geometric relations between optimal queries (in Rocchio's sense) and principal components of the data. The advantages that Rocchio's method saw with respect to classification in this study were predicated on very

narrow assumptions; would they be evident in a less controlled environment? On the other hand, it will be of interest to learn whether term-term covariances are problematic enough that dimensionality reduction mitigates model variance to a significant degree. Empirical results, to be sure, will complicate the understanding of projection optimality outlined here.

References

Anderson, T. W. (2003). *An Introduction to Multivariate Statistical Analysis*. Wiley, New York.

Bast, H., & Majumdar, D. (2005). Why spectral retrieval works. In *SIGIR '05: Proceedings of the 28th annual international ACM SIGIR conference on Research and development in information retrieval*, pp. 11–18 New York, NY, USA. ACM Press.

Buckley, C., Salton, G., Allan, J., & Singhal, A. (1994). Automatic Query Expansion Using SMART: TREC 3. In *Overview of the Third Text REtrieval Conference*, pp. 69–80.

Deerwester, S. C., Dumais, S. T., Landauer, T. K., Furnas, G. W., & Harshman, R. A. (1990). Indexing by Latent Semantic Analysis.. *JASIS*, *41*(6), 391–407.

Duda, R. O., Hart, P. E., & Stork, D. G. (2001). *Pattern Classification* (2nd edition). Wiley-Interscience Publication, New York, NY.

Dumais, S. T. (1992). LSI meets TREC: A Status Report. In *Text REtrieval Conference*, pp. 137–152.

Dumais, S. T. (2004). Latent semantic analysis. *Annual Review of Information Science and Technology (ARIST)*, *38*, 189–230.

Eckart, C., & Young, G. (1936). The approximation of one matrix by another of lower rank. *Psychometrika*, *1*, 211–218.

Efron, M. (2005). Eigenvalue-based model selection during latent semantic indexing. *Journal of the American Society for Information Science and Technology*, *56*(9), 969–988.

Golub, G. H., & Van Loan, C. F. (1996). *Matrix Computations (Johns Hopkins Studies in Mathematical Sciences)* (3rd edition). The Johns Hopkins University Press, Baltimore.

Hastie, T., Tibshirani, R., & Friedman, J. H. (2001). *The Elements of Statistical Learning*. Springer.

Hofmann, T. (1999). Probabilistic latent semantic indexing. In *SIGIR '99: Proceedings of the 22nd annual international ACM SIGIR conference on Research and development in information retrieval*, pp. 50–57 New York, NY, USA. ACM Press.

Hull, D. A. (1995). *Information Retrieval using Statistical Classification. Ph.D. thesis*. Stanford University.

Hyvarinen, A., Karhunen, J., & Oja, E. (2001). *Independent Component Analysis*. Wiley-Interscience.

Jolliffe, I. T. (2002). *Principal Component Analysis* (2nd edition). Springer, New York.

Kontostathis, A., & Pottenger, W. M. (2006). A framework for understanding Latent Semantic Indexing (LSI) performance.. *Inf. Process. Manage.*, *42*(1), 56–73.

Losee, R. M. (1998). *Text Retrieval and Filtering: Analytic Models of Performance*. Kluwer, Boston.

Neter, J., Kutner, M. H., Nachtsheim, C. J., & Wasserman, W. (1996). *Applied Linear Statistical Models* (4th edition). Irwin, Chicago.

Plackett, R. L. (1972). The discovery of the method of least squares. *Biometrika*, *59*(2), 239–251.

Rocchio, J. J. (1971). *Relevance Feedback in Information Retrieval*, pp. 313–323. Prentice Hall, Englewood Cliffs, NJ.

Ruthven, I., & Lalmas, M. (2003). A survey on the use of relevance feedback for information access systems. *Knowledge Engineering Review*, *18*(2), 95–145.

Salton, G., Wong, A., & Yang, C. S. (1975). A vector space model for automatic indexing. *Commun. ACM*, *18*(11), 613–620.

Salton, G., & Buckley, C. (1997). *Improving retrieval performance by relevance feedback*, pp. 355–364. Morgan Kaufmann Publishers Inc., San Francisco, CA, USA.

Story, R. E. (1996). An Explanation of the Effectiveness of Latent Semantic Indexing by Means of a Bayesian Regression Model. *Information Processing and Management*, *32*(3), 329–344.

Wong, S. K. M., Ziarko, W., & Wong, P. C. N. (1985). Generalized vector spaces model in information retrieval. In *SIGIR '85: Proceedings of the 8th annual international ACM SIGIR conference on Research and development in information retrieval*, pp. 18–25 New York, NY, USA. ACM Press.