CS 674/INFO 630: Advanced Language Technologies **Fall 2007**

Lecture 17 — October 30

Lecturer: Lillian Lee *Scribes: Nikos Karampatziakis & Ainur Yessenalina*

1 Connection to the Previous Lecture

At the end of the previous lecture we were talking about how to incorporate implicit relevance feedback which came in the form of preferences, i.e. instead of absolute judgments (this document is relevant and that document is not) we had information from clickthrough data in the form of relative judgments (this document is more relevant than that document). We ended up with some sort of vector space model. We considered "items" in a vector space, the "items" were little odd: query-and-document vectors rather than document vectors. So we were back to the vector space model, maybe a bit different than the one we were used to, but given that, this is a good time to go back and talk about the term-document matrix.

Let's consider more generalized features than just term counts. We will allow more general attributes and in fact our features can be negative. For example, we can have a feature that represents the number of words in the document relative to the average document length. We can also have some features like the number of English terms minus the number of Finnish terms. We are back to a vector space representation. We are going to refer to the entries in the document vectors as "terms" for convenience, but we will actually mean more general attributes or features.

2 The Term-Document Matrix

The entire corpus can be represented as a term-document matrix D. D consists of document vectors $\vec{d_i}$. The document vectors are the columns of this matrix. Document vector $\vec{d_i}$ contains m generalized terms

$$D = \begin{bmatrix} | & | & \dots & | \\ \vec{d_1} & \vec{d_2} & \dots & \vec{d_n} \\ | & | & \dots & | \end{bmatrix}, \quad D \in \mathbb{R}^{m \times n}.$$

Since the terms might not be of the same type (i.e. one term may be a term count, one may be some relative quantity) it no longer makes sense to normalize across terms. So, we assume that the columns of D are not normalized document vectors, because we consider generalized terms.

Before, we were considering the document vectors individually. We have not addressed the question of how the documents in the corpus relate to each other, except implicitly with respect to the query and in our consideration of the IDF (see below). Now let's ask a different

question: Suppose we have a term-document matrix D, is there any way to get a succinct representation of the information within the term-document matrix? Can we understand the corpus in a compact way from the term-document matrix? What succinct descriptions of corpus structure can be derived from D?

Why do we care? We have a query and we have to find documents relevant to that query. How does this help us at all? The idea is to get a better representation of the documents that corresponds to the structure of the corpus, since a better representation may help us do better retrieval.

We have seen before some sort of analogy to what we are looking for. Inverse document frequency (IDF) considers the whole document corpus. IDF looks at the distribution of the terms in the whole corpus and shrinks term frequencies for terms that occur a lot in the corpus. IDF thus uses the overall corpus characteristics to alter our document representation.

Let's ask an "easy" question first. How "spread out" are the documents? (How varied is the corpus C?) We should think of the $\vec{d}_i$'s as vectors and look at some geometric notion of variation. Therefore to answer the previous question one can consider their *span* (also known as the column space of D). If the document vectors are spread out, then they have a big span. The span of a set of vectors is the set of all possible linear combinations of these vectors. Hence the span is a space. We can use the *dimensionality* of the span to summarize that space. The dimensionality of the span of $\{\vec{d}_1, \vec{d}_2, \ldots, \vec{d}_n\}$ is denoted as rank(D) and represents the number of vectors needed for a basis for the span. If the vectors are really spread out we will need a lot of basis vectors to describe them and if they are close to each other we may need only few basis vectors.[1] A bigger rank means that the vectors are more spread out. But we can have different sets of vectors whose column spaces have the same rank but do not look the same structurally. For example, in Figure 1 we have two different corpora but our intuition says that the one on the left is more "spread out" than the one on the right.

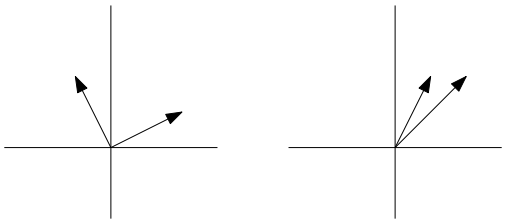

Figure 1: Two sets of document vectors. The one on the left is more "spread out" than the one on the right

We want more information on the corpus structure, because we want to be able to tell the difference between two sets of vectors with the same rank. Let's consider an easy case

[1]It is worthwhile to note that in this setting we can use many tools and theorems from linear algebra. For example, a non-obvious fact about ranks is that rank(D^T)=rank(D), a proof of which appears in the finger exercises.

when our corpus has rank one. This means that one basis vector suffices. In other words, there exists a $\vec{b} \in \mathbb{R}^m - \{0\}$ with $||\vec{b}||_2 = 1$ such that for every column $\vec{d_i}$ of D there exists an $\alpha \in \mathbb{R}$ such that $\vec{d_i} = \alpha \cdot \vec{b}$.

This means that $\vec{b}$ is some sort of prototypical term profile. Every document is a multiple of the prototypical term profile. A basis vector is a good representation of our corpus, because we can describe every document in the corpus using this vector.

How many $\vec{b}$'s can we find in an one dimensional space, such that $||\vec{b}||_2 = 1$? The answer is two. In an one dimensional space $\vec{b}$ is unique up to sign. If there was no normalization constraint then we would have infinite possible choices for a basis vector.

What happens when rank(D)= 2? Here we have a basis consisting of two vectors. There are a lot of possible choices for the basis vectors. Even if we normalize our basis vectors and make an additional constraint that they are orthogonal, still our basis will not be unique up to rotation. In the general case there is no unique basis. So this means we cannot rely on the rank of D (i.e, the dimensionality of the column space) to give us uniquely specifying information about the corpus.

3 Some Specific Examples

Now let's proceed in the following direction. Let's not consider all possible linear combinations of column vectors (which is what the span is). Let's restrict our attention to specific linear combinations of the columns of D. A useful fact regarding these linear combinations is that

$$\sum_{i=1}^{n} \alpha[i] \cdot \vec{d_i} = D \cdot \vec{\alpha}, \quad \text{where} \quad \vec{\alpha} = \begin{bmatrix} \alpha[1] \\ \alpha[2] \\ \vdots \\ \alpha[n] \end{bmatrix},$$

a proof of which appears in the finger exercises. Hence we can think of the term-document matrix D as an operator: $\mathbb{R}^n \to \mathbb{R}^m$. We can interpret $D \cdot \vec{\alpha}$ as applying an operator D to the coefficients $\vec{\alpha}$. Let's see what happens when we apply the operator D to some specific coefficients. What are some interesting coefficients? A conceptually easy case is when the $\alpha[i]$'s are "fractional assignments":

$$\mathfrak{a} = \{\alpha \in \mathbb{R}^n \mid \alpha[i] \geq 0, \ \sum_{i=1}^{n} \alpha[i] = 1\}.$$

To gain some intuition let's restrict D to be a 2×3 matrix (two terms and three documents) which can also be thought as a linear operator $D : \mathbb{R}^3 \to \mathbb{R}^2$ (i.e. it acts on vectors in $\mathbb{R}^3$ to produce vectors in $\mathbb{R}^2$). What we would like to do is to look at three different such matrices that represent different types of corpus structure and see if the tools we have from linear algebra can tell the difference between them. Instead of using the matrix notation it will prove useful to consider the corpus as three document vectors in $\mathbb{R}^2$. Figure 2 shows three such corpora. Corpus D' (left) has a lot of variety as we can see from the vectors being

spread out. In contrast, corpus D'' (center) exhibits little variation and corpus D''' (right) exhibits variation mostly in one direction (all vectors point roughly to the same direction but they have different lengths). Now let's consider these corpora as matrices acting on the

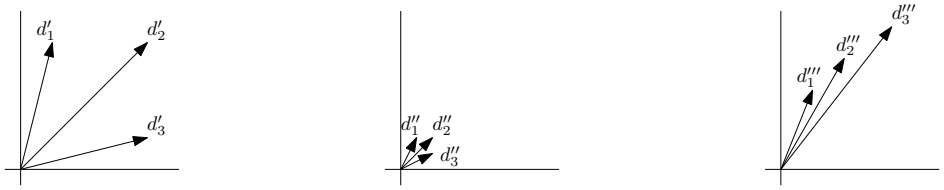

Figure 2: Three corpora. Left: "spread out" corpus D' , center: corpus D'' with little variation, right: corpus D''' with directed variation.

simplex $\mathfrak{a}$. In our example, this simplex has three vertices which define a triangle. If we apply the matrices corresponding to the corpora D', D'', and D''' to $\mathfrak{a}$, the images of these transformations are still triangles because these are linear transformations. The simplex $\mathfrak{a}$ is the convex hull of its vertices $\vec{v}_1 = [1, 0, 0]^T, \vec{v}_2 = [0, 1, 0]^T$ and $\vec{v}_3 = [0, 0, 1]^T$ and therefore its image after some corpus D acts on it will be the convex hull of the images of its vertices. To see this let

$$\vec{z} = \lambda_1 \vec{v}_1 + \lambda_2 \vec{v}_2 + \lambda_3 \vec{v}_3$$

be a point on simplex $\mathfrak{a}$, where $\lambda_1 + \lambda_2 + \lambda_3 = 1$ and $\lambda_i \geq 0$ for $j = 1, 2, 3$. Then

$$D\vec{z} = \lambda_1 D\vec{v}_1 + \lambda_2 D\vec{v}_2 + \lambda_3 \vec{D} v_3$$

The images of the vertices of the simplex are simply the document vectors themselves:

$$D\vec{z} = \lambda_1 \vec{d}_1 + \lambda_2 \vec{d}_2 + \lambda_3 \vec{d}_3.$$

Since $D\vec{z}$ can be expressed as a convex combination of the document vectors it must be inside their convex hull. In other words, the simplex $\mathfrak{a}$ is mapped to the convex hull of the document vectors. Figure 3 shows how each of our three corpora maps the simplex $\mathfrak{a}$ onto $\mathbb{R}^2$. By looking at the left and center convex hulls in Figure 3 one might conclude that one way to measure the variation in these corpora could be the volume of these convex hulls. However, the convex hull on the right also has small volume even though it's very different from the one in the center. So the volume itself is not enough to describe the differences between these corpora. We need a better way to describe the shape of the corpus.

Let's try a different set of linear combinations of documents. We define

$$\mathfrak{a}_1 = \{\vec{\alpha} \in \mathbb{R}^n \mid ||\vec{\alpha}||_1 = 1\}$$

which is a proper superset of $\mathfrak{a}$. In our running example $\mathfrak{a}_1$ is an octahedron with six vertices $[\pm 1, 0, 0]^T, [0, \pm 1, 0]^T$ and $[0, 0, \pm 1]^T$. We expect that when $\mathfrak{a}_1$ is mapped onto $\mathbb{R}^2$ by our corpora D', D'' and D''', its image will be a hexagon and each of its vertices will be mapped

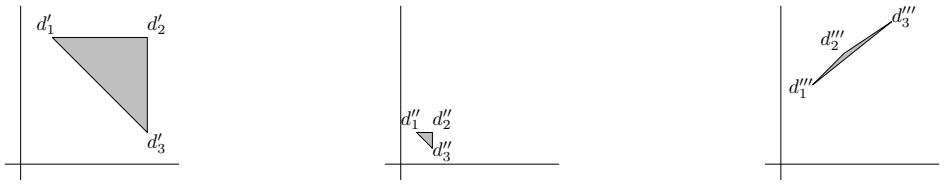

Figure 3: Left: the image of $D'\mathfrak{a}$, center: the image of $D''\mathfrak{a}$, right: the image of $D'''\mathfrak{a}$.

to a corner of this hexagon. This means that some of the eight faces of the octahedron will have to overlap when it is mapped to $\mathbb{R}^2$. As before, we expect the image of $\mathfrak{a}_1$ under D', D'' and D''', to be a convex hull. This time however we claim that it will be the convex hull of the document vectors and their negatives. Figure 4 shows these convex hulls for D', D'' and D'''. In this figure we can tell the differences between the three corpora by looking at the shapes of the hexagons, which are quite distinct. However, we are using six document vectors (the corners of the hexagon) to describe a corpus with three documents. In general, in a corpus with n documents, this convex hull will have $2n$ corners which is undesirable because we are looking for a succinct representation of the corpus. We would like to remove the dependence on the number of documents and still be able to describe the shape of the corpus.

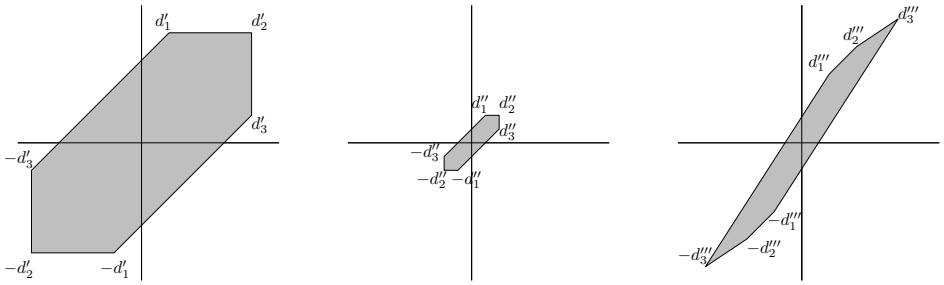

Figure 4: Left: the image of $D'\mathfrak{a}_1$, center: the image of $D''\mathfrak{a}_1$, right: the image of $D'''\mathfrak{a}_1$.

We can try a smooth approximation to these convex hulls. The reason we are getting $2n$ corners is because the set $\mathfrak{a}_1$ is a polyhedron with $2n$ vertices. To avoid this, let's use the following set

$$\mathfrak{a}_2 = \{\vec{\alpha} \in \mathbb{R}^n \mid ||\vec{\alpha}||_2 = 1\}$$

which is a hypersphere in $\mathbb{R}^n$. Notice that the vertices of $\mathfrak{a}_1$ still belong in $\mathfrak{a}_2$ so the corners of the hexagons in Figure 4 will still be part of the image of $\mathfrak{a}_2$. A linear transformation can only stretch and rotate a sphere so the image of $\mathfrak{a}_2$ will be a hyperellipse. A rough sketch of the ellipses for our corpora D', D'' and D''' are shown in Figure 5 along with the convex hulls that they are approximating. Note that we only need to specify the axes of the ellipses and their lengths. In our example we only need to specify two lengths and two axes in two dimensions but in general we will need to specify r lengths and r vectors in $\mathbb{R}^m$

where r =rank(D). The reason is that by multiplying any vector with the term document

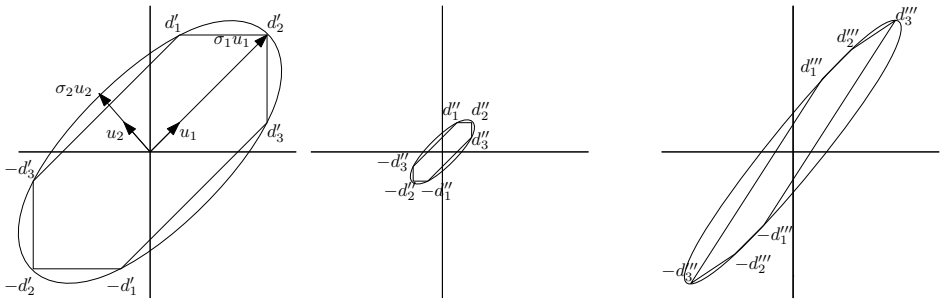

Figure 5: Rough sketches of $D'\mathfrak{a}_2$ (left), $D''\mathfrak{a}_2$ (center) and $D'''\mathfrak{a}_2$ (right)

matrix we can only get vectors that are linear combinations of D's columns. We already said that the dimensionality of the span of the columns is the rank of the matrix so r vectors are enough to form a basis for the span. The axes of the ellipse are orthogonal so they form a convenient basis. A somewhat simpler basis is an orthonormal one, consisting of orthogonal *unit* vectors. Choosing an orthonormal basis allows us to deal with the directions of the axes and their lengths separately. So we need to specify the lengths which by convention we take in descending order

$$\sigma_1 \geq \sigma_2 \geq \ldots \geq \sigma_r > 0$$

and a set of corresponding directions:

$$\vec{u}_1, \vec{u}_2, \ldots, \vec{u}_r.$$

In the left ellipse of Figure 5 we have drawn the directions $\vec{u}_1$ and $\vec{u}_2$ of the axes as well as the axes themselves $\sigma_1\vec{u}_1$ and $\sigma_2\vec{u}_2$. The values σ_i are called the singular values and in the next lecture we will see the connection of all these to the singular value decomposition of the term-document matrix.

4 Finger Exercises

1. Show that
$$\sum_{i=1}^{n} \alpha[i] \cdot \vec{d}_i = D \cdot \vec{\alpha}$$

2. In this question we will establish a series of facts in order to show that rank(D)=rank(D^T). Let's first recall some linear algebra terms. Let $D \in \mathbb{R}^{m \times n}$. The *null space* of D is the set of all vectors $\vec{x}$ such that $D\vec{x} = \vec{0}$. The *column space* of D is the span of its columns. The *row space* of D is the span of its rows (equivalently, the column space of D^T).

(i) Show that the vectors in the null space of D are orthogonal to vectors in the row space of D. Which vectors belong to both spaces?

(ii) Show that multiplying the vectors in a basis for the row space by D gives the same number of linearly independent vectors in the column space. Can the dimensionality of the column space be less than that of the row space?

(iii) Apply what you showed in (ii) to D^T to show that $\text{rank}(D^T) = \text{rank}(D)$.

3. In lecture we saw what happened when we apply D to different sets of coefficients. In this exercise we will see what happens when we apply D to a random vector of coefficients. Suppose $\vec{x}$ is a random vector in $\mathbb{R}^n$ normally distributed with mean $\vec{0}$ and covariance matrix $\sigma^2 I$. What is the probability distribution of $\vec{y} = D\vec{x}$, what is its mean and covariance matrix? What is the shape of the set of all points with equal probability density, say c, for the distribution of $\vec{x}$? And for the distribution of $D\vec{x}$? You may make use of the following fact without proof: any symmetric matrix A can be written as a product $Q\Lambda Q^T$ where Q is a matrix whose columns are the eigenvectors of A and Λ is a diagonal matrix with the eigenvalues of A on the diagonal.

4. The approach we have taken so far seems to be only applicable to the vector space model. How can we do corpus structure analysis in the language modeling paradigm?

5 Solutions

1.
$$D\vec{\alpha} = \begin{bmatrix} d_1[1] & d_2[1] & \dots & d_n[1] \\ d_1[2] & d_2[2] & \dots & d_n[2] \\ \vdots & \vdots & \ddots & \vdots \\ d_1[m] & d_2[m] & \dots & d_n[m] \end{bmatrix} \cdot \begin{bmatrix} \alpha[1] \\ \alpha[2] \\ \vdots \\ \alpha[n] \end{bmatrix} =$$

$$\begin{bmatrix} \alpha[1]d_1[1] + \alpha[2]d_2[1] + \cdots + \alpha[n]d_n[1] \\ \alpha[1]d_1[2] + \alpha[2]d_2[2] + \cdots + \alpha[n]d_n[2] \\ \vdots \\ \alpha[1]d_1[m] + \alpha[2]d_2[m] + \cdots + \alpha[n]d_n[m] \end{bmatrix} = \alpha[1]\vec{d_1} + \alpha[2]\vec{d_2} + \cdots + \alpha[n]\vec{d_n} = \sum_{i=1}^{n} \alpha[i]\vec{d_i}.$$

.

2. The following proof is by no means the only proof of this fact. Using the SVD of D and the fact that $D^T = V\Sigma U^T$ the desired result can follow easily. However, we think that the following proof is more intuitive.

(i) If a vector $\vec{x}$ is in the null space of D, then $D\vec{x} = \vec{0}$. The row space of D is the same as the column space of D^T. Therefore if a vector $\vec{y}$ is in the row space of D, it can be written as a linear combination of the columns of D^T. From the first

finger exercise we know that such a linear combination can be expressed as $D^T\vec{\lambda}$ for some $\vec{\lambda} \in \mathbb{R}^m$. But if $\vec{y} = D^T\vec{\lambda}$ for some $\vec{\lambda} \in \mathbb{R}^m$ then

$$\vec{y} \cdot \vec{x} = (D^T\vec{\lambda})^T \cdot \vec{x} = \vec{\lambda} \cdot (D\vec{x}) = \vec{\lambda} \cdot \vec{0} = 0$$

so $\vec{y}$ and $\vec{x}$ are orthogonal. A vector z that belongs to both spaces has to be orthogonal to itself. That is $\vec{z} \cdot \vec{z} = 0$ or $\sum_{i=1}^{n} z[i]^2 = 0$. This can only happen when $\vec{z} = \vec{0}$.

(ii) This part of the proof is heavily based on [1], which gives a very elegant proof of what we are showing here. As Mackiw notes, given $D \in \mathbb{R}^{m\times n}$, let the vectors $\vec{x}_1, \vec{x}_2, \ldots, \vec{x}_r \in \mathbb{R}^n$ form a basis for the row space of D. Then the vectors $D\vec{x}_1, D\vec{x}_2, \cdots, D\vec{x}_r$ are in the column space of D and further we claim that they are linearly independent. For, if $c_1 D\vec{x}_1 + c_2 D\vec{x}_2 + \cdots + c_r D\vec{x}_r = 0$ for some real scalars $c_1, c_2, \ldots, c_r$ then $D(c_1\vec{x}_1 + c_2\vec{x}_2 + \cdots + c_r\vec{x}_r) = 0$ and the vector $\vec{v} = c_1\vec{x}_1 + c_2\vec{x}_2 + \cdots + c_r\vec{x}_r$ would be in the null space of D. But $\vec{v}$ is also in the row space of D since it is a linear combination of basis elements. So, $\vec{v}$ is the zero vector and the linear independence of $\vec{x}_1, \vec{x}_2, \vec{x}_3, \ldots, \vec{x}_r$ guarantees that $c_1 = c_2 = \cdots = c_r = 0$. The existence of r linearly independent vectors in the column space requires that the dimensionality of the column space is at least as big as r, the dimensionality of the row space.

(iii) Now we can apply the fact that we just showed in (ii) to D^T and get that the dimensionality of its column space is at least as big as the dimensionality of its row space. But the column space of D^T is the row space of D and the row space of D^T is the column space of D. Combining this with what we have from (ii) we conclude that the dimensionality of the row space of D is equal to the dimensionality of the column space of D. Equivalently, the dimensionality of the column space of D^T is equal to the dimensionality of the column space of D or simply rank(D^T)=rank(D).

3. In the following we will treat all vectors as $n \times 1$ or $m \times 1$ matrices. A linear transformation of a normally distributed random variable still results in a normally distributed random variable. For the mean we have

$$E[\vec{y}] = E[D\vec{x}] = D \cdot E[\vec{x}] = D \cdot \vec{0} = \vec{0}.$$

where the second step is justified by linearity of expectation. Notice that $E[\vec{x}]$ is a $n \times 1$ vector while $E[\vec{y}]$ is a $m \times 1$ vector. For the covariance matrix we have

$$E[(\vec{y}-E[\vec{y}])(\vec{y}-E[\vec{y}])^T] = E[(\vec{y}-\vec{0})(\vec{y}-\vec{0})^T] = E[\vec{y}\vec{y}^T] = E[(D\vec{x})(D\vec{x})^T] = E[D\vec{x}\vec{x}^T D^T]$$

and again by linearity of expectation

$$E[(\vec{y} - E[\vec{y}])(\vec{y} - E[\vec{y}])^T] = DE[\vec{x}\vec{x}^T]D^T. \tag{1}$$

We know that the covariance matrix of the distribution of $\vec{x}$ is $\sigma^2 I$. That is

$$E[(\vec{x} - \vec{0})(\vec{x} - \vec{0})^T] = E[\vec{x}\vec{x}^T] = \sigma^2 I.$$

Substituting $E[\vec{x}\vec{x}^T]$ in (1) we get the covariance matrix

$$E[(\vec{y} - E[\vec{y}])(\vec{y} - E[\vec{y}])^T] = \sigma^2 DD^T.$$

So the distribution of $\vec{y} = D\vec{x}$ is the multivariate normal with mean zero and covariance matrix $\sigma^2 DD^T$:

$$p(\vec{y}) = \frac{1}{(2\pi\sigma^2)^{m/2}|DD^T|^{1/2}} e^{-\frac{1}{2\sigma^2}\vec{y}^T(DD^T)^{-1}\vec{y}}.$$

To find the points with equal probability density for the multivariate normal we solve $p(\vec{x}) = c$ for $\vec{x}$

$$\frac{1}{(2\pi\sigma^2)^{n/2}} e^{-\frac{1}{2\sigma^2}\vec{x}^T\vec{x}} = c$$

$$-\frac{1}{2\sigma^2}\vec{x}^T\vec{x} = \ln(c(2\pi\sigma^2)^{n/2})$$

$$\vec{x}^T\vec{x} = 2\sigma^2 \ln \frac{1}{c(2\pi\sigma^2)^{n/2}}$$

If we set $C_1 \stackrel{def}{=} 2\sigma^2 \ln \frac{1}{c(2\pi\sigma^2)^{n/2}}$ then the above equation can be written as

$$\sum_{i=1}^{n} x[i]^2 = C_1$$

which we can recognize as the equation of a hypersphere with radius $\sqrt{C_1}$. Thus, the distribution that $\vec{x}$ comes from assigns equal probability density to all the points in the same hypersphere. Can you guess where will the points with equal probability density be for the distribution of $D\vec{x}$? Let's see:

$$\frac{1}{(2\pi\sigma^2)^{m/2}|DD^T|^{1/2}} e^{-\frac{1}{2\sigma^2}\vec{y}^T(DD^T)^{-1}\vec{y}} = c$$

$$-\frac{1}{2\sigma^2}\vec{y}^T(DD^T)^{-1}\vec{y} = \ln(c(2\pi\sigma^2)^{m/2}|DD^T|^{1/2})$$

$$\vec{y}^T(DD^T)^{-1}\vec{y} = \underbrace{2\sigma^2 \ln\left(\frac{1}{c(2\pi\sigma^2)^{m/2}|DD^T|^{1/2}}\right)}_{C_2}$$

Let's set $A \stackrel{def}{=} (DD^T)^{-1}$. Then the points with equal probability density satisfy the equation

$$\vec{y}^T A\vec{y} = C_2. \tag{2}$$

The quadratic form on the left hand side can be written as

$$\sum_{i=1}^{m}\sum_{j=1}^{m} a_{ij}y[i]y[j] = C_2.$$

You might have expected to find the equation of an ellipse in its *canonical form* $\sum_{i=1}^{m} \lambda[i]y[i]^2 = C_2$ for some $\lambda[i] \geq 0$. The above equation doesn't look like that. So does that mean that this equation doesn't describe an ellipse? Well no, because the ellipse equation in canonical form is describing an unrotated ellipse centered at the origin. Our ellipse may be rotated, in which case its equation will not be in canonical form. Can we find a rotation such that in the new coordinate system the above equation looks like an ellipse? In other words we would like to find a matrix Q whose columns form an orthonormal basis of $\mathbb{R}^m$ such that $\vec{y}$ is the rotated version of some other vector $\vec{z}$ ($\vec{y} = Q\vec{z}$) and $\vec{z}$ satisfies the ellipse equation in canonical form

$$\sum_{i=1}^{m} \lambda[i]z[i]^2 = C_2$$

for some $\lambda[i] \geq 0$. We can write this in matrix form

$$\vec{z}^T \Lambda \vec{z} = C_2$$

where the matrix Λ is a diagonal matrix with the values $\lambda[i]$ across the diagonal. Because the columns of Q are orthonormal we have that $Q^TQ = I$ or $Q^{-1} = Q^T$. This means that $\vec{z} = Q^T\vec{y}$ and substituting above gives

$$\vec{y}^T Q\Lambda Q^T \vec{y} = C_2.$$

By comparing this with equation (2) we conclude that to get the ellipse equation in canonical form we need to be able to factorize A as

$$A = Q\Lambda Q^T \tag{3}$$

with the columns of Q orthonormal and $\lambda[i] \geq 0$. If A is symmetric we know from the problem statement that such a factorization is possible.

We can easily show that A^{-1} is symmetric. Let the SVD of D be $D = \mathcal{U}\mathcal{S}\mathcal{V}^T$. Then

$$A^{-1} = DD^T = \mathcal{U}\mathcal{S}\mathcal{V}^T(\mathcal{U}\mathcal{S}\mathcal{V}^T)^T = \mathcal{U}\mathcal{S}\mathcal{V}^T\mathcal{V}\mathcal{S}\mathcal{U}^T = \mathcal{U}\mathcal{S}^2\mathcal{U}^T$$

because $\mathcal{V}^T\mathcal{V} = I$. However $(DD^T)^T = (\mathcal{U}\mathcal{S}^2\mathcal{U}^T)^T = \mathcal{U}\mathcal{S}^2\mathcal{U}^T$ so DD^T is symmetric. To show that $A = (DD^T)^{-1}$ is symmetric we can show that the inverse of any symmetric matrix M is symmetric. We start from the identity

$$M^{-1}M = I$$

which can be written as

$$M^{-1}M^T = I.$$

Transposing both sides of the equality gives

$$(M^{-1}M^T)^T = I$$

$$M(M^{-1})^T = I$$

and left multiplying by M^{-1} gives

$$M^{-1} = (M^{-1})^T$$

which is the desired result. We also need to show that the eigenvalues $\lambda[i]$ of A are greater than zero. First, A^{-1} is positive semidefinite because for any vector $\vec{x}$

$$\vec{x}^T A^{-1}\vec{x} = \vec{x}^T DD^T\vec{x} = (D^T\vec{x})^T(D^T\vec{x}) \geq 0.$$

so all its eigenvalues are greater than zero. But if $\lambda[i]$ is an eigenvalue of A and $\vec{e}$ is the corresponding eigenvector then $A\vec{e} = \lambda[i]\vec{e}$ and by multiplying with A^{-1} we get

$$A^{-1}\vec{e} = \frac{1}{\lambda[i]}\vec{e}$$

which means that $\frac{1}{\lambda[i]}$ is an eigenvalue of A^{-1}. Hence $\frac{1}{\lambda[i]} \geq 0$ or simply $\lambda[i] \geq 0$.
Therefore, we have an ellipse whose axes are in the directions of the eigenvectors of $(DD^T)^{-1}$ and the lengths of the axes are the eigenvalues of $(DD^T)^{-1}$.

4. This is more of a discussion and a partial solution rather than a well worked out answer. If we are not careful enough we may fall into the trap of thinking that we can easily apply all the things we have been discussing to the language modeling paradigm by considering the "parameter-document" matrix. This would be like the term-document matrix but the columns are now the vectors of multinomial parameters for the language model induced by each document. However these vectors don't even form a vector space (there are many ways to see this, e.g. they don't include the zero vector, adding two of them gives something that isn't a vector of multinomial parameters, etc). Moreover, the semantics of similarity are different for language models and document vectors. We have seen that an appropriate similarity measure for language models (probability distributions) is the K-L divergence. We would like to develop a method that resembles the SVD in the sense that it can learn a succinct representation of the corpus and can reconstruct the parameter-document matrix in some optimal way with respect to the K-L divergence (instead of the Frobenius norm).

References

[1] George Mackiw, *A Note on the Equality of the Column and Row Rank of a Matrix*, Mathematics Magazine, Vol. 68, No. 4. (Oct. 1995), pp. 285-286. (Accessible via JSTOR)