Linear Algebra Fundamentals

Representation, Basis and Dimension

Let us quickly review some notation and basic ideas from linear algebra:

Suppose that the matrix V is composed of the columns $\boldsymbol{v}_1, \ldots, \boldsymbol{v}_k$, and that these columns form a basis basis for some subspace, H, in $\mathbb{R}^n$ (notice that this implies $k \leq n$). Then every data point in H can be written as a linear combination of the basis vectors. In particular, if $\boldsymbol{x} \in H$, then we can write:

$$\boldsymbol{x} = c_1\boldsymbol{v}_1 + \ldots + c_k\boldsymbol{v}_k \doteq V\boldsymbol{c}$$

so that every data point in our subset of $\mathbb{R}^n$ is identified with a point in $\mathbb{R}^k$:

$$\boldsymbol{x} = \begin{bmatrix} x_1 \\ x_2 \\ \vdots \\ x_n \end{bmatrix} \longleftrightarrow \begin{bmatrix} c_1 \\ \vdots \\ c_k \end{bmatrix} = \boldsymbol{c}$$

The vector $\boldsymbol{c}$, which contains the coordinates of $\boldsymbol{x}$, is the **low dimensional representation** of the point $\boldsymbol{x}$. That is, the data point $\boldsymbol{x}$ resides in $\mathbb{R}^n$, but $\boldsymbol{c}$ is in $\mathbb{R}^k$, where $k \leq n$.

Furthermore, we would say that the subspace H (a subspace of $\mathbb{R}^n$) is *isomorphic* to $\mathbb{R}^k$. We'll recall the definition:

Definition 0.1. Any one-to-one (and onto) linear map is called an isomorphism. In particular, any change of coordinates is an isomorphism. Spaces that are isomorphic have essentially the same algebraic structure- adding vectors in one space is corresponds to adding vectors in the second space, and scalar multiplication in one space is the same as scalar multiplication in the second.

Definition 0.2. Let H be a subspace of vector space X. Then H has dimension k if a basis for H requires k vectors.

Given a linearly independent spanning set (the columns of V) to compute the coordinates of a data point with respect to that basis requires a matrix inversion (or more generally, Gaussian elimination) to solve the equation:

$$\boldsymbol{x} = V\boldsymbol{c}$$

In the case where we have n basis vectors of $\mathbb{R}^n$, then V is an invertible matrix, and we write:

$$\boldsymbol{c} = V^{-1}\boldsymbol{x}$$

If we have fewer than n basis vectors, V will not be square, and thus not invertible in the usual sense. However, if $\boldsymbol{x}$ is contained in the span of the basis, then we will be able to solve for the coordinates of $\boldsymbol{x}$.

Example 0.1. Let the subspace H be formed by the span of the vectors $\boldsymbol{v}_1, \boldsymbol{v}_2$ given below. Given the point $\boldsymbol{x}_1, \boldsymbol{x}_2$ below, find which one belongs to H, and if it does, give its coordinates.

$$\boldsymbol{v}_1 = \begin{bmatrix} 1 \\ 2 \\ -1 \end{bmatrix} \quad \boldsymbol{v}_2 = \begin{bmatrix} 2 \\ -1 \\ 1 \end{bmatrix} \quad \boldsymbol{x}_1 = \begin{bmatrix} 7 \\ 4 \\ 0 \end{bmatrix} \quad \boldsymbol{x}_2 = \begin{bmatrix} 4 \\ 3 \\ -1 \end{bmatrix}$$

SOLUTION: Rather than row-reduce twice, we'll do it once on the augmented matrix below.

$$\left[\begin{array}{rr|rr} 1 & 2 & 7 & 4 \\ 2 & -1 & 4 & 3 \\ -1 & 1 & 0 & -1 \end{array}\right] \rightarrow \left[\begin{array}{rr|rr} 1 & 0 & 3 & 2 \\ 0 & 1 & 2 & 1 \\ 0 & 0 & 1 & 0 \end{array}\right]$$

How should this be interpreted? The second vector, $\boldsymbol{x}_2$ is in H, as it can be expressed as $2\boldsymbol{v}_1 + \boldsymbol{v}_2$. Its low dimensional representation is thus $[2, 1]^T$.

The first vector, $\boldsymbol{x}_1$, cannot be expressed as a linear combination of $\boldsymbol{v}_1$ and $\boldsymbol{v}_2$, so it does not belong to H.

If the basis is orthonormal, we do not need to perform any row reduction. Let us recall a few more definitions:

Definition 0.3. A real $n \times n$ matrix $\mathbb{Q}$ is said to be *orthogonal* if

$$\mathbb{Q}^T\mathbb{Q} = I$$

This is the property that makes an orthonormal basis nice to work with- it's inverse is its transpose. Thus, it is easy to compute the coordinates of a vector $\boldsymbol{x}$ with respect to this basis. That is, suppose that

$$\boldsymbol{x} = c_1\boldsymbol{u}_1 + \ldots + c_k\boldsymbol{u}_k$$

Then the coordinate c_j is just a dot product:

$$\boldsymbol{x} \cdot \boldsymbol{u}_j = 0 + \ldots + 0 + c_j\boldsymbol{u}_j \cdot \boldsymbol{u}_j + 0 + \ldots 0 \quad \Rightarrow \quad c_j = \boldsymbol{x} \cdot \boldsymbol{u}_j$$

We can also interpret each individual coordinate as the projection of $\boldsymbol{x}$ onto the appropriate basis vector. Recall that the orthogonal projection of $\boldsymbol{x}$ onto a vector $\boldsymbol{u}$ is the following:

$$\text{Proj}_{\boldsymbol{u}}(\boldsymbol{x}) = \frac{\boldsymbol{u}_j \cdot \boldsymbol{x}}{\boldsymbol{u} \cdot \boldsymbol{u}}\boldsymbol{u}$$

If $\boldsymbol{u}$ has been normalized (length is 1), then the **coefficient of the projection** could be written as $\boldsymbol{u}_j^T\boldsymbol{x}$. Writing the coefficients in matrix form, with the columns of U being the orthonormal vectors forming the basis, we have:

$$\boldsymbol{c} = U^T\boldsymbol{x}$$

You might reflect on the dimensions of each item in that equation. Additionally, the **projection of $\boldsymbol{x}$** onto the subspace spanned by the (orthonormal) columns of a matrix U is:

$$\text{Proj}_U(\boldsymbol{x}) = U\boldsymbol{c} = UU^T\boldsymbol{x} \tag{1}$$

Example 0.2. We'll change our previous example slightly so that $\boldsymbol{u}_1$ and $\boldsymbol{u}_2$ are orthonormal. Find the coordinates of $\mathbf{x}_1$ with respect to this basis.

$$\boldsymbol{u}_1 = \frac{1}{\sqrt{5}}\begin{bmatrix} 1 \\ 2 \\ 0 \end{bmatrix} \quad \boldsymbol{u}_2 = \frac{1}{\sqrt{6}}\begin{bmatrix} 2 \\ -1 \\ 1 \end{bmatrix} \quad \boldsymbol{x}_1 = \begin{bmatrix} -1 \\ 8 \\ -2 \end{bmatrix}$$

SOLUTION: We'll do this in Matlab- You should be sure you can reproduce the results.

```
>> U=[1 2;2 -1;0 1];
>> s=sqrt(sum(U.*U));
>> U=U./repmat(s,3,1);
%U has orthonormal columns-
>> U'*U
ans =
    1.0000         0
         0    1.0000
>> U*U'
ans =
    0.8667    0.0667    0.3333
    0.0667    0.9667   -0.1667
    0.3333   -0.1667    0.1667
%Now on with the example:
>> x=[-1;8;-2];
>> c=U'*x
c =
    6.7082
   -4.8990
>> U*U'*x
ans =
   -1.0000
    8.0000
   -2.0000
% What if x is not in the subspace?  Here's an example:
>> x=[-1;0;2];
>> U*U'*x
ans =
   -0.2000
   -0.4000
         0
```

We summarize our discussion with the following theorem:

Change of Basis Theorem. Suppose H is a subspace of $\mathbb{R}^n$ with orthonormal basis vectors given by the k columns of a matrix U (so U is $n \times k$). Then, given $\boldsymbol{x} \in H$,

- The **low dimensional representation** of $\boldsymbol{x}$ with respect to U is the vector of coordinates, $\boldsymbol{c} \in \mathbb{R}^k$:

$$\boldsymbol{c} = U^T \boldsymbol{x}$$

- The **reconstruction** of $\boldsymbol{x}$ as a vector in $\mathbb{R}^n$ is:

$$\hat{\boldsymbol{x}} = UU^T \boldsymbol{x}$$

  where, if the subspace formed by U contains $\boldsymbol{x}$, then $\boldsymbol{x} = \hat{\boldsymbol{x}}$- Notice in this case, the projection of $\boldsymbol{x}$ into the column space of U is the same as $\boldsymbol{x}$.

This last point may seem trivial since we started by saying that $\boldsymbol{x} \in U$, however, soon we'll be loosening that requirement.

Projections are important part of our work in modeling data- so much so that we'll spend a bit of time formalizing the ideas in the next section.

Special Mappings: The Projectors

In the previous section, we looked at projecting one vector onto a subspace by using Equation 1. In this section, we think about the projection as a function whose domain and range will be subspaces of $\mathbb{R}^n$.

The defining equation for such a function comes from the idea that if one projects a vector, then projecting it again will leave it unchanged. Therefore, if matrix P performs the projection, then

$$P(Px) = Px$$

Or, more formally, we have the definition:

Definition 0.4. A *projector* is a square matrix P so that:

$$P^2 = P$$

In particular, $P\boldsymbol{x}$ is the projection of $\boldsymbol{x}$.

Notes about the definition:

- The projector is necessarily a square matrix.
- We might call the identity matrix the trivial projector.
- If P is invertible, then $P^2 = P \Rightarrow P = I$. Therefore, to be a nontrivial projector, P must not be invertible.
- The direction of the projection is the vector (this is head minus tail): $P\mathbf{x} - \mathbf{x}$.

Example 0.3. The following are two projectors. Their matrix representations are given by:

$$P_1 = \begin{bmatrix} 1 & 0 \\ 1 & 0 \end{bmatrix} \qquad P_2 = \frac{1}{2}\begin{bmatrix} 1 & 1 \\ 1 & 1 \end{bmatrix}$$

Some samples of the projections are given in Figure 1, where we see that both project to the subspace spanned by $[1, 1]^T$.

Let's consider the action of these matrices on an arbitrary point:

$$P_1\boldsymbol{x} = \begin{bmatrix} 1 & 0 \\ 1 & 0 \end{bmatrix}\begin{bmatrix} x \\ y \end{bmatrix} = \begin{bmatrix} x \\ x \end{bmatrix}, P_1(P_1\boldsymbol{x}) = \begin{bmatrix} 1 & 0 \\ 1 & 0 \end{bmatrix}\begin{bmatrix} x \\ x \end{bmatrix} = \begin{bmatrix} x \\ x \end{bmatrix}$$

$$P_2\boldsymbol{x} = \frac{1}{2}\begin{bmatrix} 1 & 1 \\ 1 & 1 \end{bmatrix}\begin{bmatrix} x \\ y \end{bmatrix} = \begin{bmatrix} \frac{x+y}{2} \\ \frac{x+y}{2} \end{bmatrix} = \frac{x+y}{2}\begin{bmatrix} 1 \\ 1 \end{bmatrix}$$

You should verify that $P_2^2\boldsymbol{x} = P_2(P_2(\boldsymbol{x})) = \boldsymbol{x}$.

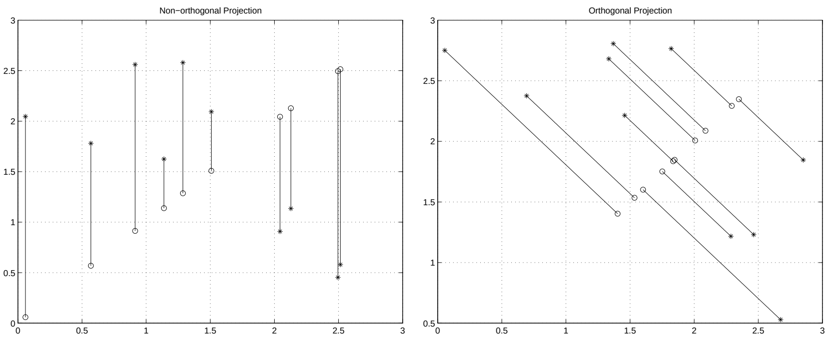

Figure 1: Projections P_1 and P_2 in the first and second graphs (respectively). Asterisks denote the original data point, and circles represent their destination, the projection of the asterisk onto the vector $[1,1]^T$. The line segment follows the direction $P\boldsymbol{x} - \boldsymbol{x}$. Note that P_1 does not project in an orthogonal fashion, while the second matrix P_2 does.

From the previous examples, we see that $\mathbb{P}\boldsymbol{x} - \boldsymbol{x}$ is given by:

$$P_1\boldsymbol{x} - \boldsymbol{x} = \begin{bmatrix} 0 \\ x - y \end{bmatrix}, \text{ and } P_2\boldsymbol{x} - \boldsymbol{x} = \begin{bmatrix} \frac{-x+y}{2} \\ \frac{x-y}{2} \end{bmatrix} = \frac{x-y}{2}\begin{bmatrix} -1 \\ 1 \end{bmatrix}$$

You'll notice that in the case of P_2, $P_2\boldsymbol{x} - \boldsymbol{x} = (P_2 - I)\boldsymbol{x}$ is orthogonal to $P_2\boldsymbol{x}$.

Definition 0.5. $\mathbb{P}$ is said to be an *orthogonal* projector if it is a projector, and the range of $\mathbb{P}$ is orthogonal to the range of $(I - \mathbb{P})$. We can show orthogonality by taking an arbitrary point in the range, $\mathbb{P}\boldsymbol{x}$ and an arbitrary point in $(I - \mathbb{P})$, $(I - \mathbb{P})\boldsymbol{y}$, and show the dot product is 0.

There is a property of real projectors that make them nice to work with: They are also symmetric matrices:

Theorem 0.1. *The (real) projector $\mathbb{P}$ is an orthogonal projector iff $\mathbb{P} = \mathbb{P}^T$.*

Here's a short proof that shows one direction: Suppose P is a projector and P is symmetric. Then, for arbitrary $\mathbf{x}$ and $\mathbf{y}$, we have:

$$(I - P)\boldsymbol{y} \cdot P\boldsymbol{x} = \boldsymbol{y} \cdot P\boldsymbol{x} - P\boldsymbol{y} \cdot P\boldsymbol{x} = \boldsymbol{y}^T P\boldsymbol{x} - \boldsymbol{y}^T P^T P\boldsymbol{x} = \boldsymbol{y}^T P\boldsymbol{x} - \boldsymbol{y}^T P^2\boldsymbol{x}$$

which is zero if $P^2 = P$. To prove the other direction would take us to far afield, so we omit it for now.

Our main use is to either project to a vector or to a subspace:

Projecting to a vector: Let $\boldsymbol{a}$ be an arbitrary, real, non-zero vector. We show that

$$\mathbb{P}_{\boldsymbol{a}} = \frac{\boldsymbol{a}\boldsymbol{a}^T}{\|\boldsymbol{a}\|^2}$$

is a rank one orthogonal projector onto the span of $\boldsymbol{a}$:

- The matrix $\boldsymbol{a}\boldsymbol{a}^T$ has rank one, since every column is a multiple of $\boldsymbol{a}$.
- The given matrix is a projector:

$$\mathbb{P}^2 = \frac{\boldsymbol{a}\boldsymbol{a}^T}{\|\boldsymbol{a}\|^2} \cdot \frac{\boldsymbol{a}\boldsymbol{a}^T}{\|\boldsymbol{a}\|^2} = \frac{1}{\|\boldsymbol{a}\|^4}\boldsymbol{a}(\boldsymbol{a}^T\boldsymbol{a})\boldsymbol{a}^T = \frac{\boldsymbol{a}\boldsymbol{a}^T}{\|\boldsymbol{a}\|^2} = \mathbb{P}$$

- The matrix is an orthogonal projector, since $\mathbb{P}^T = \mathbb{P}$.

Projecting to a Subspace: Let $U = [\boldsymbol{u}_1, \boldsymbol{u}_2, \ldots, \boldsymbol{u}_k]$ be a matrix with orthonormal columns (note that U is typically NOT square). Then

$$\mathbb{P} = UU^T$$

is an orthogonal projector to the column space of Q. See the exercises for a proof.

Exercises

1. Let $\boldsymbol{x} = [3, 2, 3]^T$ and let the basis vectors be $\boldsymbol{u}_1 = \frac{1}{\sqrt{2}}[1, 0, 1]^T$ and let $\boldsymbol{u}_2 = [0, 1, 0]^T$. Compute the low dimensional representation of $\boldsymbol{x}$, and its reconstruction (to verify that $\boldsymbol{x}$ is in the right subspace).

2. Show that the plane H defined by:

$$H = \left\{ \alpha_1 \begin{bmatrix} 1 \\ 1 \\ 1 \end{bmatrix} + \alpha_2 \begin{bmatrix} 1 \\ -1 \\ 0 \end{bmatrix} \text{ such that } \alpha_1, \alpha_2 \in \mathbb{R} \right\}$$

is isormorphic to $\mathbb{R}^2$.

3. Let the subspace G be the plane defined below, and consider the vector $\boldsymbol{x}$, where:

$$G = \left\{ \alpha_1 \begin{bmatrix} 1 \\ 3 \\ -2 \end{bmatrix} + \alpha_2 \begin{bmatrix} 3 \\ -1 \\ 0 \end{bmatrix} \text{ such that } \alpha_1, \alpha_2 \in \mathbb{R} \right\} \qquad \boldsymbol{x} = \begin{bmatrix} 1 \\ 0 \\ 2 \end{bmatrix}$$

   (a) Find the projector P that takes an arbitrary vector and projects it (orthogonally) to the plane G.

   (b) Find the orthogonal projection of the given $\boldsymbol{x}$ onto the plane G.

   (c) Find the distance from the plane G to the vector $\boldsymbol{x}$.

4. If the low dimensional representation of a vector $\boldsymbol{x}$ is $[9, -1]^T$ and the basis vectors are $[1, 0, 1]^T$ and $[3, 1, 1]^T$, then what was the original vector $\boldsymbol{x}$?

5. If the vector $\boldsymbol{x} = [10, 4, 2]^T$ and the basis vectors are $[1, 0, 1]^T$ and $[3, 1, 1]^T$, then what is the low dimensional representation for $\boldsymbol{x}$?

6. Let $\boldsymbol{a} = [-1, 3]^T$. Find a square matrix P so that $P\boldsymbol{x}$ is the orthogonal projection of $\boldsymbol{x}$ onto the span of $\boldsymbol{a}$.

7. Let U be an $n \times k$ matrix with k orthonormal columns. Show that $P = UU^T$ is an orthogonal projector to the column space of U by showing that:

   - P is an orthogonal projector
   - The range is the column space of U.

8. Show that multiplication by an orthogonal matrix preserves lengths: $\|\mathbb{Q}\boldsymbol{x}\|^2 = \|\boldsymbol{x}\|^2$ (Hint: Use properties of inner products and recall that an orthogonal matrix is square.). Conclude that multiplication by $\mathbb{Q}$ represents a rigid rotation.

9. Prove the Pythagorean Theorem by induction: Given a set of n orthogonal vectors $\{\boldsymbol{x}_i\}$

$$\|\sum_{i=1}^{n} \boldsymbol{x}_i\|^2 = \sum_{i=1}^{n} \|\boldsymbol{x}_i\|^2$$

(NOTE: This doesn't have much to do with projections, but is useful later).

The Decomposition Theorems

The Eigenvector/Eigenvalue Decomposition

This section focuses on the decomposition for symmetric matrices. You might also find Section 7.1 of Lay's linear algebra text useful.

From linear algebra, to diagonalize a matrix A means to factor it as:

$$A = PDP^{-1}$$

where D is a diagonal matrix whose diagonal entries are the eigenvalues of A and the matrix P has the corresponding eigenvectors. For example, an $n \times n$ matrix A is diagonalizable if it has n distinct eigenvalues. However, there's a better property we're after- It would be nice if P were actually an orthogonal matrix, because then the inverse is simply the transpose. This extra condition gives us orthogonal diagonalization:

Definition: A matrix A is *orthogonally diagonalizable* if there is an orthogonal matrix V and diagonal matrix Λ so that so that $A = V\Lambda V^T$. We will see that in the case that A is symmetric, the matrix V comes from the eigenvectors of A and the diagonal values in Λ are the eigenvalues.

We ran into problems in Linear Algebra because "most" matrices are not diagonalizable- It seemed like we could only look at special cases. The next theorem solves this problem by telling us that if we have a symmetric matrix, then the factorization is quite beautiful:

The Spectral Theorem: If A is an $n \times n$ symmetric matrix, then:

1. A has n real eigenvalues (counting multiplicity).

2. For all λ, the algebraic and geometric multipliciilties are equal: $a_\lambda = g_\lambda$[1].

3. The eigenspaces are mutually orthogonal.

4. A is orthogonally diagonalizeable, with $D = \text{diag}(\lambda_1, \lambda_2, \ldots, \lambda_n)$.

Some remarks about the Spectral Theorem:

- We assume that inside each eigenspace, we have an orthonormal basis of eigenvectors. This is not a restriction, since we can always construct such a basis using Gram-Schmidt.

[1]This means that if λ_i is a root of multiplicity k for the characteristic equation, we will find k eigenvectors.

- If a matrix is real and symmetric, the Spectral Theorem says that its eigenvectors form an orthonormal basis for $\mathbb{R}^n$.
- The first part is somewhat difficult to prove in that we would have to bring in more machinery than we would like. If you would like to see a proof, it comes from the *Schur Decomposition*, which is given, for example, in "Matrix Computations" by Golub and Van Loan.

The following is a proof of the orthogonal eigenspaces. Supply justification for each step: Let $\boldsymbol{v}_1$, $\boldsymbol{v}_2$ be eigenvectors from distinct eigenvalues, λ_1, λ_2. We show that $\boldsymbol{v}_1 \cdot \boldsymbol{v}_2 = 0$:

$$\lambda_1 \boldsymbol{v}_1 \cdot \boldsymbol{v}_2 = (A\boldsymbol{v}_1)^T \boldsymbol{v}_2 = \boldsymbol{v}_1^T A^T \boldsymbol{v}_2 = \boldsymbol{v}_1^T A \boldsymbol{v}_2 = \lambda_2 \boldsymbol{v}_1 \cdot \boldsymbol{v}_2$$

Now, $(\lambda_1 - \lambda_2)\boldsymbol{v}_1 \cdot \boldsymbol{v}_2 = 0$.

Instead of the formula $A = PDP^{-1}$, we get something a little stronger with orthonormal eigenvectors- **The Spectral Decomposition**:

$$A = (\boldsymbol{q}_1 \ \boldsymbol{q}_2 \ \ldots \ \boldsymbol{q}_n) \begin{pmatrix} \lambda_1 & 0 & \ldots & 0 \\ 0 & \lambda_2 & \ldots & 0 \\ \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & \ldots & \lambda_n \end{pmatrix} \begin{pmatrix} \boldsymbol{q}_1^T \\ \boldsymbol{q}_2^T \\ \vdots \\ \boldsymbol{q}_n^T \end{pmatrix} = (\lambda_1 \boldsymbol{q}_1 \ \lambda_2 \boldsymbol{q}_2 \ \ldots \ \lambda_n \boldsymbol{q}_n) \begin{pmatrix} \boldsymbol{q}_1^T \\ \boldsymbol{q}_2^T \\ \vdots \\ \boldsymbol{q}_n^T \end{pmatrix}$$

so finally:

$$A = \lambda_1 \boldsymbol{q}_1 \boldsymbol{q}_1^T + \lambda_2 \boldsymbol{q}_2 \boldsymbol{q}_2^T + \ldots + \lambda_n \boldsymbol{q}_n \boldsymbol{q}_n^T$$

That is, A is a sum of n rank one matrices, each of which is a projection matrix.

Here's a numerical example using a symmetric matrix:

$$A = \begin{bmatrix} 3 & 1 \\ 1 & 3 \end{bmatrix}$$

The eigenvalues are found by using the characteristic equation, $(3 - \lambda)^2 - 1 = 0$, so that $\lambda = 2, 4$. Using $\lambda = 2$, we find a basis for the eigenspace:

$$(A - \lambda I)\mathbf{v} = \mathbf{0} \quad \Rightarrow \quad \begin{bmatrix} 1 & 1 \\ 1 & 1 \end{bmatrix} \mathbf{v} = \mathbf{0} \quad \Rightarrow \quad \mathbf{v} = \frac{1}{\sqrt{2}} \begin{bmatrix} 1 \\ -1 \end{bmatrix}$$

Similarly, for $\lambda = 4$,

$$(A - \lambda I)\mathbf{v} = \mathbf{0} \quad \Rightarrow \quad \begin{bmatrix} -1 & 1 \\ -1 & 1 \end{bmatrix} \mathbf{v} = \mathbf{0} \quad \Rightarrow \quad \mathbf{v} = \frac{1}{\sqrt{2}} \begin{bmatrix} 1 \\ 1 \end{bmatrix}$$

The eigenvector decomposition is (I've factored out the constants)

$$A = V \Lambda V^T = \frac{1}{2} \begin{bmatrix} 1 & 1 \\ -1 & 1 \end{bmatrix} \begin{bmatrix} 2 & 0 \\ 0 & 4 \end{bmatrix} \begin{bmatrix} 1 & -1 \\ 1 & 1 \end{bmatrix}$$

And the spectral decomposition gives a sum of rank one matrices:

$$A = 2 \cdot \frac{1}{2} \begin{bmatrix} 1 & -1 \\ -1 & 1 \end{bmatrix} + 4 \cdot \frac{1}{2} \begin{bmatrix} 1 & 1 \\ 1 & 1 \end{bmatrix}$$

Exercises

1. Show that, if λ_i is an eigenvalue of A^TA, then $\lambda_i \geq 0$ for $i = 1..n$ by showing that $\|A\boldsymbol{v}_i\|^2 = \lambda_i$.

2. (Matlab) Verify the spectral decomposition for a symmetric matrix. Type the following into Matlab :

```
%Construct a random, symmetric, 6 x 6 matrix:
for i=1:6
  for j=1:i
    A(i,j)=rand;
    A(j,i)=A(i,j);
  end
end

%Compute the eigenvalues of A:
[Q,L]=eig(A);   %NOTE:  A = Q L Q'
                %L is a diagonal matrix

%Now form the spectral sum
S=zeros(6,6); for i=1:6
  S=S+L(i,i)*Q(:,i)*Q(:,i)';
end

max(max(S-A))
```

   Note that the maximum of $S - A$ should be a very small number! (By the spectral decomposition theorem).

3. Prove that, if $\boldsymbol{v}_i$ and $\boldsymbol{v}_j$ are distinct eigenvectors of A^TA, then their corresponding images, $A\boldsymbol{v}_i$ and $A\boldsymbol{v}_j$, are orthogonal.

4. Prove that, if $\boldsymbol{x} = \alpha_1\boldsymbol{v}_1 + \ldots \alpha_n\boldsymbol{v}_n$, then

$$\|A\boldsymbol{x}\|^2 = \alpha_1^2\lambda_1 + \ldots + \alpha_n^2\lambda_n$$

5. Let W be the subspace generated by the basis $\{\boldsymbol{v}_j\}_{j=k+1}^n$, where $\boldsymbol{v}_j$ are the eigenvectors associated with the *zero* eigenvalues of A^TA (therefore, we are assuming that the first k eigenvalues are NOT zero). Show that $W = \text{Null}(A)$.

6. Prove that if the rank of A^TA is r, then so is the rank of A.

7. Prove that if λ_i is an eigenvalue of A^TA, then λ_i is also an eigenvalue of AA^T (Hint: Let $\mathbf{u}_i = A\mathbf{v}_i$, where $\mathbf{v}_i$ is an eigenvector associated with λ_i).

A New Factorization

There is a special matrix factorization that is extremely useful, both in applications and in proving theorems. This is mainly due to two facts, which we shall investigate in this section: (1) We can use this factorization on *any* matrix, (2) The factorization defines explicitly the rank of the matrix, and all four matrix subspaces.

In what follows, assume that A is an $m \times n$ matrix. Although A itself is not symmetric, but A^TA and AA^T are symmetric. Therefore, they are orthogonally diagonalizable.

Let $\{\lambda_i\}_{i=1}^n$ and $V = [\boldsymbol{v}_1, \boldsymbol{v}_2, \ldots, \boldsymbol{v}_n]$ be the eigenvalues and orthonormal eigenvectors for A^TA.

In the rest of the section, we will assume any list (or diagonal matrix) of eigenvalues of A^TA (or singular values of A) will be ordered from highest to lowest: $\lambda_1 \geq \lambda_2 \geq \ldots \geq \lambda_n$.

Definition: We define the singular values of A by:

$$\sigma_i = \sqrt{\lambda_i}$$

where λ_i is an eigenvalue of A^TA (we will show that these are the same as the eigenvalues of AA^T, so it won't matter which you use).

Define

$$\boldsymbol{u}_i = \frac{1}{\|A\boldsymbol{v}_i\|}A\boldsymbol{v}_i = \frac{1}{\sigma_i}A\boldsymbol{v}_i$$

and let U be the matrix whose i^{th} column is $\boldsymbol{u}_i$.

This definition only makes sense sense for the first r vectors $\boldsymbol{v}$ (otherwise, $A\boldsymbol{v}_i = \mathbf{0}$). Thus, we'll have to extend the basis to span all of $\mathbb{R}^m$ (How would we do that in practice?)

The vector $\boldsymbol{u}_i$ is an eigenvector of AA^T whose eigenvalue is also λ_i (Show this).

And for future reference, we note that if we know either $\mathbf{v}_i$ or $\mathbf{u}_i$, we can find the other using the relationships:

$$A^T\boldsymbol{u}_i = \sigma_i\boldsymbol{v}_i \qquad\qquad \mathbf{u}_i = \frac{1}{\sigma_i}A\mathbf{v}_i$$

Now, from the Spectral Theorem, we know that for the matrix AA^T, we can find a full set of eigenvalues λ_i together with a full set of orthonormal eigenvectors, $\mathbf{u}_i$ (or equivalently, a full set of eigenvectors $\mathbf{v}_i$, for A^TA). We want to put these together now.

Let A be $m \times n$. Define the $m \times n$ matrix Σ to be a diagonal (but not necessarily square) matrix with the singular values along the diagonal:

$$\Sigma = \text{diag}(\sigma_1, \ldots, \sigma_n)$$

where σ_i is the i^{th} singular value of the matrix A and zeros fill the rest of the matrix. Using the relationships we've shown in the exercises, we see that:

$$AV = U\Sigma$$

The Singular Value Decomposition (SVD)

Let A be any $m \times n$ matrix of rank r. Then

$$A = U\Sigma V^T$$

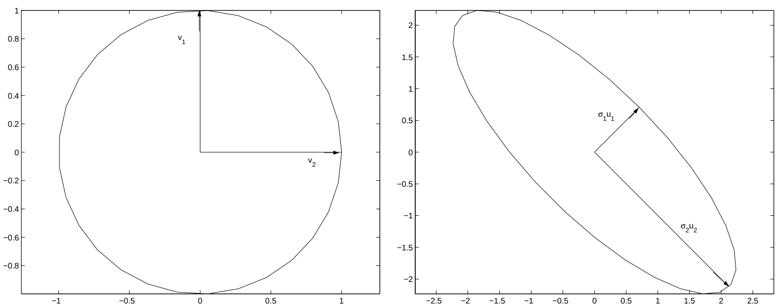

Figure 2: The geometric meaning of the right and left singular vectors of the SVD decomposition. Note that $A\boldsymbol{v}_i = \sigma_i \boldsymbol{u}_i$. The mapping $x \to Ax$ will map the unit circle on the left to the ellipse on the right.

where U, Σ, V are the matrices defined in the previous exercises. That is, U is an orthogonal $m \times m$ matrix, Σ is a diagonal $m \times n$ matrix, and V is an orthogonal $n \times n$ matrix. The $\boldsymbol{u}$'s are called the *left singular vectors* and the $\boldsymbol{v}$'s are called the *right singular vectors*.

The usefulness of the SVD comes in its ability to explicitly describe the four fundamental subspaces to a matrix A: Let $A = U\Sigma V^T$ be the SVD of A with rank r. Be sure that the singular values are ordered from highest to lowest. Then:

1. The column space of A is spanned by the first r columns of U.
2. The row space of A is spanned by the first r columns of V.
3. The null space of A is spanned by the remaining columns of V.
4. The null space of A^T is spanned by the remaining columns of U.

We can visualize the right and left singular values as in Figure 2. We think of the $\boldsymbol{v}_i$ as a special orthogonal basis in R^n (Domain) that maps to the ellipse whose axes are defined by $\sigma_i \boldsymbol{u}_i$.

The SVD provides a decomposition of any linear mapping into two "rotations" and a scaling. This will become important later when we try to deduce a mapping matrix from data, but it also gives some intuition about the SVD.

The Reduced SVD

If m or n is very large, it might not make sense to keep the full matrix U and V. In what is called the **reduced SVD**, we throw out the vectors corresponding to the null spaces:

The Reduced SVD Let A be $m \times n$ with rank r. Then we can write:

$$A = \tilde{U}\tilde{\Sigma}\tilde{V}^T$$

where $\tilde{U}$ is an $m \times r$ matrix with orthogonal columns, $\tilde{\Sigma}$ is an $r \times r$ *square* matrix, and $\tilde{V}$ is an $n \times r$ matrix.

The SVD Decomposition Theorem: (Actually, this is just another way to express the SVD). Let $A = U\Sigma V^T$ be the SVD of A, which has rank r. Then:

$$A = \sum_{i=1}^{r} \sigma_i \boldsymbol{u}_i \boldsymbol{v}_i^T$$

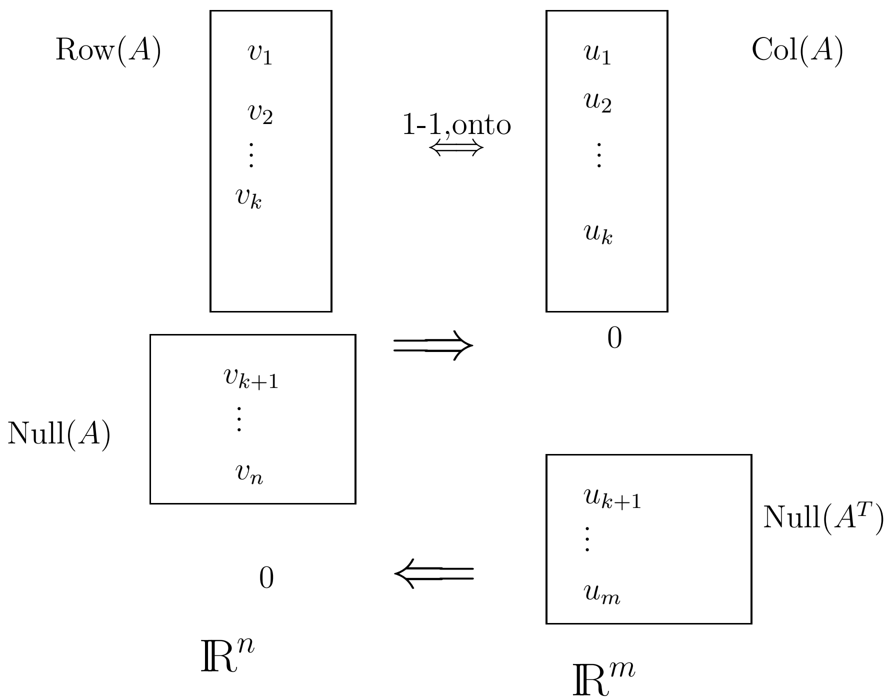

Figure 3: The SVD of A (`[U,S,V]=svd(A)`) completely and explicitly describes the 4 fundamental subspaces associated with the matrix, as shown. We have a one to one correspondence between the rowspace and columnspace of A, the remaining $\boldsymbol{v}$'s map to zero, and the remaining $\boldsymbol{u}$'s map to zero (under A^T).

Therefore, we can approximate A by the sum of rank one matrices. Compare this to the Spectral Decomposition Theorem we had previously.

Matlab has the SVD built in. The function specifications are: `[U,S,V]=svd(A)` and `[U,S,V]=svd(A,'econ')` where the first function call returns the full SVD, and the second call returns the "economy-sized" SVD (which is a bit different than the reduced SVD)- see Matlab's help file for the details on the second call.

Here is a Matlab example with a matrix A given below:

$$A = \begin{bmatrix} 1 & 2 & 3 \\ 4 & 5 & 6 \end{bmatrix}$$

First, we perform the reduced SVD. I can tell by the columns of A that the matrix has rank 2.

```
>> [U,S,V]=svd(A,'econ')
U =
   -0.3863   -0.9224
   -0.9224    0.3863
S =
    9.5080         0
         0    0.7729
V =
   -0.4287    0.8060
   -0.5663    0.1124
   -0.7039   -0.5812
```

Now, to get a partial reconstruction I can try using just one vector:

$$A \approx \sigma_1 \boldsymbol{u}_1 \boldsymbol{v}_1^T$$

In Matlab, the command would be the following, which looks like it is somewhat close to the matrix A:

```
>> S(1,1)*U(:,1)*V(:,1)'
ans =
    1.5745    2.0801    2.5857
    3.7594    4.9664    6.1735
```

And if we use both basis vectors in U and V, we should get "perfect" reconstruction (may be numerical round off error). In notation, we are computing

$$A = \sigma_1 \boldsymbol{u}_1 \boldsymbol{v}_1^T + \sigma_2 \boldsymbol{u}_2 \boldsymbol{v}_2^T$$

In Matlab, we get

```
>> S(1,1)*U(:,1)*V(:,1)'+S(2,2)*U(:,2)*V(:,2)'
ans =
    1.0000    2.0000    3.0000
    4.0000    5.0000    6.0000
```

Exercises

1. Compute the SVD by hand of the following matrices:

$$\begin{pmatrix} 1 & 1 \\ 0 & 0 \end{pmatrix} \qquad \begin{pmatrix} 0 & 2 \\ 0 & 0 \\ 0 & 0 \end{pmatrix}$$

2. (Matlab) The SVD can be used in image processing- Here is an example using the image of a mandrill (it comes with Matlab).

```
clear
close all;  %This closes all open figures

load mandrill
image(X)
colormap(map)
[U,S,V]=svd(X);
NumVecs=[3,6,10,50,150,450];
for k=1:6
    j=NumVecs(k);
    H=U(:,1:j)*S(1:j,1:j)*V(:,1:j)';
    figure(1)
    subplot(2,3,k)
    image(H)
    figure(2)
    subplot(2,3,k)
    imagesc(H-X);
end
```

   Programming Note: There are three new things here- One is the SVD and its decomposition. Another is the use of the **subplot** command- It is very useful when you want to make a lot of plots! Finally, we can also move back and forth between several figures with the **figure** command.

3. Revisiting the Iris Data: Go onto our class website and download the iris data again. Recall that the data is arranged as a 150×4 matrix, where there are 3 classes of iris- they are currently arranged in order.

   Here is the problem: Think of the data as 150 points in $\mathbb{R}^4$. Use the SVD of the mean-subtracted data (as 150 vectors in $\mathbb{R}^4$) as basis vectors (the eigenvectors are orthonormal, so they make a great basis!).

   Construct the basis vectors and construct the two dimensional representation of the data (Recall our discussion of the low dimensional representation?).

   Before you set up the graph, you might recall that our classifier (from Homework 7) did a great job with Class 1, but got a little mixed up between classes 2 and 3. For your convenience, you might recall a confusion matrix like:

$$\begin{bmatrix} 1.0000 & 0 & 0 \\ 0 & 0.82 & 0.18 \\ 0 & 0.20 & 0.80 \end{bmatrix}$$

Here is some code to get you started:

```
load IrisData
Xm=X-repmat(mean(X),150,1);

(Do the reduced SVD on Xm to get basis vectors in R^4)


(If the low dimensional data is in a 2x150 array called Coords, then:)
plot(Coords(1,1:50),Coords(2,1:50),'r.');
hold on
plot(Coords(1,51:100),Coords(2,51:100),'b.');
plot(Coords(1,101:150),Coords(2,101:150),'m.');
hold off
```

Generalized Inverses

Let a matrix A be $m \times n$ with rank r. Since this mapping is not invertible as a mapping from $\mathbb{R}^n$ to $\mathbb{R}^m$, is there a way of *restricting* the mapping to an $r-$dimensional subspace of $\mathbb{R}^n$ so that the restricted map *is* invertible?

An obvious choice might be to restrict the mapping to the row space, which has dimension r. We already know that the image of the row space is the column space which also has dimension r. Therefore, there exists a 1-1 and onto map between these spaces. Now, how do we compute the "inverse" of the mapping given our restrictions?

We proceed first by assuming that $\mathbf{y}$ is not in the column space of A. Then we know that there is no solution to $A\mathbf{x} = \mathbf{y}$, but there is a least squares solution by projecting $\mathbf{y}$ into the column space of A, and define this vector as $\hat{\mathbf{y}}$.

To get a unique solution, we replace $\boldsymbol{x}$ by its projection to the row space of A, $\hat{\boldsymbol{x}}$. The problem

$$\hat{\boldsymbol{y}} = A\hat{\boldsymbol{x}}$$

now has a unique solution which can be expressed using the (reduced) SVD of A.

$$\hat{\boldsymbol{x}} = VV^T\boldsymbol{x}, \qquad \hat{\boldsymbol{y}} = UU^T\boldsymbol{y}$$

Now we can write:

$$UU^T\boldsymbol{y} = U\Sigma V^T\left(VV^T\boldsymbol{x}\right)$$

so that

$$V\Sigma^{-1}U^T\boldsymbol{y} = VV^T\boldsymbol{x}$$

(Exercise: Verify that these computations are correct!)

Given an $m \times n$ matrix A, define its pseudo-inverse, $A^\dagger$ by:

$$A^\dagger = V\Sigma^{-1}U^T$$

We have shown that the least squares solution to $\boldsymbol{y} = A\boldsymbol{x}$ is given by:

$$\hat{\boldsymbol{x}} = A^\dagger \boldsymbol{y}$$

where $\hat{\boldsymbol{x}}$ is in the row space of A, and its image, $A\hat{\boldsymbol{x}}$ is the projection of $\boldsymbol{y}$ into the column space of A.

Exercises

1. Recall finding the slope and intercept for the line of best fit. Instead of using the "slash" command, we can now use the SVD. The matrix A comes from the data- For example, given the following data:

$$\begin{array}{c|cccc} x & -1 & 0 & 1 & 2 \\ \hline y & 1 & 0 & 3 & 4 \end{array} \quad \Rightarrow \quad \begin{bmatrix} -1 & 1 \\ 0 & 1 \\ 1 & 1 \\ 2 & 1 \end{bmatrix} \begin{bmatrix} m \\ b \end{bmatrix} = \begin{bmatrix} 1 \\ 0 \\ 3 \\ 4 \end{bmatrix} \quad \Rightarrow \quad A\mathbf{c} = \mathbf{y}$$

   In Matlab, write a script file that solves for m, b using the SVD of A, then plot the data together with the line (this is the line of best fit).

2. We had to assume something when we tried to use the *normal equations* to solve a system, $A\mathbf{x} = \mathbf{b}$. What was it? Does the pseudo-inverse get us around that issue?

3. Consider

$$\begin{bmatrix} 2 & 1 & -1 & 3 \\ -1 & 0 & 1 & -2 \\ 7 & 2 & -5 & 12 \\ -3 & -2 & 0 & -4 \\ 4 & 1 & -3 & 7 \end{bmatrix} \begin{bmatrix} x_1 \\ x_2 \\ x_3 \\ x4 \end{bmatrix} = \begin{bmatrix} 5 \\ 1 \\ 0 \\ -2 \\ 6 \end{bmatrix}$$

   (a) Find the dimensions of the four fundamental subspaces by using the SVD of A (in Matlab).

   (b) Solve the problem.

   (c) Check your answer explicitly and verify that $\hat{\boldsymbol{x}}$ and $\hat{\boldsymbol{y}}$ are in the rowspace and columnspace.