

MATH 235: Inner Product Spaces, Assignment 7

Hand in questions 3,4,5,6,9,10 by 9:30 am on Wednesday March 26, 2008.

Contents

1	Orthogonal Basis for Inner Product Space	2
2	Inner-Product Function Space	2
3	Weighted Inner Product in \mathbb{R}^2 ***	2
4	Inner Product in S^3 ***	2
5	Inner Products in \mathcal{M}_{22} and \mathbb{P}_2 ***	3
6	Cauchy-Schwarz and Triangle Inequalities ***	3
7	Length in Function Space	3
8	More on Function Space	3
9	Linear Transformations and Inner Products ***	4
10	MATLAB ***	5

1 Orthogonal Basis for Inner Product Space

If $V = \mathbf{P}_3$ with the inner product $\langle f, g \rangle = \int_{-1}^1 f(x)g(x)dx$, apply the Gram-Schmidt algorithm to obtain an orthogonal basis from $B = \{1, x, x^2, x^3\}$.

2 Inner-Product Function Space

Consider the vector space $C[0, 1]$ of all continuously differentiable functions defined on the closed interval $[0, 1]$. The inner product in $C[0, 1]$ is defined by $\langle f, g \rangle = \int_0^1 f(x)g(x)dx$. Find the inner products of the following pairs of functions and state whether they are orthogonal

1. $f(x) = \cos(2\pi x)$ and $g(x) = \sin(2\pi x)$
2. $f(x) = x$ and $g(x) = e^x$
3. $f(x) = x$ and $g(x) = 3x$

3 Weighted Inner Product in \mathbb{R}^2 ***

Show whether or not the following are valid inner products in \mathbb{R}^2 . If one is a valid inner product, then find a nonzero vector that is orthogonal to the vector $y = \begin{pmatrix} 2 \\ 1 \end{pmatrix}^T$.

1. $\langle u, v \rangle := 7u_1v_1 + 1.2u_2v_2$.
2. $\langle u, v \rangle := -7u_1v_1 - 1.2u_2v_2$.
3. $\langle u, v \rangle := 7u_1v_1 - 1.2u_2v_2$.

4 Inner Product in \mathcal{S}^3 ***

Let \mathcal{S}^3 denote the vector space of 3×3 symmetric matrices.

1. Prove that $\langle S, T \rangle = \text{tr} ST$ is an inner product on \mathcal{S}^3 .
2. Let $S_1 = \begin{pmatrix} 1 & 0 & 1 \\ 0 & 1 & 0 \\ 1 & 0 & 0 \end{pmatrix}$ and $S_2 = \begin{pmatrix} 1 & 0 & 1 \\ 0 & -3 & 0 \\ 1 & 0 & 0 \end{pmatrix}$. Apply the Gram-Schmidt process in \mathcal{S}^3 to find a matrix orthogonal to both S_1 and S_2 .
3. Find an orthonormal basis for \mathcal{S}^3 using the above three matrices.
4. Let $T = \begin{pmatrix} 1 & 0 & 1 \\ 0 & 1 & -1 \\ 1 & -1 & 1 \end{pmatrix}$. Find the projection of T on the span of $\{S_1, S_2\}$.

5 Inner Products in \mathcal{M}_{22} and \mathbb{P}_2 ***

Show which of the following are valid inner products.

1. Let $A = \begin{bmatrix} a_1 & a_2 \\ a_3 & a_4 \end{bmatrix}$ and $B = \begin{bmatrix} b_1 & b_2 \\ b_3 & b_4 \end{bmatrix}$ be real matrices.

(a) Define $\langle A, B \rangle := a_1b_1 + 2a_2b_3 + 3a_3b_2 + a_4b_4$.

(b) Define $\langle A, B \rangle := \text{tr } B^T A$.

2. Let $p(x)$ and $q(x)$ be polynomials in \mathbb{P}_2 .

(a) Define $\langle p, q \rangle := p(-1)q(-1) + p(\frac{1}{2})q(\frac{1}{2}) + p(-3)q(-3)$.

(b) Define $\langle p, q \rangle := p(-3)q(-3) + p(\frac{1}{2})q(\frac{1}{2}) + p(-1)q(-1)$.

6 Cauchy-Schwarz and Triangle Inequalities ***

1. Let $D = \begin{pmatrix} 3 & 0 & 0 \\ 0 & \pi & 0 \\ 0 & 0 & 2 \end{pmatrix}$. Consider the inner product defined on \mathbb{R}^3 by: $\langle u, v \rangle = u^T D v$. Let

$u = \begin{pmatrix} -2 \\ 3 \\ -7 \end{pmatrix}$ and $v = \begin{pmatrix} -4 \\ 8 \\ 9 \end{pmatrix}$. Verify that both inequalities hold.

2. Find the projection of v on the span of u and call it w . Verify that equality holds for the Cauchy-Schwarz inequality applied to u, w .

7 Length in Function Space

Let $W = \text{Span}\{1, \cos(x), \cos(2x)\} \in C[-\pi, \pi]$ with inner product $\langle f, g \rangle = \int_{-\pi}^{\pi} f(x)g(x)dx$. It is known that $\{\frac{1}{\sqrt{2\pi}}, \frac{1}{\sqrt{\pi}} \cos(x), \frac{1}{\sqrt{\pi}} \cos(2x)\}$ is an orthonormal basis for W . A function $f \in W$ satisfies $\int_{-\pi}^{\pi} f(x)dx = a$, $\int_{-\pi}^{\pi} f(x) \cos(x)dx = b$ and $\int_{-\pi}^{\pi} f(x) \cos(2x)dx = c$. Find an expression for $\|f\|$ in terms of a, b and c .

8 More on Function Space

Consider the vector space P_2 consisting of polynomials of degree at most 2 together with the inner product

$$\langle f, g \rangle = \int_0^1 f(x)g(x)dx, \quad f, g \in P_2.$$

Let W be the subspace of P_2 having a basis $\{1, x\}$.

1. Find an orthonormal basis for W .

2. Let $h(x) = x^2 + 1$. Find the vector in W that is closest to $h(x)$.

9 Linear Transformations and Inner Products ***

Let V be a real inner product space (with the usual notation) and let $\{\mathbf{v}_1, \mathbf{v}_2, \dots, \mathbf{v}_n\}$ be an orthonormal basis for V (so V is finite dimensional). Fix a linear transformation $T : V \rightarrow \mathbf{R}$. Define $\mathbf{v}_0 = T(\mathbf{v}_1)\mathbf{v}_1 + T(\mathbf{v}_2)\mathbf{v}_2 + \dots + T(\mathbf{v}_n)\mathbf{v}_n$.

1. Show that for all $\mathbf{v} \in V$, $T(\mathbf{v}) = \langle \mathbf{v}_0, \mathbf{v} \rangle$.
2. Prove that, for vectors $\mathbf{u}_0, \mathbf{w}_0 \in V$, if $\langle \mathbf{u}_0, \mathbf{v} \rangle = \langle \mathbf{w}_0, \mathbf{v} \rangle$ for all $\mathbf{v} \in V$, then $\mathbf{u}_0 = \mathbf{w}_0$.
3. Deduce from (b) that \mathbf{v}_0 is the *only* vector in V with the displayed property in part (a).

NOTE: This question shows how a suitable inner product produces a 1 – 1 correspondence between vectors in the vector space and linear *functionals* T from it to the reals. This is used to identify tangent vectors with what are called cotangent vectors in differential geometry, general relativity and in other symbol manipulations of mathematical physics, where it is sometimes referred to as “moving indices up-and-down”. A slightly surprising consequence of part (c) is that the vector \mathbf{v}_0 , despite appearances, depends only on the inner product and T , but not on the choice of orthonormal basis.

10 MATLAB ***

Gram-Schmidt & QR Factorization

Overview of the Gram-Schmidt Process

Given a basis $\{x_1, \dots, x_p\}$ for a subspace W of \mathbb{R}^n , define

$$\begin{aligned}v_1 &= x_1 \\v_2 &= x_2 - \frac{x_2 \cdot v_1}{v_1 \cdot v_1} v_1 \\v_3 &= x_3 - \frac{x_3 \cdot v_1}{v_1 \cdot v_1} v_1 - \frac{x_3 \cdot v_2}{v_2 \cdot v_2} v_2 \\&\vdots \\v_p &= x_p - \frac{x_p \cdot v_1}{v_1 \cdot v_1} v_1 - \frac{x_p \cdot v_2}{v_2 \cdot v_2} v_2 - \dots - \frac{x_p \cdot v_{p-1}}{v_{p-1} \cdot v_{p-1}} v_{p-1}\end{aligned}$$

Then $\{v_1, \dots, v_p\}$ is an *orthogonal basis* for W .

When working on a computer, a basis is often presented as a matrix A with linearly independent columns. The columns of matrix A are a basis for its column space (or range). The Gram-Schmidt process can be applied to the column vectors of matrix A so that the output of the process is an orthogonal set of vectors.

For example, let

$$x_1 = \begin{bmatrix} 1 \\ 1 \\ 1 \\ 1 \end{bmatrix}, x_2 = \begin{bmatrix} 0 \\ 1 \\ 1 \\ 1 \end{bmatrix}, x_3 = \begin{bmatrix} 0 \\ 0 \\ 1 \\ 1 \end{bmatrix}$$

Then, by inspection, $\{x_1, x_2, x_3\}$ are linearly independent and thus form a basis for the subspace W of \mathbb{R}^4 . Consider the following matrix A formed by using the basis $\{x_1, x_2, x_3\}$ as column vectors:

$$A = \begin{bmatrix} 1 & 0 & 0 \\ 1 & 1 & 0 \\ 1 & 1 & 1 \\ 1 & 1 & 1 \end{bmatrix}$$

Matrix A will be used to illustrate the Gram-Schmidt process in MATLAB.

First, enter matrix A .

```
>> A = [1 0 0; 1 1 0; 1 1 1; 1 1 1]      %Create matrix A
```

Next, calculate the orthogonal basis $\{v_1, v_2, v_3\}$ by using the Gram-Schmidt algorithm stated above. Recall that in MATLAB, the notation $A(:, 1)$ selects the first *column* of A , $A(:, 2)$ selects the second column, and so on... Also recall that v_1' is the short form for $\text{transpose}(A)$ (changes a row vector to a column vector, or vice versa).

```
>> v1 = A(:,1)    %Calculate v1
```

```
>> v2 = A(:,2) - (A(:,2)'*v1)/(v1'*v1)*v1    %Calculate v2
```

```
>> v3 = A(:,3) - (A(:,3)'*v1)/(v1'*v1)*v1 - (A(:,3)'*v2)/(v2'*v2)*v2
```

```
%Calculate v3
```

MATLAB tells us that by using the Gram-Schmidt process on matrix A , the corresponding *orthogonal* basis is

$$v_1 = \begin{bmatrix} 1 \\ 1 \\ 1 \\ 1 \end{bmatrix}, v_2 = \begin{bmatrix} -0.75 \\ 0.25 \\ 0.25 \\ 0.25 \end{bmatrix}, v_3 = \begin{bmatrix} 0 \\ -0.6667 \\ 0.3333 \\ 0.3333 \end{bmatrix}$$

To convert from an Orthogonal basis to an Orthonormal Basis:

We must simply divide each orthogonal vector by its norm:

```
>> vn1 = v1/norm(v1)    %Calculate normalized v1
>> vn2 = v2/norm(v2)    %Calculate normalized v2
>> vn3 = v3/norm(v3)    %Calculate normalized v3
```

The following theorem provides a method for checking whether the columns of a matrix are *orthonormal*:

An $m \times n$ matrix U has orthonormal columns if and only if

$$U^T U = I$$

Using this theorem and MATLAB, we can check whether $\{vn1, vn2, vn3\}$ forms an orthonormal basis. First define a new matrix U that uses $\{vn1, vn2, vn3\}$ as its column vectors.

```
>> U = [vn1 vn2 vn3]    %Create matrix U
>> transpose(U) * U     %Apply theorem
```

Indeed, MATLAB returns the identity matrix I , so by the theorem, $\{vn1, vn2, vn3\}$ form an orthonormal basis.

QR Factorization

QR factorization is the *matrix version* of the Gram-Schmidt method. QR factorization is particularly useful because computer algorithms exist that implement this factorization method with good control over computer round-off error. Additionally, there is no simple method for computing eigenvalues of matrices that are larger than 3×3 since numerically it is difficult to calculate the roots of the characteristic polynomial. Using QR factorization iteratively is the most successful algorithm that exists today for computing eigenvalues. For these reasons, QR is an important factorization technique in Linear Algebra.

Overview of QR Factorization

If A is an $m \times n$ matrix with linearly independent columns, then A can be factored as

$$A = QR$$

where Q is an $m \times n$ matrix with orthonormal columns consisting of an orthonormal basis of the column space, and R is an invertible, upper triangular matrix.

To illustrate this method in MATLAB, consider the following matrix A :

$$A = \begin{bmatrix} 1 & 0 & 0 \\ 1 & 1 & 0 \\ 1 & 1 & 1 \\ 1 & 1 & 1 \end{bmatrix}$$

This is the same matrix that was used in the section on the Gram-Schmidt process. We have already used MATLAB to calculate the orthonormal basis for matrix A as:

$$vn1 = \begin{bmatrix} 0.5000 \\ 0.5000 \\ 0.5000 \\ 0.5000 \end{bmatrix}, vn2 = \begin{bmatrix} -0.8660 \\ 0.2887 \\ 0.2887 \\ 0.2887 \end{bmatrix}, vn3 = \begin{bmatrix} 0 \\ -0.8165 \\ 0.4082 \\ 0.4082 \end{bmatrix}$$

As such, according to the description of the QR factorization method, it must be that we find:

$$Q = \begin{bmatrix} 0.5000 & -0.8660 & 0 \\ 0.5000 & 0.2887 & -0.8165 \\ 0.5000 & 0.2887 & 0.4082 \\ 0.5000 & 0.2887 & 0.4082 \end{bmatrix}$$

Let's use MATLAB to verify this result. MATLAB's command for QR Factorization is

$$[Q \ R] = \text{qr}(A)$$

where A is an $m \times n$ matrix, Q is returned as an $m \times m$ orthogonal matrix and R is returned as an $m \times n$ upper triangular matrix.

This MATLAB command for QR factorization is very *general* since it applies to *any* matrix and it returns more information than required in some cases.

If A has rank n , then only the first n columns of Q will be an orthonormal basis for the column space of A .

Enter the following in MATLAB:

```
>> A = [1 0 0; 1 1 0; 1 1 1; 1 1 1]      %Create matrix A
>> rank(A)      %Compute rank of matrix A
```

Since A is a 4×3 matrix and its rank is 3, we should expect that the orthonormal basis is the first 3 columns of Q in result of the following command:

```
>> [Q R] = qr(A)      %Calculate the QR factorization of A
```

Indeed, the first 3 columns of Q match those of the orthonormal basis that we calculated using the Gram-Schmidt process.

Notice that the MATLAB `qr` command is much easier to use than the Gram-Schmidt process. The `qr` command in MATLAB is also less computationally demanding on your computer (more efficient and not prone to rounding errors). The Gram-Schmidt method is numerically unstable when implemented on a computer—the vectors that are returned are often not quite orthogonal because of rounding errors in the calculation. Therefore, the QR factorization method is usually preferred when using a computer.

Consider the following matrix A :

$$A = \begin{bmatrix} 4 & -5 & 1 \\ 1 & 2 & 1 \\ -2 & 5 & 2 \\ 4 & -8 & 8 \end{bmatrix}$$

Use MATLAB to determine if the columns of A are linearly independent? State which MATLAB commands you used and explain your answer.

Using MATLAB, find the QR factorization of matrix A (i.e. find Q and R such that $A = QR$).

What is an orthonormal basis for the column space of matrix A ?

Given any $m \times n$ matrix B that has already been entered in MATLAB, what single command could you use to determine if matrix B has orthonormal columns?