Engineering Analysis

Mathematical Modeling and Analysis for Engineers

Contents

Articles

Wikibooks:Collections Preface

This book was created by volunteers at Wikibooks (http://en.wikibooks.org).

What is Wikibooks?

Started in 2003 as an offshoot of the popular Wikipedia project, Wikibooks is a free, collaborative wiki website dedicated to creating high-quality textbooks and other educational books for students around the world. In addition to English, Wikibooks is available in over 130 languages, a complete listing of which can be found at http://www.wikibooks.org. Wikibooks is a "wiki", which means anybody can edit the content there at any time. If you find an error or omission in this book, you can log on to Wikibooks to make corrections and additions as necessary. All of your changes go live on the website immediately, so your effort can be enjoyed and utilized by other readers and editors without delay.

Books at Wikibooks are written by volunteers, and can be accessed and printed for free from the website. Wikibooks is operated entirely by donations, and a certain portion of proceeds from sales is returned to the Wikimedia Foundation to help keep Wikibooks running smoothly. Because of the low overhead, we are able to produce and sell books for much cheaper then proprietary textbook publishers can. **This book can be edited by anybody at any time, including you.** We don't make you wait two years to get a new edition, and we don't stop selling old versions when a new one comes out.

Note that Wikibooks is not a publisher of books, and is not responsible for the contributions of its volunteer editors. PediaPress.com is a print-on-demand publisher that is also not responsible for the content that it prints. Please see our disclaimer for more information: http://en.wikibooks.org/wiki/Wikibooks:General_disclaimer .

What is this book?

This book was generated by the volunteers at Wikibooks, a team of people from around the world with varying backgrounds. The people who wrote this book may not be experts in the field. Some may not even have a passing familiarity with it. The result of this is that some information in this book may be incorrect, out of place, or misleading. For this reason, you should never rely on a community-edited Wikibook when dealing in matters of medical, legal, financial, or other importance. Please see our disclaimer for more details on this.

Despite the warning of the last paragraph, however, books at Wikibooks are continuously edited and improved. If errors are found they can be corrected immediately. If you find a problem in one of our books, we ask that you **be bold** in fixing it. You don't need anybody's permission to help or to make our books better.

Wikibooks runs off the assumption that many eyes can find many errors, and many able hands can fix them. Over time, with enough community involvement, the books at Wikibooks will become very high-quality indeed. **You are invited to participate at Wikibooks to help make our books better**. As you find problems in your book don't just complain about them: Log on and fix them! This is a kind of proactive and interactive reading experience that you probably aren't familiar with yet, so log on to http://en.wikibooks.org and take a look around at all the possibilities. We promise that we won't bite!

Who are the authors?

The volunteers at Wikibooks come from around the world and have a wide range of educational and professional backgrounds. They come to Wikibooks for different reasons, and perform different tasks. Some Wikibookians are prolific authors, some are perceptive editors, some fancy illustrators, others diligent organizers. Some Wikibookians find and remove spam, vandalism, and other nonsense as it appears. Most wikibookians perform a combination of these jobs.

It's difficult to say who are the authors for any particular book, because so many hands have touched it and so many changes have been made over time. It's not unheard of for a book to have been edited thousands of times by hundreds of authors and editors. *You could be one of them too*, if you're interested in helping out.

Wikibooks in Class

Books at Wikibooks are free, and with the proper editing and preparation they can be used as cost-effective textbooks in the classroom or for independent learners. In addition to using a Wikibook as a traditional read-only learning aide, it can also become an interactive class project. Several classes have come to Wikibooks to write new books and improve old books as part of their normal course work. In some cases, the books written by students one year are used to teach students in the same class next year. Books written can also be used in classes around the world by students who might not be able to afford traditional textbooks.

Happy Reading!

We at Wikibooks have put a lot of effort into these books, and we hope that you enjoy reading and learning from them. We want you to keep in mind that what you are holding is not a finished product but instead a work in progress. These books are never "finished" in the traditional sense, but they are ever-changing and evolving to meet the needs of readers and learners everywhere. Despite this constant change, we feel our books can be reliable and high-quality learning tools at a great price, and we hope you agree. Never hesitate to stop in at Wikibooks and make some edits of your own. We hope to see you there one day. **Happy reading!**

Linear Algebra

Engineering Analysis/Vector Spaces

Vectors and Scalars

A **scalar** is a single number value, such as 3, 5, or 10. A **vector** is an ordered set of scalars.

A vector is typically described as a matrix with a row or column size of 1. A vector with a column size of 1 is a **row vector**, and a vector with a row size of 1 is a **column vector**.

[Column Vector]

$$\begin{bmatrix} a \\ b \\ c \\ \vdots \end{bmatrix}$$

[Row Vector]

$$\begin{bmatrix} a & b & c & \cdots \end{bmatrix}$$

A "common vector" is another name for a column vector, and this book will simply use the word "vector" to refer to a common vector.

Vector Spaces

A vector space is a set of vectors and two operations (addition and multiplication, typically) that follow a number of specific rules. We will typically denote vector spaces with a capital-italic letter: *V*, for instance. A space *V* is a vector space if all the following requirements are met. We will be using x and y as being arbitrary vectors in *V*. We will also use c and d as arbitrary scalar values. There are 10 requirements in all:

Given: $x, y \in V$

1. There is an operation called "Addition" (signified with a "+" sign) between two vectors, x + y, such that if both the operands are in *V*, then the result is also in *V*.
2. The addition operation is commutative for all elements in *V*.
3. The addition operation is associative for all elements in *V*.
4. There is a unique **neutral element**, φ, in *V*, such that x + φ = x. This is also called a **zero** element.
5. For every x in *V*, then there is a negative element -x in *V* such that -x + x = φ.
6. $cx \in V$
7. $c(x + y) = cx + cy$
8. $(c + d)x = cx + dx$
9. $c(dx) = cdx$
10. 1 × x = x

Some of these rules may seem obvious, but that's only because they have been generally accepted, and have been taught to people since they were children.

Engineering Analysis/Vector Basics

Scalar Product

A scalar product is a special type of operation that acts on two vectors, and returns a scalar result. Scalar products are denoted as an ordered pair between angle-brackets: <x,y>. A scalar product between vectors must satisfy the following four rules:

1. $\langle x, x \rangle \geq 0, \quad \forall x \in V$
2. $\langle x, x \rangle = 0$, only if x = 0
3. $\langle x, y \rangle = \langle y, x \rangle$
4. $\langle x, cy_1 + dy_2 \rangle = c\langle x, y_1 \rangle + d\langle x, y_2 \rangle$

If an operation satisifes all these requirements, then it is a scalar product.

Examples

One of the most common scalar products is the **dot product**, that is discussed commonly in Linear Algebra

Norm

The **norm** is an important scalar quantity that indicates the magnitude of the vector. Norms of a vector are typically denoted as $\|x\|$. To be a norm, an operation must satisfy the following four conditions:

1. $\|x\| \geq 0$
2. $\|x\| = 0$only if x = 0.
3. $\|cx\| = |c|\|x\|$
4. $\|x + y\| \leq \|x\| + \|y\|$

A vector is called **normal** if it's norm is 1. A normal vector is sometimes also referred to as a **unit vector**. Both notations will be used in this book. To make a vector normal, but keep it pointing in the same direction, we can divide the vector by its norm:

$$\bar{x} = \frac{x}{\|x\|}$$

Examples

One of the most common norms is the **cartesian norm**, that is defined as the square-root of the sum of the squares:

$$\|x\| = \sqrt{x_1^2 + x_2^2 + \cdots + x_n^2}$$

Unit Vector

A vector is said to be a **unit vector** if the norm of that vector is 1.

Orthogonality

Two vectors x and y are said to be **orthogonal** if the scalar product of the two is equal to zero:

$$\langle x, y \rangle = 0$$

Two vectors are said to be **orthonormal** if their scalar product is zero, and both vectors are unit vectors.

Cauchy-Schwartz Inequality

The cauchy-schwartz inequality is an important result, and relates the norm of a vector to the scalar product:

$$|\langle x, y \rangle| \leq \|x\| \|y\|$$

Metric (Distance)

The distance between two vectors in the vector space *V*, called the **metric** of the two vectors, is denoted by d(x, y). A metric operation must satisfy the following four conditions:

1. $d(x, y) \geq 0$
2. $d(x, y) = 0$ only if x = y
3. $d(x, y) = d(y, x)$
4. $d(x, y) \leq d(x, z) + d(z, y)$

Examples

A common form of metric is the distance between points a and b in the cartesian plane:

$$d(a, b)_{cartesian} = \sqrt{(x_a - x_b)^2 + (y_a - y_b)^2}$$

Engineering Analysis/Linear Independence and Basis

Linear Independance

A set of vectors $V = v_1, v_2, \cdots, v_n$ are said to be linearly dependant on one another if any vector v from the set can be constructed from a linear combination of the other vectors in the set. Given the following linear equation:

$$a_1 v_1 + a_2 v_2 + \cdots + a_n v_n = 0$$

The set of vectors V is linearly independent only if all the a coefficients are zero. If we combine the v vectors together into a single row vector:

$$\hat{V} = [v_1 v_2 \cdots v_n]$$

And we combine all the a coefficients into a single column vector:

$$\hat{a} = [a_1 a_2 \cdots a_n]^T$$

We have the following linear equation:

$$\hat{V}\hat{a} = 0$$

We can show that this equation can only be satisifed for $\hat{a} = 0$, the matrix $\hat{V}$ must be invertable:

$$\hat{V}^{-1}\hat{V}\hat{a} = \hat{V}^{-1}0$$
$$\hat{a} = 0$$

Remember that for the matrix to be invertable, the determinate must be non-zero.

Non-Square Matrix V

If the matrix $\hat{V}$is not square, then the determinate can not be taken, and therefore the matrix is not invertable. To solve this problem, we can premultiply by the transpose matrix:

$$\hat{V}^T\hat{V}\hat{a} = 0$$

And then the square matrix $\hat{V}^T\hat{V}$must be invertable:

$$(\hat{V}^T\hat{V})^{-1}\hat{V}^T\hat{V}\hat{a} = 0$$
$$\hat{a} = 0$$

Rank

The rank of a matrix is the largest number of linearly independent rows or columns in the matrix.

To determine the Rank, typically the matrix is reduced to row-echelon form. From the reduced form, the number of non-zero rows, or the number of non-zero columns (whichever is smaller) is the rank of the matrix.

If we multiply two matrices A and B, and the result is C:

$$AB = C$$

Then the rank of C is the minimum value between the ranks A and B:

$$\text{Rank}(C) = \min[\text{Rank}(A), \text{Rank}(B)]$$

Span

A **Span** of a set of vectors *V* is the set of all vectors that can be created by a linear combination of the vectors.

Basis

A **basis** is a set of linearly-independent vectors that span the entire vector space.

Basis Expansion

If we have a vector $y \in V$, and *V* has basis vectors $v_1 v_2 \cdots v_n$, by definition, we can write y in terms of a linear combination of the basis vectors:

$$a_1 v_1 + a_2 v_2 + \cdots + a_n v_n = y$$

or

$$\hat{V}\hat{a} = y$$

If $\hat{V}$is invertable, the answer is apparent, but if $\hat{V}$is not invertable, then we can perform the following technique:

$$\hat{V}^T\hat{V}\hat{a} = \hat{V}^T y$$
$$\hat{a} = (\hat{V}^T\hat{V})^{-1}\hat{V}^T y$$

And we call the quantity $(\hat{V}^T\hat{V})^{-1}\hat{V}^T$the **left-pseudoinverse** of $\hat{V}$.

Change of Basis

Frequently, it is useful to change the basis vectors to a different set of vectors that span the set, but have different properties. If we have a space *V*, with basis vectors $\hat{V}$ and a vector in *V* called x, we can use the new basis vectors $\hat{W}$ to represent x:

$$x = \sum_{i=0}^{n} a_i v_i = \sum_{j=1}^{n} b_j w_j$$

or,

$$x = \hat{V}\hat{a} = \hat{W}\hat{b}$$

If V is invertable, then the solution to this problem is simple.

Grahm-Schmidt Orthogonalization

If we have a set of basis vectors that are not orthogonal, we can use a process known as **orthogonalization** to produce a new set of basis vectors for the same space that are orthogonal:

Given: $\hat{V} = x_1 v_2 \cdots v_n$

Find the new basis $\hat{W} = w_1 w_2 \cdots w_n$

Such that $\langle w_i, w_j \rangle = 0 \quad \forall i, j$

We can define the vectors as follows:

1. $w_1 = v_1$
2. $w_m = v_m - \sum_{i=1}^{m-1} \frac{\langle v_m, u_i \rangle}{\langle u_i, u_i \rangle} u_i$

Notice that the vectors produced by this technique are orthogonal to each other, but they are not necessarily orthonormal. To make the *w* vectors orthonormal, you must divide each one by its norm:

$$\bar{w} = \frac{w}{\|w\|}$$

Reciprocal Basis

A Reciprocal basis is a special type of basis that is related to the original basis. The reciprocal basis $\hat{W}$ can be defined as:

$$\hat{W} = [\hat{V}^T]^{-1}$$

Engineering Analysis/Linear Transformations

Linear Transformations

A linear transformation is a matrix M that operates on a vector in space *V*, and results in a vector in a different space *W*. We can define a transformation as such:

$$T : V \to W$$

In the above equation, we say that *V* is the **domain space** of the transformation, and *W* is the **range space** of the transformation. Also, we can use a "function notation" for the transformation, and write it as:

$$M(x) = Mx = y$$

Where x is a vector in *V*, and y is a vector in *W*. To be a linear transformation, the principle of superposition must hold for the transformation:

$$M(av_1 + bv_2) = aM(v_1) + bM(v_2)$$

Where a and b are arbitrary scalars.

Null Space

The Nullspace of an equation is the set of all vectors x for which the following relationship holds:

$$Mx = 0$$

Where M is a linear transformation matrix. Depending on the size and rank of M, there may be zero or more vectors in the nullspace. Here are a few rules to remember:

1. If the matrix M is invertable, then there is no nullspace.
2. The number of vectors in the nullspace (N) is the difference between the rank(R) of the matrix and the number of columns(C) of the matrix:

$$N = R - C$$

If the matrix is in row-eschelon form, the number of vectors in the nullspace is given by the number of rows without a leading 1 on the diagonal. For every column where there is not a leading one on the diagonal, the nullspace vectors can be obtained by placing a negative one in the leading position for that column vector.

We denote the nullspace of a matrix A as:

$$\mathcal{N}\{A\}$$

Linear Equations

If we have a set of linear equations in terms of variables x, scalar coefficients a, and a scalar result b, we can write the system in matrix notation as such:

$$Ax = b$$

Where x is a m × 1 vector, b is an n × 1 vector, and A is an n × m matrix. Therefore, this is a system of n equations with m unknown variables. There are 3 possibilities:

1. If Rank(A) is not equal to Rank([A b]), there is no solution
2. If Rank(A) = Rank([A b]) = n, there is exactly one solution
3. If Rank(A) = Rank([A b]) < n, there are infinitely many solutions.

Complete Solution

The complete solution of a linear equation is given by the sum of the **homogeneous solution**, and the **particular solution**. The homogeneous solution is the nullspace of the transformation, and the particular solution is the values for x that satisfy the equation:

$$A(x) = b$$
$$A(x_h + x_p) = b$$

Where

x_h is the homogeneous solution, and is the nullspace of A that satisfies the equation $A(x_h) = 0$

x_p is the particular solution that satisfies the equation $A(x_p) = b$

Minimum Norm Solution

If Rank(A) = Rank([A b]) < n, then there are infinitely many solutions to the linear equation. In this situation, the solution called the **minimum norm** solution must be found. This solution represents the "best" solution to the problem. To find the minimum norm solution, we must minimize the norm of x subject to the constraint of:

$$Ax - b = 0$$

There are a number of methods to minimize a value according to a given constraint, and we will talk about them later.

Least-Squares Curve Fit

If Rank(A) doesnt equal Rank([A b]), then the linear equation has no solution. However, we can find the solution which is the closest. This "best fit" solution is known as the Least-Squares curve fit.

We define an error quantity E, such that:

$$E = Ax - b \neq 0$$

Our job then is to find the minimum value for the norm of E:

$$\|E\|^2 = \|Ax - b\|^2 =< Ax - b, Ax - b >$$

We do this by differentiating with respect to x, and setting the result to zero:

$$\frac{\partial \|E\|^2}{\partial x} = 2A'(Ax - b) = 0$$

Solving, we get our result:

$$x = (A^T A)^{-1} A^T b$$

Engineering Analysis/Minimization

Khun-Tucker Theorem

The Khun-Tucker Theorem is a method for minimizing a function f(x) under the constraint g(x). We can define the theorem as follows:

$$L(x) = f(x) + \langle \Lambda, g(x) \rangle$$

Where Λ is the *lagrangian vector*, and < , > denotes the **scalar product** operation. We will discuss scalar products more later. If we differentiate this equation with respect to x first, and then with respect to Λ, we get the following two equations:

$$\frac{\partial L(x)}{\partial x} = x + A\Lambda$$
$$\frac{\partial L(x)}{\partial \Lambda} = Ax - b$$

We have the final result:

$$x = A^T[AA^T]^{-1}b$$

Engineering Analysis/Projections

Projection

The projection of a vector $v \in V$ onto the vector space $W \in V$ is the minimum distance between v and the space *W*. In other words, we need to minimize the distance between vector v, and an arbitrary vector $w \in W$:

$$\|w - v\|^2 = \|\hat{W}\hat{a} - v\|^2$$
$$\frac{\partial \|\hat{W}\hat{a} - v\|^2}{\partial \hat{a}} = \frac{\partial \langle \hat{W}\hat{a} - v, \hat{W}\hat{a} - v \rangle}{\partial \hat{a}} = 0$$

[Projection onto space W]

$$\hat{a} = (\hat{W}^T\hat{W})^{-1}\hat{W}^T v$$

For every vector $v \in V$ there exists a vector $w \in W$ called the projection of v onto W such that <v-w, p> = 0, where p is an arbitrary element of *W*.

Orthogonal Complement

$$w^{\perp} = x \in V : \langle x, y \rangle = 0, \forall y \in W$$

Distance between v and W

The distance between $v \in V$ and the space *W* is given as the minimum distance between v and an arbitrary $w \in W$:

$$\frac{\partial d(v, w)}{\partial \hat{a}} = \frac{\partial \|v - \hat{W}\hat{a}\|}{\partial \hat{a}} = 0$$

Intersections

Given two vector spaces *V* and *W*, what is the overlapping area between the two? We define an arbitrary vector z that is a component of both *V*, and *W*:

$$z = \hat{V}\hat{a} = \hat{W}\hat{b}$$
$$\hat{V}\hat{a} - \hat{W}\hat{b} = 0$$
$$\begin{bmatrix}\hat{a}\\ \hat{b}\end{bmatrix} = \mathcal{N}([\hat{v} - \hat{W}])$$

Where N is the nullspace.

Engineering Analysis/Linear Spaces

Linear Spaces

Linear Spaces are like Vector Spaces, but are more general. We will define Linear Spaces, and then use that definition later to define Function Spaces.

If we have a space X, elements in that space f and g, and scalars a and b, the following rules must hold for X to be a linear space:

1. $f + g \in X$
2. $f + g = g + f$
3. There is a null element φ such that φ + *f* = *f*. $\phi \in X$
4. $f \in X, -f \in X$
5. *f* + *(-f)* = φ

Engineering Analysis/Matrices

Norms

Derivatives

Consider the following set of linear equations:

$$a = bx_1 + cx_2$$
$$d = ex_1 + fx_2$$

We can define the matrix A to represent the coefficients, the vector B as the results, and the vector x as the variables:

$$A = \begin{bmatrix} b & c \\ e & f \end{bmatrix}$$
$$B = \begin{bmatrix} a \\ d \end{bmatrix}$$
$$x = \begin{bmatrix} x_1 \\ x_2 \end{bmatrix}$$

And rewriting the equation in terms of the matrices, we get:

$$B = Ax$$

Now, let's say we want the derivative of this equation with respect to the vector x:

$$\frac{d}{dx}B = \frac{d}{dx}Ax$$

We know that the first term is constant, so the derivative of the left-hand side of the equation is zero. Analyzing the right side shows us:

Pseudo-Inverses

There are special matrices known as **pseudo-inverses**, that satisfies some of the properties of an inverse, but not others. To recap, If we have two square matrices A and B, that are both n × n, then if the following equation is true, we say that A is the inverse of B, and B is the inverse of A:

$$AB = BA = I$$

Right Pseudo-Inverse

Consider the following matrix:

$$R = A^T[AA^T]^{-1}$$

We call this matrix R the **right pseudo-inverse** of A, because:

$$AR = I$$

but

$$RA \neq I$$

We will denote the right pseudo-inverse of A as $A^\dagger$

Left Pseudo-Inverse

Consider the following matrix:

$$L = [A^TA]^{-1}A^T$$

We call L the **left pseudo-inverse** of A because

$$LA = I$$

but

$$AL \neq I$$

We will denote the left pseudo-inverse of A as $A^\ddagger$

Engineering Analysis/Matrix Forms

Matrices that follow certain predefined formats are useful in a number of computations. We will discuss some of the common matrix formats here. Later chapters will show how these formats are used in calculations and analysis.

Diagonal Matrix

A diagonal matrix is a matrix such that:

$$a_{ij} = 0, i \neq j$$

In otherwords, all the elements off the main diagonal are zero, and the diagonal elements may be (but don't need to be) non-zero.

Companion Form Matrix

If we have the following characteristic polynomial for a matrix:

$$|A - \lambda I| = \lambda^n + a_{n-1}\lambda^{n-1} + \cdots + a_1\lambda^1 + a_0$$

We can create a **companion form** matrix in one of two ways:

$$\begin{bmatrix} 0 & 0 & 0 & \cdots & 0 & -a_0 \\ 1 & 0 & 0 & \cdots & 0 & -a_1 \\ 0 & 1 & 0 & \cdots & 0 & -a_2 \\ 0 & 0 & 1 & \cdots & 0 & -a_3 \\ \vdots & \vdots & \vdots & \ddots & \vdots & \vdots \\ 0 & 0 & 0 & \cdots & 1 & -a_{n-1} \end{bmatrix}$$

Or, we can also write it as:

$$\begin{bmatrix} -a_{n-1} & -a_{n-2} & -a_{n-3} & \cdots & a_1 & a_0 \\ 0 & 0 & 0 & \cdots & 0 & 0 \\ 1 & 0 & 0 & \cdots & 0 & 0 \\ 0 & 1 & 0 & \cdots & 0 & 0 \\ 0 & 0 & 1 & \cdots & 0 & 0 \\ \vdots & \vdots & \vdots & \ddots & \vdots & \vdots \\ 0 & 0 & 0 & \cdots & 1 & 0 \end{bmatrix}$$

Jordan Canonical Form

To discuss the Jordan canonical form, we first need to introduce the idea of the **Jordan Block**:

Jordan Blocks

A jordan block is a square matrix such that all the diagonal elements are equal, and all the super-diagonal elements (the elements directly above the diagonal elements) are all 1. To illustrate this, here is an example of an n-dimensional jordan block:

$$\begin{bmatrix} a & 1 & 0 & \cdots & 0 \\ 0 & a & 1 & \cdots & 0 \\ 0 & 0 & a & \cdots & 0 \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & a & \cdots & 1 \\ 0 & 0 & 0 & \cdots & a \end{bmatrix}$$

Canonical Form

A square matrix is in **Jordan Canonical form**, if it is a diagonal matrix, or if it has one of the following two block-diagonal forms:

$$\begin{bmatrix} D & 0 & \cdots & 0 \\ 0 & J_1 & \cdots & 0 \\ \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & \cdots & J_n \end{bmatrix}$$

Or:

$$\begin{bmatrix} J_1 & 0 & \cdots & 0 \\ 0 & J_2 & \cdots & 0 \\ \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & \cdots & J_n \end{bmatrix}$$

The where the D element is a diagonal block matrix, and the J blocks are in Jordan block form.

Engineering Analysis/Quadratic Forms

If we have an n × 1 vector x, and an n × n symmetric matrix M, we can write:

$$x^T M x = a$$

Where a is a scalar value. Equations of this form are called **quadratic forms**.

Matrix Definiteness

Based on the quadratic forms of a matrix, we can create a certain number of categories for special types of matrices:

1. if $x^T M x > 0$ for all x, then the matrix is **positive definite**.
2. if $x^T M x \geq 0$ for all x, then the matrix is **positive semi-definite**.
3. if $x^T M x < 0$ for all x, then the matrix is **negative definite**.
4. if $x^T M x \leq 0$ for all x, then the matrix is **negative semi-definite**.

These classifications are used commonly in control engineering.

Engineering Analysis/Eigenvalues and Eigenvectors

The Eigen Problem

This page is going to talk about the concept of **Eigenvectors** and **Eigenvalues**, which are important tools in linear algebra, and which play an important role in State-Space control systems. The "Eigen Problem" stated simply, is that given a square matrix A which is n × n, there exists a set of n scalar values λ and n corresponding non-trivial vectors v such that:

$$Av = \lambda v$$

We call λ the **eigenvalues** of A, and we call **v** the corresponding **eigenvectors** of A. We can rearrange this equation as:

$$(A - \lambda I)v = 0$$

For this equation to be satisfied so that v is non-trivial, the matrix (A - λI) must be singular. That is:

$$|A - \lambda I| = 0$$

Characteristic Equation

The characteristic equation of a square matrix A is given by:

[Characteristic Equation]

$$|A - \lambda I| = 0$$

Where I is the identity matrix, and λ is the set of **eigenvalues** of matrix A. From this equation we can solve for the eigenvalues of A, and then using the equations discussed above, we can calculate the corresponding eigenvectors.

In general, we can expand the characteristic equation as:

[Characteristic Polynomial]

$$|A - \lambda I| = (-1)^n(\lambda^n + c_{n-1}\lambda^{n-1} + \cdots + c_1\lambda^1 + c_0)$$

This equation satisfies the following properties:

1. $|A| = (-1)^n c_0$

2. A is nonsingular if c_0 is non-zero.

Example: 2 × 2 Matrix

Let's say that X is a square matrix of order 2, as such:

$$X = \begin{bmatrix} a & b \\ c & d \end{bmatrix}$$

Then we can use this value in our characteristic equation:

$$\begin{vmatrix} a-\lambda & b \\ c & d-\lambda \end{vmatrix} = 0$$

$$(a-\lambda)(d-\lambda) - (b)(c) = 0$$

The roots to the above equation (the values for λ that satisfies the equality) are the eigenvalues of X.

Eigenvalues

The solutions, λ, of the characteristic equation for matrix X are known as the **eigenvalues** of the matrix X.

Eigenvalues satisfy the following properties:

1. If λ is an eigenvalue of A, λ^n is an eigenvalue of A^n.
2. If λ is a complex eigenvalue of A, then λ^* (the complex conjugate) is also an eigenvalue of A.
3. If any of the eigenvalues of A are zero, then A is singular. If A is non-singular, all the eigenvalues of A are nonzero.

Eigenvectors

The characteristic equation can be rewritten as such:

$$Xv = \lambda v$$

Where X is the matrix under consideration, and λ are the eigenvalues for matrix X. For every unique eigenvalue, there is a solution vector v to the above equation, known as an **eigenvector**. The above equation can also be rewritten as:

$$|X - \lambda I|v = 0$$

Where the resulting values of v for each eigenvalue λ is an eigenvector of X. There is a unique eigenvector for each unique eigenvalue of X. From this equation, we can see that the eigenvectors of A form the nullspace:

$$v = \mathcal{N}\{A - \lambda I\}$$

And therefore, we can find the eigenvectors through row-reduction of that matrix.

Eigenvectors satisfy the following properties:

1. If v is a complex eigenvector of A, then v^* (the complex conjugate) is also an eigenvector of A.
2. Distinct eigenvectors of A are linearly independent.
3. If A is n × n, and if there are n distinct eigenvectors, then the eigenvectors of A form a complete basis set for $\mathcal{R}^n$

Generalized Eigenvectors

Let's say that matrix **A** has the following characteristic polynomial:

$$(A - \lambda I) = (-1)^n(\lambda - \lambda_1)^{d_1}(\lambda - \lambda_2)^{d_2} \cdots (\lambda - \lambda_s)^{d_s}$$

Where d_1, d_2, ... , d_s are known as the **algebraic multiplicity** of the eigenvalue λ_i. Also note that $d_1 + d_2 + ... + d_s = n$, and $s < n$. In other words, the eigenvalues of A are repeated. Therefore, this matrix doesnt have n distinct eigenvectors. However, we can create vectors known as **generalized eigenvectors** to make up the missing eigenvectors by satisfying the following equations:

$$(A - \lambda I)^k v_k = 0$$
$$(A - \lambda I)^{k-1} v_k = 0$$

Right and Left Eigenvectors

The equation for determining eigenvectors is:

$$(A - \lambda I)v = 0$$

And because the eigenvector *v* is on the right, these are more appropriately called "right eigenvectors". However, if we rewrite the equation as follows:

$$u(A - \lambda I) = 0$$

The vectors *u* are called the "left eigenvectors" of matrix A.

Engineering Analysis/Diagonalization

Similarity

Matrices *A* and *B* are said to be similar to one another if there exists an invertable matrix *T* such that:

$$T^{-1}AT = B$$

If there exists such a matrix *T*, the matrices are similar. Similar matrices have the same eigenvalues. If *A* has eigenvectors v_1, v_2 ..., then *B* has eigenvectors *u* given by:

$$u_i = Tv_i$$

Matrix Diagonalization

Some matricies are similar to diagonal matrices using a **transition matrix**, *T*. We will say that matrix *A* is diagonalizable if the following equation can be satisfied:

$$T^{-1}AT = D$$

Where *D* is a diagonal matrix. An $n \times n$ square matrix is diagonalizable if and only if it has *n* linearly independent eigenvectors.

Transition Matrix

If an $n \times n$ square matrix has n distinct eigenvalues λ, and therefore n distinct eigenvectors v, we can create a transition matrix T as:

$$T = [v_1 v_2 ... v_n]$$

And transforming matrix X gives us:

$$T^{-1}AT = \begin{bmatrix} \lambda_1 & 0 & \cdots & 0 \\ 0 & \lambda_2 & \cdots & 0 \\ \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & \cdots & \lambda_n \end{bmatrix}$$

Therefore, if the matrix has n distinct eigenvalues, the matrix is diagonalizable, and the diagonal entries of the diagonal matrix are the corresponding eigenvalues of the matrix.

Complex Eigenvalues

Consider the situation where a matrix A has 1 or more complex conjugate eigenvalue pairs. The eigenvectors of A will also be complex. The resulting diagonal matrix D will have the complex eigenvalues as the diagonal entries. In engineering situations, it is often not a good idea to deal with complex matrices, so other matrix transformations can be used to create matrices that are "nearly diagonal".

Generalized Eigenvectors

If the matrix A does not have a complete set of eigenvectors, that is, that they have d eigenvectors and n - d generalized eigenvectors, then the matrix A is not diagonalizable. However, the next best thing is acheived, and matrix A can be transformed into a Jordan Cannonical Matrix. Each set of generalized eigenvectors that are formed from a single eigenvector basis will create a jordan block. All the distinct eigenvectors that do not spawn any generalized eigenvectors will form a diagonal block in the Jordan matrix.

Engineering Analysis/Spectral Decomposition

If λ_i are the *n* distinct eigenvalues of matrix *A*, and v_i are the corresponding n distinct eigenvectors, and if w_i are the *n* distinct left-eigenvectors, then the matrix *A* can be represented as a sum:

$$A = \sum_{i=1}^{n} \lambda_i v_i w_i^T$$

this is known as the **spectral decomposition** of *A*.

Engineering Analysis/Error Estimation

Consider a scenario where the matrix representation of a system A differs from the actual implementation of the system by a factor of ΔA. In other words, our system uses the matrix:

$$A + \Delta A$$

From the study of Control Systems, we know that the values of the eigenvectors can affect the stability of the system. For that reason, we would like to know how a small error in A will affect the eigenvalues.

First off, we assume that ΔA is a *small* shift. The definition of "small" in this sense is arbitrary, and will remained open. Keep in mind that the techniques discussed here are more accurate the smaller ΔA is.

If ΔA is the error in the matrix A, then Δλ is the error in the eigenvalues and Δv is the error in the eigenvectors. The characteristic equation becomes:

$$(A + \Delta A)(v + \Delta v) = (\lambda + \Delta\lambda)(v + \Delta v)$$

We have an equation now with two unknowns: Δλ and Δv. In other words, we don't know how a small change in A will affect the eigenvalues and eigenvectors. If we multiply out both sides, we get:

$$Av + \Delta A v + A\Delta v + O(\Delta^2) = \lambda v + \Delta\lambda v + v\Delta\lambda + O(\Delta^2)$$

This situation seems hopeless, until we multiply both sides by the corresponding left-eigenvector w from the left:

$$w^T A v + w^T \Delta A v + w^T v \Delta A = w^T \lambda v + w^T \Delta\lambda v + w^T v \Delta\lambda + O(\Delta^2)$$

Terms where two Δs (which are known to be small, by definition) are multiplied together, we can say are negligible, and ignore them. Also, we know from our right-eigenvalue equation that:

$$w^T A = \lambda w^T$$

Another fact is that the right-eigenvectors and left eigenvectors are orthogonal to each other, so the following result holds:

$$w^T v = 0$$

Substituting these results, where necessary, into our long equation above, we get the following simplification:

$$w^T \Delta A v = \Delta\lambda w^T \Delta v$$

And solving for the change in the eigenvalue gives us:

$$\Delta\lambda = \frac{w^T \Delta A v}{w^T \Delta v}$$

This approximate result is only good for small values of ΔA, and the result is less precise as the error increases.

Matrix Calculus

Engineering Analysis/Matrix Functions

If we have functions, and we use a matrix as the input to those functions, the output values are not always intuitive. For instance, if we have a function *f(x)*, and as the input argument we use matrix *A*, the output matrix is not necessarily the function *f* applied to the individual elements of *A*.

Diagonal Matrix

In the special case of diagonal matrices, the result of *f(A)* is the function applied to each element of the diagonal matrix:

$$A = \begin{bmatrix} a_{11} & 0 & \cdots & 0 \\ 0 & a_{22} & \cdots & 0 \\ \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & \cdots & a_{nn} \end{bmatrix}$$

Then the function *f(A)* is given by:

$$f(A) = \begin{bmatrix} f(a_{11}) & 0 & \cdots & 0 \\ 0 & f(a_{22}) & \cdots & 0 \\ \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & \cdots & f(a_{nn}) \end{bmatrix}$$

Jordan Cannonical Form

Matrices in Jordan Canonical form also have an easy way to compute the functions of the matrix. However, this method is not nearly as easy as the diagonal matrices described above.

If we have a matrix in Jordan Block form, *A*, the function *f(A)* is given by:

$$f(A) = \begin{bmatrix} \frac{f(a)}{0!} & \frac{f'(a)}{1!} & \cdots & \frac{f^{(r-1)}(a)}{(r-1)!} \\ 0 & \frac{f(a)}{0!} & \cdots & \frac{f^(r-2)(a)}{(r-2)!} \\ \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & \cdots & \frac{f(a)}{0!} \end{bmatrix}$$

The matrix indices have been removed, because in Jordan block form, all the diagonal elements must be equal.

If the matrix is in Jordan Block form, the value of the function is given as the function applied to the individual diagonal blocks.

Engineering Analysis/Cayley Hamilton Theorem

If the characteristic equation of matrix *A* is given by:

$$\Delta(\lambda) = |A - \lambda I| = (-1)^n(\lambda^n + a_{n-1}\lambda^{n-1} + \cdots + a_0) = 0$$

Then the Cayley-Hamilton theorem states that the matrix *A* itself is also a valid solution to that equation:

$$\Delta(A) = (-1)^n(A^n + a_{n-1}A^{n-1} + \cdots + a_0) = 0$$

Another theorem worth mentioning here (and by "worth mentioning", we really mean "fundamental for some later topics") is stated as:

If λ are the eigenvalues of matrix *A*, and if there is a function *f* that is defined as a linear combination of powers of λ:

$$f(\lambda) = \sum_{i=0}^{\infty} b_i \lambda^i$$

If this function has a radius of convergence *S*, and if all the eigenvectors of *A* have magnitudes less then *S*, then the matrix *A* itself is also a solution to that function:

$$f(A) = \sum_{i=0}^{\infty} b_i A^i$$

Engineering Analysis/Matrix Exponentials

Matrix Exponentials

If we have a matrix *A*, we can raise that matrix to a power of *e* as follows:

$$e^A$$

It is important to note that this is not necessarily (not usually) equal to each individual element of *A* being raised to a power of *e*. Using taylor-series expansion of exponentials, we can show that:

$$e^A = I + A + \frac{1}{2}A^2 + \frac{1}{6}A^3 + ... = \sum_{k=0}^{\infty} \frac{1}{k!}A^k.$$

In other words, the matrix exponential can be reducted to a sum of powers of the matrix. This follows from both the taylor series expansion of the exponential function, and the cayley-hamilton theorem discussed previously.

However, this infinite sum is expensive to compute, and because the sequence is infinite, there is no good cut-off point where we can stop computing terms and call the answer a "good approximation". To alleviate this point, we can turn to the Cayley-Hamilton Theorem. Solving the Theorem for A^n, we get:

$$A^n = -c_{n-1}A^{n-1} - c_{n-2}A^{n-2} - \cdots - c_1A - c_0I$$

Multiplying both sides of the equation by *A*, we get:

$$A^{n+1} = -c_{n-1}A^n - c_{n-2}A^{n-1} - \cdots - c_1A^2 - c_0A$$

We can substitute the first equation into the second equation, and the result will be A^{n+1} in terms of the first *n - 1* powers of *A*. In fact, we can repeat that process so that A^m, for any arbitrary high power of m can be expressed as a linear combination of the first *n - 1* powers of *A*. Applying this result to our exponential problem:

$$e^A = \alpha_0 I + \alpha_1 A + \cdots + \alpha_{n-1}A^{n-1}$$

Where we can solve for the α terms, and have a finite polynomial that expresses the exponential.

Inverse

The inverse of a matrix exponential is given by:

$$(e^A)^{-1} = e^{-A}$$

Derivative

The derivative of a matrix exponential is:

$$\frac{d}{dx}e^{Ax} = Ae^{Ax} = e^{Ax}A$$

Notice that the exponential matrix is commutative with the matrix *A*. This is not the case with other functions, necessarily.

Sum of Matrices

If we have a sum of matrices in the exponent, we cannot separate them:

$$e^{(A+B)x} \neq e^{Ax}e^{Bx}$$

Differential Equations

If we have a first-degree differential equation of the following form:

$$x'(t) = Ax(t) + f(t)$$

With initial conditions

$$x(t_0) = c$$

Then the solution to that equation is given in terms of the matrix exponential:

$$x(t) = e^{a(t-t_0)}c + \int_{t_0}^{t} e^{a(t-\tau)}f(\tau)d\tau$$

This equation shows up frequently in control engineering.

Laplace Transform

As a matter of some interest, we will show the Laplace Transform of a matrix exponential function:

$$\mathcal{L}[e^{At}] = (sI - A)^{-1}$$

We will not use this result any further in this book, although other books on engineering might make use of it.

Engineering Analysis/Lyapunov Equation

[Lyapunov's Equation]

$$AM + MB = C$$

Where *A*, *B* and *C* are constant square matrices, and M is the solution that we are trying to find. If *A*, *B*, and *C* are of the same order, and if *A* and *B* have no eigenvalues in common, then the solution can be given in terms of matrix exponentials:

$$M = -\int_0^\infty e^{Ax} C e^{Bx} dx$$

Engineering Analysis/Function Spaces

Function Space

A function space is a linear space where all the elements of the space are functions. A function space that has a norm operation is known as a **normed function space**. The spaces we consider will all be normed.

Continuity

f(x) is continuous at x_0 if, for every ε > *0* there exists a δ(ε) > *0* such that |*f(x)* - *f*(x_0)| < &epsilon when |*x* - x_0| < δ(ε).

Common Function Spaces

Here is a listing of some common function spaces. This is not an exhaustive list.

C Space

The *C* function space is the set of all functions that are continuous.

The metric for *C* space is defined as:

$$\rho(x, y)_{L_2} = \max |f(x) - g(x)|$$

Consider the metric of *sin(x)* and *cos(x)*:

$$\rho(sin(x), cos(x))_{L_2} = \sqrt{2}, x = \frac{3\pi}{4}$$

C^p Space

The C^p is the set of all continuous functions for which the first *p* derivatives are also continuous. If $p = \infty$ the function is called "infinitely continuous. The set C^∞ is the set of all such functions. Some examples of functions that are infinitely continuous are exponentials, sinusoids, and polynomials.

L Space

The *L* space is the set of all functions that are finitely integrable over a given interval *[a, b]*.

f(x) is in *L(a, b)* if:

$$\int_a^b |f(x)| dx < \infty$$

L_p Space

The L_p space is the set of all functions that are finitely integrable over a given interval *[a, b]* when raised to the power *p*:

$$\int_a^b |f(x)|^p dx < \infty$$

Most importantly for engineering is the L_2 space, or the set of functions that are "square integrable".

Engineering Analysis/L2 Space

The L_2 space is very important to engineers, because functions in this space do not need to be continuous. Many discontinuous engineering functions, such as the delta (impulse) function, the unit step function, and other discontinuous finctions are part of this space.

L2 Functions

A large number of functions qualify as L_2 functions, including uncommon, discontinuous, piece-wise, and other functions. A function which, over a finite range, has a finite number of discontinuties is an L_2 function. For example, a unit step and an impulse function are both L_2 functions. Also, other functions useful in signal analysis, such as square waves, triangle waves, wavelets, and other functions are L_2 functions.

In practice, most physical systems have a finite amount of noise associated with them. Noisy signals and random signals, if finite, are also L_2 functions: this makes analysis of those functions using the techniques listed below easy.

Null Function

The null functions of L_2 are the set of all functions φ in L_2 that satisfy the equation:

$$\int_a^b |\phi(x)|^2 dx = 0$$

for all *a* and *b*.

Norm

The L_2 norm is defined as follows:

[L2 Norm]

$$\|f(x)\|_{L_2} = \sqrt{\int_a^b |f(x)|^2 dx}$$

If the norm of the function is 1, the function is normal.

We can show that the derivative of the norm squared is:

$$\frac{\partial \|x\|^2}{\partial x} = 2x$$

Scalar Product

The scalar product in L_2 space is defined as follows:

[L2 Scalar Product]

$$\langle f(x), g(x) \rangle_{L_2} = \int_a^b f(x)g(x)dx$$

If the scalar product of two functions is zero, the functions are orthogonal.

We can show that given coefficient matrices *A* and *B*, and variable *x*, the derivative of the scalar product can be given as:

$$\frac{\partial}{\partial x}\langle Ax, Bx \rangle = A^T Bx + B^T Ax$$

We can recognize this as the product rule of differentiation. Generalizing, we can say that:

$$\frac{\partial}{\partial x}\langle f(x), g(x) \rangle = f'(x)g(x) + f(x)g'(x)$$

We can also say that the derivative of a matrix *A* times a vector *x* is:

$$\frac{d}{dx}Ax = A^T$$

Metric

The metric of two functions (we will not call it the "distance" here, because that word has no meaning in a function space) will be denoted with $\rho(x,y)$. We can define the metric of an L_2 function as follows:

[L2 Metric]

$$\rho(x,y)_{L_2} = \sqrt{\int_a^b |f(x) - g(x)|^2 dx}$$

Cauchy-Schwarz Inequality

The Cauchy-Schwarz Inequality still holds for L_2 functions, and is restated here:

$$|\langle f(x), g(x) \rangle| \leq \|f\|\|g\|$$

Linear Independance

A set of functions in L_2 are linearly independent if:

$$a_1 f_1(x) + a_2 f_2(x) + \cdots + a_n f_n(x) = 0$$

If and only if all the *a* coefficients are 0.

Grahm-Schmidt Orthogonalization

The Grahm-Schmidt technique that we discussed earlier still works with functions, and we can use it to form a set of linearly independent, orthogonal functions in L_2.

For a set of functions φ, we can make a set of orthogonal functions ψ that space the same space but are orthogonal to one another:

[Grahm-Schmidt Orthogonalization]

$$\psi_1 = \phi_1$$
$$\psi_i = \phi_i - \sum_{n=1}^{i-1} \frac{\langle \psi_n, \phi_i \rangle}{\langle \psi_n, \psi_n \rangle} \psi_n$$

Basis

The L_2 is an infinite-basis set, which means that any basis for the L_2 set will require an infinite number of basis functions. To prove that an infinite set of orthogonal functions is a basis for the L_2 space, we need to show that the null function is the only function in L_2 that is orthogonal to all the basis functions. If the null function is the only function that satisfies this relationship, then the set is a basis set for L_2.

By definition, we can express any function in L_2 as a linear sum of the basis elements. If we have basis elements φ, we can define any other function ψ as a linear sum:

$$\psi(x) = \sum_{n=1}^{\infty} a_n \phi_n(x)$$

We will explore this important result in the section on ../Fourier Series/.

Engineering Analysis/Banach and Hilbert Spaces

There are some special spaces known as **Banach** spaces, and **Hilbert** spaces.

Convergent Functions

Let's define the piece-wise function φ(x) as:

$$\phi_n(x) = \begin{cases} 0 & x \leq 0 \\ nx & 0 < x \leq \frac{1}{n} \\ 1 & \frac{1}{n} < x \end{cases}$$

We can see that as we set $n \to \infty$, this function becomes the unit step function. We can say that as n approaches infinity, that this function converges to the unit step function. Notice that this function only converges in the L_2 space, because the unit step function does not exist in the C space (it is not continuous).

Convergence

We can say that a function φ converges to a function φ^* if:

$$\lim_{n\to\infty} \|\phi_n - \phi^*\| = 0$$

We can call this sequences, and all such sequences that converge to a given function as n approaches infinity a **cauchy sequence**.

Complete Function Spaces

A function space is called complete if all sequences in that space converge to another function in that space.

Banach Space

A Banach Space is a complete normed function space.

Hilbert Space

A Hilbert Space is a Banach Space with respect to a norm induced by the scalar product. That is, if there is a scalar product in the space X, then we can say the norm is induced by the scalar product if we can write:

$$\|f\| = g(\langle f, f\rangle)$$

That is, that the norm can be written as a function of the scalar product. In the L_2 space, we can define the norm as:

$$\|f\| = \sqrt{\langle f, f\rangle}$$

If the scalar product space is a Banach Space, if the norm space is also a Banach space.

In a Hilbert Space, the Parallelogram rule holds for all members f and g in the function space:

$$\|f+g\|^2 + \|f-g\|^2 = 2\|f\|^2 + 2\|g\|^2$$

The L_2 space is a Hilbert Space. The C space, however, is not.

Engineering Analysis/Fourier Series

The L_2 space is an infinite function space, and therefore a linear combination of any infinite set of orthogonal functions can be used to represent any single member of the L_2 space. The decomposition of an L_2 function in terms of an infinite basis set is a technique known as the **Fourier Decomposition** of the function, and produces a result called the **Fourier Series**.

Fourier Basis

Let's consider a set of L_2 functions, ϕ , as follows:

$$\phi = \{1, \sin(\pi x), \cos(\pi x), \sin(2\pi x), \cos(2\pi x), \sin(3\pi x), \cos(3\pi x)...\}.$$

We can prove that over a range $[0, 2\pi]$, all of these functions are orthogonal:

$$\int_0^{2\pi} 1 \cdot \cos(n\pi x)dx = 0$$

$$\int_0^{2\pi} 1 \cdot \sin(n\pi x)dx = 0$$

$$\int_0^{2\pi} \sin(n\pi x)\sin(m\pi x)dx = 0, n \neq m$$

$$\int_0^{2\pi} \sin(n\pi x)\cos(m\pi x)dx = 0$$

$$\int_0^{2\pi} \cos(n\pi x)\cos(m\pi x)dx = 0, n \neq m$$

Because ϕ is as an infinite orthogonal set in L_2, ϕ is also a valid basis set in the L_2 space. Therefore, we can decompose any function in L_2 as the following sum:

[Classical Fourier Series]

$$\psi(x) = a_0(1) + \sum_{n=1}^{\infty} a_n \sin(n\pi x) + \sum_{m=1}^{\infty} b_m \cos(m\pi x)$$

However, the difficulty occurs when we need to calculate the a and b coefficients. We will show the method to do this below:

a_0: The Constant Term

Calculation of a_0 is the easiest, and therefore we will show how to calculate it first. We use the value of a_0 which minimizes the error in approximating $f(x)$ by the Fourier series.

First, define an error function, E, that is equal to the squared norm of the difference between the function f(x) and the infinite sum above:

$$E = \frac{1}{2}\int_0^{2\pi} \|f(x) - a_0(1) - \sum_{n=1}^{\infty} a_n \sin(n\pi x) - \sum_{m=1}^{\infty} b_m \cos(m\pi x)\|^2 dx$$

For ease, we will write all the basis functions as the set φ, described above:

$$\sum_{i=0}^{\infty} a_i\phi_i = a_0 + \sum_{n=1}^{\infty} a_n \sin(n\pi x) + \sum_{m=1}^{\infty} b_m \cos(m\pi x)$$

Combining the last two functions together, and writing the norm as an integral, we can say:

$$E = \frac{1}{2}\int_0^{2\pi} |\sum_{i=0}^{\infty} a_i\phi_i|^2 dx$$

We attempt to minimize this error function with respect to the constant term. To do this, we differentiate both sides with respect to a_0, and set the result to zero:

$$0 = \frac{\partial E}{\partial a_0} = \int_0^{2\pi} (f(x) - \sum_{i=0}^{\infty} a_i \phi_i(x))(-\phi_0(x))dx$$

The φ_0 term comes out of the sum because of the chain rule: it is the only term in the entire sum dependant on a_0. We can separate out the integral above as follows:

$$\int_0^{2\pi} (f(x) - \sum_{i=0}^{\infty} a_i \phi_i)(-\phi_0)dx = -\int_0^{2\pi} f(x)\phi_0(x)dx + a_0 \int_0^{2\pi} \phi_0(x)\phi_0(x)dx$$

All the other terms drop out of the infinite sum because they are all orthogonal to φ_0. Again, we can rewrite the above equation in terms of the scalar product:

$$0 = -\langle f(x), \phi_0(x) \rangle + a_0 \langle \phi_0(x), \phi_0(x) \rangle$$

And solving for a_0, we get our final result:

$$a_0 = \frac{\langle f(x), \phi_0(x) \rangle}{\langle \phi_0(x), \phi_0(x) \rangle}$$

Sin Coefficients

Using the above method, we can solve for the a_n coefficients of the sin terms:

$$a_n = \frac{\langle f(x), \sin(n\pi x) \rangle}{\langle \sin(n\pi x), \sin(n\pi x) \rangle}$$

Cos Coefficients

Also using the above method, we can solve for the b_n terms of the cos term.

$$b_n = \frac{\langle f(x), \cos(n\pi x) \rangle}{\langle \cos(n\pi x), \cos(n\pi x) \rangle}$$

Engineering Analysis/Arbitrary Basis Expansion

The classical Fourier series uses the following basis:

$$\phi(x) = 1, \sin(n\pi x), \cos(n\pi x), n = 1, 2, ...$$

However, we can generalize this concept to extend to any orthogonal basis set from the L_2 space.

We can say that if we have our orthogonal basis set that is composed of an infinite set of arbitrary, orthogonal L_2 functions:

$$\phi = \phi_1, \phi_2, \cdots,$$

We can define any L_2 function f(x) in terms of this basis set:

[Generalized Fourier Series]

$$f(x) = \sum_{n=1}^{\infty} a_n \phi_n(x)$$

Using the method from the previous chapter, we can solve for the coefficients as follows:

[Generalized Fourier Coefficient]

$$a_n = \frac{\langle f(x), \phi_n(x) \rangle}{\langle \phi_n(x), \phi_n(x) \rangle}$$

Engineering Analysis/Bessel Equation and Parseval Theorem

Bessel's equation relates the original function to the fourier coefficients a_n:

[Bessel's Equation]

$$\sum_{n=1}^{\infty} a_n^2 \leq \|f(x)\|^2$$

If the basis set is infinitely orthogonal, and if an infinite sum of the basis functions perfectly reproduces the function f(x), then the above equation will be an equality, known as Parseval's Theorem:

[Parseval's Theorem]

$$\sum_{n=1}^{\infty} a_n^2 = \|f(x)\|^2$$

Engineers may recognize this as a relationship between the energy of the signal, as represented in the time and frequency domains. However, parseval's rule applies not only to the classical Fourier series coefficients, but also to the generalized series coefficients as well.

Engineering Analysis/Multi-Dimensional Fourier Series

The concept of the fourier series can be expanded to include 2-dimensional and n-dimensional function decomposition as well. Let's say that we have a function in terms of independent variables x and y. We can decompose that function as a double-summation as follows:

$$f(x,y) = \sum_{i=1}^{\infty}\sum_{j=1}^{\infty} a_{ij}\phi_{ij}(x,y)$$

Where φ_{ij} is a 2-dimensional set of orthogonal basis functions. We can define the coefficients as:

$$a_{ij} = \frac{\langle f(x,y), \phi_{ij}(x,y)\rangle}{\langle \phi_{ij}(x,y), \phi_{ij}(x,y)\rangle}$$

This same concept can be expanded to include series with n-dimensions.

further reading

- Basic Physics of Nuclear Medicine/Fourier Methods discusses using 2D and 3D Fourier reconstruction to get images of the interior of the human body.
- Kevin Cowtan's Book of Fourier [1]: a book of pictorial 2-d Fourier Transforms.

References

[1] http://www.ysbl.york.ac.uk/~cowtan/fourier/fourier.html

Wavelet Analysis

Engineering Analysis/Wavelets

Wavelets are orthogonal basis functions that only exist for certain windows in time. This is in contrast to sinusoidal waves, which exist for all times t. A wavelet, because it is dependant on time, can be used as a basis function. A wavelet basis set gives rise to wavelet decomposition, which is a 2-variable decomposition of a 1-variable function. Wavelet analysis allows us to decompose a function in terms of time and frequency, while fourier decomposition only allows us to decompose a function in terms of frequency.

Mother Wavelet

If we have a basic wavelet function ψ(t), we can write a 2-dimensional function known as the **mother wavelet function** as such:

$$\psi_{jk} = 2^{j/2}\psi(2^j t - k)$$

Wavelet Series

If we have our mother wavelet function, we can write out a fourier-style series as a double-sum of all the wavelets:

$$f(t) = \sum_{j=0}^{\infty}\sum_{k=0}^{\infty} a_{jk}\psi_{jk}(t)$$

Scaling Function

Sometimes, we can add in an additional function, known as a **scaling function**:

$$f(t) = \sum_{i=0}^{\infty} c_i\phi_i + \sum_{j=0}^{\infty}\sum_{k=0}^{\infty} a_{jk}\psi_{jk}(t)$$

The idea is that the scaling function is larger than the wavelet functions, and occupies more time. In this case, the scaling function will show long-term changes in the signal, and the wavelet functions will show short-term changes in the signal.

Stochastic Processes

Engineering Analysis/Random Variables

Random Variables

A random variable is a variable that takes a random value at any particular point *t* in time. The properties of the random variable are known as the **distribution** of the random variable. We will denote random variables by the abbreviation "r.v.", or simply "rv". This is a common convention used in the literature concerning this subject.

Probability Function

The probability function, P[], will denote the probability of a particular occurrence happening. Here are some examples:

- $P[X < x]$, the probability that the random variable X has a value less than some variable *x*.
- $P[X = x]$, the probability that the random variable X has a value equal to some variable *x*.
- $P[X < x, Y > y]$, the probability that the random variable X has a value less than *x*, and the random variable Y has a value greater than *y*.

Example: Fair Coin

Consider the example that a fair coin is flipped. We will define X to be the random variable, and we will define "head" to be 1, and "tail" to be 0. What is the probability that the coin is a head?

$$P[X = 1] = 0.5$$

Example: Fair Dice

Consider now a fair 6-sided dice. X is the r.v., and the numerical value on the face of the die is the value that X can take. What is the probability that when the dice is rolled, the value is less than 4?

$$P[X < 4] = 0.5$$

What is the probability that the value will be even?

$$P[X \text{ is even}] = 0.5$$

Notation

We will typically write random variables as upper-case letters, such as Z, X, Y, etc. Lower-case letters will be used to denote variables that are related with the random variables. For instance, we will use "x" as a variable that is related to "X", the random variable.

When we are using random variables in conjunction with matrices, we will use the following conventions:

1. Random variables, and random vectors or matrices will be denoted with letters from the end of the alphabet, such as W, X, Y, and Z. Also, Θ and Ω will be used as a random variables, especially when we talk about random frequencies.
2. A random matrix or vector, will be denoted with a capital letter. The entries in that random vector or matrix will be denoted with capital letters and subscripts. These matrices will also use letters from the end of the alphabet, or the Greek letters Θ and Ω.

3. A regular coefficient vector or matrix that is not random will use a capital matrix from the beginning of the alphabet, such as A, B, C, or D.
4. Special vectors or matrices that are derived from random variables, such as correlation matrices, or covariance matrices, will use capital letters from the middle of the alphabet, such as K, M, N, P, or Q.

Any other variables or notations will be explained in the context of the page where it appears.

Conditional Probability

A **conditional probability** is the probability measure of one event happening given that another event already has happened. For instance, what are the odds that your computer system will suddenly break while you are reading this page?

$$P[\text{computer breaks}] = \text{small}$$

The odds that your computer will suddenly stop working is very small. However, what are the odds that your computer will break given that it just got struck by lightning?

$$P[\text{computer breaks}|\text{struck by lightning}] = \text{large}$$

The vertical bar separates the things that haven't happened yet (the *a priori* probabilities, on the left) from the things that have already happened and might affect our outcome (the *a posteriori* probabilities, on the right). As another example, what are the odds that a dice rolled will be a 2, assuming that we know the number is less than 4?

$$P[X = 2|X < 4] = 0.33$$

If X is less than 4, we know it can only be one of the values 1, 2, or 3. Or another example, what if a person asks you "I'm thinking of a number between 1 and 10", what are your odds of guessing the right number?

$$P[X = x|0 < X < 10] = 0.1$$

Where *x* is the correct number that you are trying to guess.

Engineering Analysis/Probability Functions

Probability Density Function

The **probability density function**, or **pdf** of a random variable is the function defined by:

$$f_X(x) = P[X = x]$$

Remember here that X is the random variable, and x is a related variable (but is not random). The subscript X on f_X denotes that this is the pdf for the X variable.

pdf's follow a few simple rules:

1. The pdf is always non-negative.
2. The area under the pdf curve is 1.

$$\int_{-\infty}^{\infty} f_X(x)dx = 1$$

Cumulative Distribution Function

The **cumulative distribution function**, (CDF), is also known as the **Probability Distribution Function**, (PDF). to reduce confusion with the pdf of a random variable, we will use the acronym CDF to denote this function. The CDF of a random variable is the function defined by:

$$F_X(x) = P[X \leq x]$$

The CDF and the pdf of a random variable are related:

$$f_X(x) = \frac{dF_X(x)}{dx}$$

$$F_X(x) = \int f_X(x)dx$$

The CDF is the function corresponding to the probability that a given value *x* is less than the value of the random variable X. The CDF is a non-decreasing function, and is always non-negative.

Example: X between two bounds

To determine whether our random variable X lies between two bounds, [a, b], we can take the CDF functions:

$$P[a \leq X \leq b] = F_X(b) - F_X(a)$$

Engineering Analysis/Distributions

Distributions

There are a number of common distributions, that are used in conjunction with random variables.

Uniform Distribution

The uniform distribution is one of the easiest distributions to analyze. Also, uniform distributions of random numbers are easy to generate on computers, so they are typically used in computer software.

$$f_X(x) = \begin{cases} \frac{1}{b-a} & \text{if } a < x < b \\ 0 & \text{otherwise} \end{cases}$$

$$F_X(x) = \begin{cases} 0 & \text{if } x < a \\ \frac{x}{b-a} & \text{if } a < x < b \\ 1 & \text{if } x \geq b \end{cases}$$

Gaussian Distribution

The **gaussian distribution**, or the "normal distribution" is one of the most common random distributions. A gaussian random variable is typically called a "normal" random variable.

$$f_X(x) = \mathcal{N}(\mu, \sigma^2) = \frac{1}{\sqrt{2\pi\sigma^2}} e^{-\frac{(x-\mu)^2}{\sigma^2}}$$

Where μ is the **mean** of the function, and σ^2 is the variance of the function. we will discuss both these terms later.

Engineering Analysis/Expectation and Entropy

Expectation

The expectation operator of a random variable is defined as:

$$E[x] = \int_{-\infty}^{\infty} x f_X(x) dx$$

This operator is very useful, and we can use it to derive the moments of the random variable.

Moments

A moment is a value that contains some information about the random variable. The n-moment of a random variable is defined as:

$$E[x^n] = \int_{-\infty}^{\infty} x^n f_X(x) dx$$

Mean

The **mean** value, or the "average value" of a random variable is defined as the first moment of the random variable:

$$E[x] = \mu_X = \int_{-\infty}^{\infty} x f_X(x) dx$$

We will use the Greek letter μ to denote the mean of a random variable.

Central Moments

A central moment is similar to a moment, but it is also dependant on the mean of the random variable:

$$E[(x - \mu_X)^n] = \int_{-\infty}^{\infty} (x - \mu_X)^n f_X(x) dx$$

The first central moment is always zero.

Variance

The variance of a random variable is defined as the second central moment:

$$E[(x - \mu_X)^2] = \sigma^2$$

The square-root of the variance, σ, is known as the **standard-deviation** of the random variable

Mean and Variance

the mean and variance of a random variable can be related by:

$$\sigma^2 = \mu^2 + E[x^2]$$

This is an important function, and we will use it later.

Entropy

the **entropy** of a random variable X is defined as:

$$H[X] = E\left[\frac{1}{p(X)}\right]$$

Engineering Analysis/SISO Transformations

Let's say that we have a random variable X that is the input into a given system. The system output, Y is then also a random variable that is related to the input X by the response of the system. In other words, we can say that:

$$Y = g(X)$$

Where *g* is the mathematical relationship between the system input and the system output.

To discover information about Y, we can use the information we know about the r.v. X, and the relationship *g*:

$$f_Y(y) = \sum_i \frac{f_X(y)}{\left|\frac{dy}{dx}\right|}$$

Where x_i are the roots of *g*.

Engineering Analysis/MISO Transformations

Consider now a system with two inputs, both of which are random (or pseudorandom, in the case of non-deterministic data). For instance, let's consider a system with the following inputs and outputs:

- X: non-deterministic data input
- Y: disruptive noise
- Z: System output

Our system satisfies the following mathematical relationship:

$$Z = g(X, Y)$$

Where *g* is the mathematical relationship between the system input, the disruptive noise, and the system output. By knowing information about the distributions of X and Y, we can determine the distribution of Z.

Engineering Analysis/Correlation

Independance

Two random variables are called **independent** if changes in one do not affect, and are not affected by, changes in the other.

Correlation

Two random variables are said to have **correlation** if they take the same values, or similar values, at the same point in time. Independence implies that two random variables will be uncorrelated, but two random variables being uncorrelated does not imply that they are independent.

Engineering Analysis/Random Vectors

Many of the concepts that we have learned so far have been dealing with random variables. However, these concepts can all be translated to deal with vectors of random numbers. A random vector X contains N elements, X_i, each of which is a distinct random variable. The individual elements in a random vector may or may not be correlated or dependent on one another.

$$X = \begin{bmatrix} X_1 \\ X_2 \\ \vdots \\ X_N \end{bmatrix}$$

Expectation

The expectation of a random vector is a vector of the expectation values of each element of the vector. For instance:

$$E[X] = \begin{bmatrix} E[X_1] \\ E[X_2] \\ \vdots \\ E[X_N] \end{bmatrix}$$

Using this definition, the mean vector of random vector X, denoted μX is the vector composed of the means of all the individual elements of X:

$$\mu_X = \begin{bmatrix} \mu_{X_1} \\ \mu_{X_2} \\ \vdots \\ \mu_{X_N} \end{bmatrix}$$

Correlation Matrix

The correlation matrix of a random vector X is defined as:

$$R_X = E[XX^T]$$

Where each element of the correlation matrix corresponds to the correlation between the row element of X, and the column element of X^T. The correlation matrix is a real-symmetric matrix. If the off-diagonal elements of the correlation matrix are all zero, the random vector is said to be uncorrelated. If the R matrix is an identity matrix, the random vector is said to be "white". For instance, "white noise" is uncorrelated, and each element of the vector has an equal correlation value.

Matrix Diagonalization

As discussed earlier, we can diagonalize a matrix by constructing the V matrix from the eigenvectors of that matrix. If X is our non-diagonal matrix, we can create a diagonal matrix D by:

$$D = V^{-1}XV$$

If the X matrix is real symmetric (as is always the case with the correlation matrix), we can simplify this to be:

$$D = V^TXV$$

Whitening

A matrix can be whitened by constructing a matrix W that contains the inverse squareroots of the eigenvalues of X on the diagonal:

$$W = \begin{bmatrix} \frac{1}{\sqrt{\lambda_1}} & \cdots & \\ 0 & \frac{1}{\sqrt{\lambda_2}} \cdots & \\ \vdots & \vdots & \ddots \end{bmatrix}$$

Using this W matrix, we can convert X into the identity matrix:

$$I = W^TV^TXVW$$

Simultaneous Diagonalization

If we have two matrices, X and Y, we can construct a matrix A that will satisfy the following relationships:

$$A^TXA = I$$
$$A^TYA = D$$

Where I is an identity matrix, and D is a diagonal matrix. This process is known as simultaneous diagonalization. If we have the V and W matrices described above such that

$$I = W^TV^TXVW,$$

We can then construct the B matrix by applying this same transformation to the Y matrix:

$$W^TV^TYVW = B$$

We can combine the eigenvalues of B into a transformation matrix Z such that:

$$Z^TBZ = D$$

We can then define our A matrix as:

$$A = VWZ$$
$$A^T = Z^TW^TV^T$$

This A matrix will satisfy the simultaneous diagonalization procedure, outlined above.

Covariance Matrix

The Covariance Matrix of two random vectors, X and Y, is defined as:

$$Q_X = E[(X - \mu_X)(Y - \mu_Y)^T] = E[(Y - \mu_Y)(X - \mu_X)^T]$$

Where each element of the covariance matrix expresses the variance relationship between the row element of X, and the column element of Y. The covariance matrix is real symmetric.

We can relate the correlation matrix and the covariance matrix through the following formula:

$$R = Q + \mu_X \mu_X^T$$

Cumulative Distribution Function

An N-vector X has a cumulative distribution function F_x of N variables that is defined as:

$$F_X(X) = P[X \leq x] = P[X_1 \leq x_1, X_2 \leq x_2, \cdots, X_N \leq x_N]$$

Probability Density Function

The probability density function of a random vector can be defined in terms of the N^{th} partial derivative of the cumulative distribution function:

$$f_X(X) = \frac{\partial^N F_X(X)}{\partial X_1 \partial X_2 \cdots \partial X_N}$$

If we know the density function, we can find the mean of the i^{th} element of X using N-1 integrations:

$$\mu_{X_i} = \int_{-\infty}^{\infty} \int_{-\infty}^{\infty} \cdots \int_{-\infty}^{\infty} x_i f_X(x) dx_1 dx_2 \cdots dx_n$$

Optimization and Minimization

Engineering Analysis/Optimization

Optimization

Optimization is an important concept in engineering. Finding any solution to a problem is not nearly as good as finding the one "optimal solution" to the problem. Optimization problems are typically reformatted so they become **minimization problems**, which are well-studied problems in the field of mathematics.

Typically, when optimizing a system, the costs and benefits of that system are arranged into a **cost function**. It is the engineers job then to minimize this cost function (and thereby minimize the cost of the system). It is worth noting at this point that the word "cost" can have multiple meanings, depending on the particular problem. For instance, cost can refer to the actual monetary cost of a system (number of computer units to host a website, amount of cable needed to connect Philadelphia and New York), the delay of the system (loading time for a website, transmission delay for a communication network), the reliability of the system (number of dropped calls in a cellphone network, average lifetime of a car transmission), or any other types of factors that reduce the effectiveness and efficiency of the system.

Because optimization typically becomes a mathematical minimization problem, we are going to discuss minimization here.

Minimization

Minimization is the act of finding the numerically lowest point in a given function, or in a particular range of a given function. Students of mathematics and calculus may remember using the derivative of a function to find the maxima and minima of a function. If we have a function $f(x)$, we can find the maxima, minima, or saddle-points (points where the function has zero slope, but is not a maxima or minima) by solving for x in the following equation:

$$\frac{df(x)}{dx} = 0$$

In other words, we are looking for the roots of the derivative of the function f plus those points where f has a corner. Once we have the so called critical points of the function (if any), we can test them to see if they are relatively high (maxima), or relatively low (minima). Some words to remember in this context are:

Global Minima

: A global minimum of a function is the lowest value of that function anywhere. If the domain of the function is restricted, say $A < x < B$, then the minima can also occur at the boundary, here A or B.

Local Minima

: A local minimum of a function is the lowest value of that function within a small range. A value can thus be a local minimum even though there are smaller function values, but not in a small neighborhood.

Unconstrained Minimization

Unconstrained Minimization refers to the minimization of the given function without having to worry about any other rules or caveats. **Constrained Minimization**, on the other hand, refers to minimization problems where other relations called *constraints* must be satisfied at the same time.

Beside the method above (where we take the derivative of the function and set that equal to zero), there are several numerical methods that we can use to find the minima of a function. For these methods there are useful computational tools such as **Matlab**.

Hessian Matrix

The function has a local minima at a point x if the Hessian matrix $H(x)$ is positive definite:

$$H(x) = \frac{\partial^2 f(x)}{\partial x^2}$$

Where x is a vector of all the independant variables of the function. If x is a scalar variable, the hessian matrix reduces to the second derivative of the function f.

Newton-Raphson Method

The **Newton-Raphson Method** of computing the minima of a function f uses an iterative computation. We can define the sequence:

$$x^{n+1} = x^n - \frac{f'(x)}{f''(x)}$$

Where

$$f'(x) = \frac{df(x)}{dx}$$

$$f''(x) = \frac{d^2 f(x)}{dx^2}$$

As we repeat the above computation, plugging in consecutive values for n, our solution will converge on the true solution. However, this process will take infinitely many iterations to converge, but if an approximation of the true solution will suffices, you can stop after only few iterations, because the sequence converges rather quickly (quadratic).

Steepest Descent Method

The Newton-Raphson method can be tricky because it relies on the second derivative of the function f, and this can oftentimes be difficult (if not impossible) to accurately calculate. The **Steepest Descent Method**, however, does not require the second derivative, but it does require the selection of an appropriate scalar quantity ε, which cannot be chosen arbitrarily (but which can also not be calculated using a set formula). The Steepest Descent method is defined by the following iterative computation:

$$x^{n+1} = x^n - \epsilon \frac{df(x)}{dx}$$

Where epsilon needs to be sufficiently small. If epsilon is too large, the iteration may diverge. If this happens, a new epsilon value needs to be chosen, and the process needs to be repeated.

Constrained Minimization

Constrained Minimization' is the process of finding the minimum value of a function under a certain number of additional rules called constraints. For instance, we could say "Find the minium value of $f(x)$, but $g(x)$ must equal 10". These kinds of problems are more difficult, but the **Khun-Tucker** theorem, and also the **Karush-Khun-Tucker** theorem help to solve them.

There are two different types of constraints: equality constraints and inequality constraints. We will consider them individually, and then mixed constraints.

Equality Constraints

The Khun-Tucker Theorem is a method for minimizing a function $f(x)$ under the equality constraint $g(x)$. The theorem reads as follows:

Given the cost function f, and an equality constraint g in the following form:

$$g(x) = 0,$$

Then we can convert this problem into an unconstrained minimization problem by constructing the **Lagrangian** function of f and g:

$$L(x) = f(x) + \langle \Lambda, g(x) \rangle$$

Where Λ is the *lagrange multiplier*, and < , > denotes the **scalar product** of the vector space R^n (where n is the number of equality constraints). We will discuss scalar products in more detail later. If we differentiate this equation with respect to x, we can find the minimum of this whole function $L(x,\Lambda)$, and that will be the minimum of our function f.

$$\frac{df(x)}{dx} + \left\langle \Lambda, \frac{dg(x)}{dx} \right\rangle = 0$$

$$g(x) = 0$$

This is a set of $n+k$ equations with $n+k$ unknown variables (n Λs and k xs).

Inequality Constraints

Similar to the method above, let us say that we have a cost function f, and an inequality constraint in the following form:

$$g(x) \leq 0$$

Then we can take the Lagrangian of this again:

$$L(x) = f(x) + \langle \Lambda, g(x) \rangle$$

But we now must use the following three equations/ inequalities in determining our solution:

$$\frac{df}{dx} = 0$$

$$\langle \Lambda, g(x) \rangle = 0$$

$$\Lambda \geq 0$$

These last second equation can be interpreted in the following way:

if $g(x) < 0$, then $\Lambda = 0$

if $g(x) \leq 0$, then $\Lambda \geq 0$

Using these two additional equations/ inequalities, we can solve in a similar manner as above.

Mixed Constraints

If we have a set of equality and inequality constraints

$$g(x) = 0$$
$$h(x) \leq 0$$

we can combine them into a single Lagrangian with two additional conditions:

$$L(x) = f(x) + \langle \Lambda, g(x) \rangle + \langle \mu, h(x) \rangle$$
$$g(x) = 0$$
$$\langle \mu, h(x) \rangle = 0$$
$$\mu \geq 0$$

Infinite Dimensional Minimization

The above methods work well if the variables involved in the analysis are finite-dimensional vectors, like those in the $\mathbf{R}^N$. However, when we are trying to minimize something that is more complex than a vector, i.e. a function we need the following concept. We consider functions that live in a subspace of $L_2(\mathbf{R}^N)$, which is an infinite-dimensional vector space. We will define the term **functional** as follows:

Functional

> A functional is a map that takes one or more functions as arguments, and which returns a scalar value.

Let us say that we consider functions x of time t (N=1). Suppose further we have a fixed function f in two variables. With that function, we can associate a cost functional J:

$$J[x] = \int_a^b f(x,t)dt$$

Where we are explicitly taking account of t in the definition of f. To minimize this function, like all minimization problems, we need to take the derivative of the function, and set the derivative to zero. However, we need slightly more sophisticated version of derivative, because x is a function. This is where the **Gateaux Derivative** enters the field.

Gateaux Derivative

We can define the **Gateaux Derivative** in terms of the following limit:

$$\delta F(x,h) = \lim_{\epsilon \to 0} \frac{1}{\epsilon}[F(x+\epsilon h) - F(x)]$$

Which is similar to the classical definition of the derivative in the direction h. In plain words, we took the derivative of F with respect to x in the direction of h. h is an arbitrary function of time, in the same space as x (here we are talking about the space L_2). Analog to the one-dimensional case a function is differentiable at x iff the above limit exists. We can use the Gateaux derivative to find the minimization of our functional above.

Euler-Lagrange Equation

We will now use the Gateaux derivative, discussed above, to find the minimizer of the following types of function:

$$J(x(t)) = \int_a^b f(x(t), x'(t), t)dt$$

We thus have to find the solutions to the equation:

$$\delta J(x) = 0$$

The solution is the **Euler-Lagrange Equation**:

$$\frac{\partial f}{\partial x} - \frac{d}{dt}\frac{\partial f}{\partial x'} = 0$$

The partial derivatives are done in an ordinary way ignoring the fact that *x* is a function of *t*. Solutions to this equation are either maxima, minima, or saddle points of the cost functional *J*.

Example: Shortest Distance

We've heard colloquially that the shortest distance between two points is a straight line. We can use the Euler-Lagrange equation to prove this rule.

If we have two points in $\mathbf{R}^2$, *a*, and *b*, we would like to find the minimum curve *(x,y(x))* that joins these two points. Line element *ds* reads:

$$ds = \sqrt{dx^2 + dy^2}$$

Our function that we are trying to minimize then is defined as:

$$J[y] = \int_a^b ds$$

or:

$$J[y] = \int_a^b \sqrt{1 + \left(\frac{dy}{dx}\right)^2}\, dx$$

We can take the Gateaux derivative of the function J and set it equal to zero to find the minimum function between these two points. Denoting the square root as *f*, we get

$$0 = \frac{\partial f}{\partial y} - \frac{d}{dx}\left(\frac{\partial f}{\partial y'}\right) = y'' \frac{1}{(1 + y'^2)^{3/2}} \ .$$

Knowing that the line element will be finite this boils down to the equation

$$\frac{d^2 y}{dx^2} = 0$$

with the well known solution

$$y(x) = mx + n = \frac{b_y - a_y}{b_x - a_x}(x - a_x) + a_y \ .$$

Resources and Licensing

Engineering Analysis/Resources

Wikibooks

- Algebra
- Linear Algebra
- Calculus
- Ordinary Differential Equations

Article Sources and Contributors

Wikibooks:Collections Preface *Source*: http://en.wikibooks.org/w/index.php?oldid=2347851 *Contributors*: Adrignola, Jomegat, Magesha, Martin Kraus, Mike.lifeguard, RobinH, Whiteknight

Engineering Analysis/Vector Spaces *Source*: http://en.wikibooks.org/w/index.php?oldid=2247372 *Contributors*: Herraotic, Whiteknight, 2 anonymous edits

Engineering Analysis/Vector Basics *Source*: http://en.wikibooks.org/w/index.php?oldid=2290418 *Contributors*: Avicennasis, Mewilhel, Recent Runes, Whiteknight, 3 anonymous edits

Engineering Analysis/Linear Independence and Basis *Source*: http://en.wikibooks.org/w/index.php?oldid=2065502 *Contributors*: Avicennasis, DavidCary, Whiteknight, 2 anonymous edits

Engineering Analysis/Linear Transformations *Source*: http://en.wikibooks.org/w/index.php?oldid=2065503 *Contributors*: Avicennasis, Whiteknight, 1 anonymous edits

Engineering Analysis/Minimization *Source*: http://en.wikibooks.org/w/index.php?oldid=615354 *Contributors*: Whiteknight

Engineering Analysis/Projections *Source*: http://en.wikibooks.org/w/index.php?oldid=615343 *Contributors*: Whiteknight

Engineering Analysis/Linear Spaces *Source*: http://en.wikibooks.org/w/index.php?oldid=834553 *Contributors*: Whiteknight

Engineering Analysis/Matrices *Source*: http://en.wikibooks.org/w/index.php?oldid=1019144 *Contributors*: Whiteknight, 2 anonymous edits

Engineering Analysis/Matrix Forms *Source*: http://en.wikibooks.org/w/index.php?oldid=961506 *Contributors*: Whiteknight, 2 anonymous edits

Engineering Analysis/Quadratic Forms *Source*: http://en.wikibooks.org/w/index.php?oldid=2032674 *Contributors*: Reyk, Whiteknight, 1 anonymous edits

Engineering Analysis/Eigenvalues and Eigenvectors *Source*: http://en.wikibooks.org/w/index.php?oldid=2065497 *Contributors*: Avicennasis, Whiteknight

Engineering Analysis/Diagonalization *Source*: http://en.wikibooks.org/w/index.php?oldid=2065496 *Contributors*: Avicennasis, Whiteknight, Xz64

Engineering Analysis/Spectral Decomposition *Source*: http://en.wikibooks.org/w/index.php?oldid=2065509 *Contributors*: Avicennasis, Whiteknight

Engineering Analysis/Error Estimation *Source*: http://en.wikibooks.org/w/index.php?oldid=2065498 *Contributors*: Avicennasis, MelchiorG, Whiteknight, 2 anonymous edits

Engineering Analysis/Matrix Functions *Source*: http://en.wikibooks.org/w/index.php?oldid=2065504 *Contributors*: Avicennasis, Webaware, Whiteknight, 2 anonymous edits

Engineering Analysis/Cayley Hamilton Theorem *Source*: http://en.wikibooks.org/w/index.php?oldid=834572 *Contributors*: Whiteknight

Engineering Analysis/Matrix Exponentials *Source*: http://en.wikibooks.org/w/index.php?oldid=1713024 *Contributors*: DavidCary, Whiteknight, 1 anonymous edits

Engineering Analysis/Lyapunov Equation *Source*: http://en.wikibooks.org/w/index.php?oldid=834579 *Contributors*: Whiteknight

Engineering Analysis/Function Spaces *Source*: http://en.wikibooks.org/w/index.php?oldid=834584 *Contributors*: Whiteknight

Engineering Analysis/L2 Space *Source*: http://en.wikibooks.org/w/index.php?oldid=2065500 *Contributors*: Avicennasis, Whiteknight, 7 anonymous edits

Engineering Analysis/Banach and Hilbert Spaces *Source*: http://en.wikibooks.org/w/index.php?oldid=1208660 *Contributors*: Jake Wasdin, Nostraticispeak, Whiteknight, 3 anonymous edits

Engineering Analysis/Fourier Series *Source*: http://en.wikibooks.org/w/index.php?oldid=1944623 *Contributors*: Whiteknight, 5 anonymous edits

Engineering Analysis/Arbitrary Basis Expansion *Source*: http://en.wikibooks.org/w/index.php?oldid=615383 *Contributors*: Whiteknight

Engineering Analysis/Bessel Equation and Parseval Theorem *Source*: http://en.wikibooks.org/w/index.php?oldid=606804 *Contributors*: Whiteknight

Engineering Analysis/Multi-Dimensional Fourier Series *Source*: http://en.wikibooks.org/w/index.php?oldid=2065505 *Contributors*: Avicennasis, DavidCary, Whiteknight

Engineering Analysis/Wavelets *Source*: http://en.wikibooks.org/w/index.php?oldid=2065511 *Contributors*: Avicennasis, Whiteknight

Engineering Analysis/Random Variables *Source*: http://en.wikibooks.org/w/index.php?oldid=2255024 *Contributors*: Avicennasis, Ugujjwal, Van der Hoorn, Whiteknight, 3 anonymous edits

Engineering Analysis/Probability Functions *Source*: http://en.wikibooks.org/w/index.php?oldid=2065506 *Contributors*: Avicennasis, Whiteknight

Engineering Analysis/Distributions *Source*: http://en.wikibooks.org/w/index.php?oldid=645437 *Contributors*: Whiteknight

Engineering Analysis/Expectation and Entropy *Source*: http://en.wikibooks.org/w/index.php?oldid=2065499 *Contributors*: Adrignola, Avicennasis, Whiteknight, 1 anonymous edits

Engineering Analysis/SISO Transformations *Source*: http://en.wikibooks.org/w/index.php?oldid=648477 *Contributors*: Whiteknight

Engineering Analysis/MISO Transformations *Source*: http://en.wikibooks.org/w/index.php?oldid=648656 *Contributors*: Whiteknight

Engineering Analysis/Correlation *Source*: http://en.wikibooks.org/w/index.php?oldid=2065495 *Contributors*: Avicennasis, Whiteknight

Engineering Analysis/Random Vectors *Source*: http://en.wikibooks.org/w/index.php?oldid=2065508 *Contributors*: Avicennasis, Whiteknight

Engineering Analysis/Optimization *Source*: http://en.wikibooks.org/w/index.php?oldid=1944804 *Contributors*: MelchiorG, Whiteknight, 6 anonymous edits

Engineering Analysis/Resources *Source*: http://en.wikibooks.org/w/index.php?oldid=2362778 *Contributors*: Recent Runes, Whiteknight

Image Sources, Licenses and Contributors

Image:Wikibooks-logo-en-noslogan.svg *Source*: http://en.wikibooks.org/w/index.php?title=File:Wikibooks-logo-en-noslogan.svg *License*: logo *Contributors*: User:Bastique, User:Ramac et al.

License