

Polynomial L^2 Approximation

Relating Orthonormal Polynomials, Gram–Schmidt Orthonormalization, QR Factorization, Normal Equations and Vandermonde and Hilbert Matrices

Gottlob Gienger

This didactic synthesis compares three solution methods for polynomial L^2 approximation and systematically presents their common characteristics and their close interrelations:

1. Classical Gram–Schmidt orthonormalization and Fourier approximation in $L^2(a, b)$
2. Linear least-squares solution via QR factorization on an equally spaced grid in $[a, b]$
3. Linear least-squares solution via the normal equations method in $L^2(a, b)$ and on an equally spaced grid in $[a, b]$

The first two methods are linear least-squares systems with Vandermonde matrices V ; the normal equations contain matrices of Hilbert type $H = V^T V$. The solutions on equally spaced grids in $[a, b]$ converge to the solutions in $L^2(a, b)$. All solution characteristics and their relations are illustrated by symbolic or numeric examples and graphs.

■ Definitions and Notation

Let $-\infty < a < b < \infty$. Consider the Hilbert space of real-valued square integrable functions $L^2([a, b])$ (or $L^2(a, b)$, for short), equipped with Lebesgue measure and scalar product

$$\langle f, g \rangle_{[a,b]} := \int_a^b f(x) g(x) dx \quad (1)$$

and the corresponding L^2 -norm

$$\|f\|_{2,[a,b]} := \sqrt{\langle f, f \rangle_{[a,b]}}.$$

L^2 scalar products can be approximated by scalar products on discrete grids in $[a, b]$ based on Riemann sums and similarly for norms.

Partition the finite interval $[a, b]$ into $m \in \mathbb{N}$ subintervals by the points

$$a = \xi_0 < \xi_1 < \dots < \xi_m = b$$

and set

$$\Xi := \{\xi_i, i = 0, 1, \dots, m\}.$$

Suppose that $f(x)$ is a bounded function on $[a, b]$. Let x_i be any point in the subinterval $\xi_{i-1} \leq x_i \leq \xi_i$ and define the grid

$$X := \{x_i, i = 1, \dots, m\}.$$

The *Riemann sums* on the partition Ξ and grid X are defined by

$$R(f, \Xi, X) := \sum_{i=1}^m (\xi_i - \xi_{i-1}) f(x_i), \quad (2)$$

$$\underline{R}(f, \Xi) := \sum_{i=1}^m (\xi_i - \xi_{i-1}) f(\xi_{i-1}), \quad (3)$$

$$\overline{R}(f, \Xi) := \sum_{i=1}^m (\xi_i - \xi_{i-1}) f(\xi_i), \quad (4)$$

$$M(f, \Xi) := \sum_{i=1}^m (\xi_i - \xi_{i-1}) f\left(\frac{\xi_i + \xi_{i-1}}{2}\right). \quad (5)$$

Equation (3) is called the left-hand Riemann sum, (4) the right-hand Riemann sum and (5) the (composite) midpoint rule.

For an equally spaced partition, the step size is

$$h := \frac{b-a}{m} = \xi_i - \xi_{i-1}, \quad i = 1, \dots, m.$$

The $m+1$ equally spaced partition points are

$$\xi_i := a + h i, \quad i = 0, 1, \dots, m, \quad \xi_0 = a, \quad \xi_m = b,$$

and the equally spaced grid of length m (excluding the endpoint b) is

$$\text{grid}(a, b, m) := \left\{ a + \frac{b-a}{m} i, i = 0, 1, \dots, m-1 \right\}.$$

It is also possible to use the grid points or grid shifted by an amount $s h$, where $0 \leq s \leq 1$ so that $0 \leq s h \leq h$, as

$$x_{i+s} := x_i + s h, \quad i = 0, 1, \dots, m-1.$$

Let

$$G = \text{grid}(a, b, m, s) := \text{grid}(a, b, m) + s h. \quad (6)$$

For equally spaced grids, the Riemann sums simplify to

$$R(f, G) = h \sum_{i=0}^{m-1} f(x_{i+s}). \quad (7)$$

Setting $s = 0$, $s = 1/2$ and $s = 1$ gives the left-hand Riemann sum, the composite mid-point rule and the right-hand Riemann sum, respectively. The error of the Riemann sums is defined as

$$E(f, G) := \int_a^b f(x) dx - R(f, G).$$

The set of continuous real-valued functions $C[a, b]$ forms a dense subspace of $L^2(a, b)$, [1], Theorem (13.21). For $f, g \in C[a, b]$, the restrictions $f|_G$ and $g|_G$ to this grid are well-defined. Define the m -dimensional scalar product on this grid:

$$\langle f, g \rangle_G := R(f g, G) = h \sum_{i=0}^{m-1} f(x_{i+s}) g(x_{i+s}). \quad (8)$$

The m -dimensional 2-norm is

$$\|f\|_{2,G} := \sqrt{\langle f, f \rangle_G}.$$

The factor $h = (b - a) / m$ ensures that the norms of constant functions agree:

$$\langle 1, 1 \rangle_{[a,b]} = \langle 1, 1 \rangle_G = b - a.$$

Denote the linear space of polynomials with real coefficients of degree at most $n - 1$ by Π_{n-1} and define the polynomial $p \in \Pi_{n-1}$ by

$$p(c_0, c_1, \dots, c_{n-1}, x) := \sum_{j=0}^{n-1} c_j x^j.$$

The polynomial can be written as a scalar product (or dot product) of two n -tuples, the monomials up to degree $n - 1$ and the n coefficients:

$$p(c_0, c_1, \dots, c_{n-1}, x) = (1, x, \dots, x^{n-1}) \cdot (c_0, c_1, \dots, c_{n-1}).$$

Introducing the $1 \times n$ *Vandermonde matrix*

$$\begin{aligned} V(n, x) &: \mathbb{R}^n \rightarrow \Pi_{n-1}, \\ V(n, x) &:= (1 \ x \ \dots \ x^{n-1}), \end{aligned} \quad (9)$$

every polynomial of degree $n - 1$ can be written as the product of a matrix and a vector as

$$V(n, x) c = (1 \ x \ \dots \ x^{n-1}) \begin{pmatrix} c_0 \\ c_1 \\ \vdots \\ c_{n-1} \end{pmatrix} = \sum_{j=0}^{n-1} c_j x^j = p(c_0, c_1, \dots, c_{n-1}, x).$$

The product of a $1 \times n$ matrix and an n -vector is regarded as a 1-vector, not a scalar, as in Mathematica.

Restricting the Vandermonde matrix to the interval $[a, b]$ gives an operator mapping into $L^2(a, b)$:

$$\begin{aligned} V(a, b, n, x) &: \mathbb{R}^n \rightarrow L^2(a, b), \\ V(a, b, n, x) &:= V(n, x|_{[a,b]}). \end{aligned} \quad (10)$$

Whereas $V(n, x)$ is an unbounded operator, $V(a, b, n, x)$ is a bounded operator with respect to the 2-norms on \mathbb{R}^n and $L^2(a, b)$.

The polynomial p approximates f in the L^2 -norm, as measured by

$$\min_{c \in \mathbb{R}^n} \|p(c_0, c_1, \dots, c_{n-1}, x) - f(x)\|_{2,[a,b]}. \quad (11)$$

In matrix-vector notation, this constitutes a linear least-squares problem for the coefficients $(c_0, c_1, \dots, c_{n-1})$:

$$\min_{c \in \mathbb{R}^n} \|V(a, b, n, x) c - f(x)\|_{2,[a,b]}, \quad (12)$$

where

$$c := \begin{pmatrix} c_0 \\ c_1 \\ \vdots \\ c_{n-1} \end{pmatrix} \in \mathbb{R}^n.$$

Now take a discrete grid

$$X := \{x_i, i = 1, \dots, m\}$$

and sample f on this discrete grid:

$$f(X) = \{f(x_i), i = 1, \dots, m\}.$$

The polynomial p of degree $n - 1$ approximates f in the 2-norm on this grid as measured by

$$\min_{c \in \mathbb{R}^n} \|p(c_0, c_1, \dots, c_{n-1}, X) - f(X)\|_2. \quad (13)$$

In matrix-vector notation, this constitutes a linear least-squares problem for the coefficients $(c_0, c_1, \dots, c_{n-1})$:

$$\min_{c \in \mathbb{R}^n} \|V(n, X) c - f(X)\|_2, \quad (14)$$

where

$$V(n, X) := [1, X, \dots, X^{n-1}] = \begin{pmatrix} 1 & x_1 & \dots & x_1^{n-1} \\ 1 & x_2 & \dots & x_2^{n-1} \\ \vdots & \vdots & \ddots & \vdots \\ 1 & x_m & \dots & x_m^{n-1} \end{pmatrix} \in \mathbb{R}^{m \times n}. \quad (15)$$

A rectangular or square matrix of this form is called a *Vandermonde matrix*.

Let $\mathcal{H}_1, \mathcal{H}_2$ be Hilbert spaces with scalar products $\langle \cdot, \cdot \rangle_1, \langle \cdot, \cdot \rangle_2$ and let $A : \mathcal{H}_1 \rightarrow \mathcal{H}_2$ be a bounded linear operator defined everywhere on \mathcal{H}_1 . Then the *adjoint operator* A^* of A is defined by the identity ([2], p. 196, Definition 1)

$$\begin{aligned} A^* : \mathcal{H}_2 &\rightarrow \mathcal{H}_1, \\ \langle x, A^* y \rangle_1 &:= \langle A x, y \rangle_2 \text{ for all } x \in \mathcal{H}_1, y \in \mathcal{H}_2. \end{aligned} \quad (16)$$

For Hilbert spaces $\mathcal{H}_1, \mathcal{H}_2$ over the reals, one writes A^T instead of A^* .

■ Theorems and Properties for Later Use

□ Simple Quadrature Formula

All Riemann sums integrate constant functions exactly on any grid, since

$$R(1, \Xi, X) := \sum_{i=1}^m (\xi_i - \xi_{i-1}) 1 = \xi_m - \xi_0 = b - a = \int_a^b 1 \, dx.$$

If f is bounded on $[a, b]$ and continuous except for finitely many points, it has a Riemann integral. Consult [3], Chapter 2, for proofs and further references on quadrature formulas.

Theorem 1

If $f'(x)$ exists and is bounded and integrable over $[a, b]$, then the errors of the right-hand Riemann sums satisfy

$$\lim_{m \rightarrow \infty} m E(f, \text{grid}(a, b, m, 1)) = \frac{b-a}{2} [f(a) - f(b)]. \quad (17)$$

A similar result holds for the left-hand Riemann sums (with $s = 0$).

If $f \in C^2[a, b]$, the error term of the elementary midpoint formula is given in [3], (2.6.5):

$$\int_a^{a+h} f(x) \, dx - h f\left(a + \frac{h}{2}\right) = \frac{1}{24} h^3 f''(\xi), \quad a < \xi < a + h. \quad (18)$$

Therefore the error of the composite midpoint formula can be bounded by

$$|E(f, \text{grid}(a, b, m, 1/2))| \leq \frac{m}{24} h^3 \max_{[a,b]} |f''(\xi)| = \frac{b-a}{24} h^2 \max_{[a,b]} |f''(\xi)|. \quad (19)$$

□ Convergence of Discrete to L^2 Scalar Products

By Theorem 1, for functions $f, g \in C^1[a, b]$ and $s = 0, 1$, the discrete scalar product $\langle f, g \rangle_{\text{grid}(a,b,m,s)}$ converges at least as fast as m^{-1} to the L^2 scalar product:

$$\begin{aligned} \langle f, g \rangle_{[a,b]} &= \langle f, g \rangle_{\text{grid}(a,b,m,s)} + h \frac{f(a)g(a) - f(b)g(b)}{2} = \langle f, g \rangle_{\text{grid}(a,b,m,s)} + O(m^{-1}), \\ s &= 0, 1. \end{aligned}$$

By equation (19), for functions $f, g \in C^2[a, b]$, the discrete scalar product $\langle f, g \rangle_{\text{grid}(a,b,m,1/2)}$ converges at least as fast as m^{-2} to the L^2 scalar product:

$$\langle f, g \rangle_{[a,b]} = \langle f, g \rangle_{\text{grid}(a,b,m,1/2)} + \text{constant } h^2 = \langle f, g \rangle_{\text{grid}(a,b,m,1/2)} + O(m^{-2}).$$

□ Singular Value Decomposition

See [4], sections 2.4 and 2.6.

Theorem 2

If A is a real $m \times n$ matrix, then there exist orthogonal matrices

$$U = [u_1, \dots, u_m] \in \mathbb{R}^{m \times m} \text{ and } V = [v_1, \dots, v_n] \in \mathbb{R}^{n \times n}$$

such that

$$U^T A V = \Sigma = \text{diag}(\sigma_1, \dots, \sigma_p) \in \mathbb{R}^{m \times n}, \quad p = \min(m, n),$$

where $\sigma_1 \geq \dots \geq \sigma_p \geq 0$.

The equivalence

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

is called the full singular value decomposition (SVD).

■ Matrix Properties from SVD

$$A^T = V \Sigma^T U^T, \quad A^T A = V \Sigma^T U^T U \Sigma V^T = V \Sigma^T \Sigma V^T. \quad (20)$$

Define r to be the index of the last positive singular value

$$\sigma_1 \geq \dots \geq \sigma_r > \sigma_{r+1} = \dots = \sigma_p = 0.$$

Then

$$\begin{aligned} \text{rank}(A) &= \text{rank}(\Sigma) = r, \quad \text{rank}(A^T) = \text{rank}(\Sigma^T) = r, \\ \text{rank}(A^T A) &= \text{rank}(\Sigma^T \Sigma) = r. \end{aligned} \quad (21)$$

The *condition number* of a rectangular matrix A with full column rank with respect to the 2-norm (in short, the 2-condition number) is defined as the quotient of its largest to its smallest singular value ([4], equation (2.6.5)):

$$\kappa_2(A) := \frac{\sigma_{\max}(A)}{\sigma_{\min}(A)}. \quad (22)$$

By equation (20), the 2-condition number has the properties

$$\kappa_2(A^T) = \kappa_2(A), \quad (23)$$

$$\kappa_2(A^T A) = \kappa_2(A)^2. \quad (24)$$

If A is an invertible $n \times n$ matrix, $\sigma_1 \geq \dots \geq \sigma_n > 0$. The SVD of A^{-1} is obtained from the SVD of A as

$$A^{-1} = V \operatorname{diag}(1/\sigma_n, \dots, 1/\sigma_1) U^T, \quad (25)$$

$$\kappa_2(A^{-1}) = \kappa_2(A). \quad (26)$$

If Q is a real $m \times n$ matrix with orthonormal columns $[q_1, \dots, q_n \in \mathbb{R}^{m \times n}]$, it can be completed to an orthogonal matrix $U = [q_1, \dots, q_n, u_{n+1}, \dots, u_m] \in \mathbb{R}^{m \times m}$. Therefore the SVD of Q is

$$Q = (U \begin{smallmatrix} 1_{m \times n} & 1_{n \times n}^T \end{smallmatrix}). \quad (27)$$

The 2-condition number of Q is

$$\kappa_2(Q) = 1. \quad (28)$$

□ Adjoint Linear Operators

See [2], sections VII.1 and VII.2.

Theorem 3

If $A: \mathcal{H}_1 \rightarrow \mathcal{H}_2$ is a bounded linear operator, defined everywhere on \mathcal{H}_1 , then $A^*: \mathcal{H}_2 \rightarrow \mathcal{H}_1$ is a bounded linear operator defined everywhere on \mathcal{H}_2 and $\|A\| = \|A^*\|$.

□ Finite-Dimensional Linear Operators

For $\mathcal{H}_1 = \mathbb{R}^n$ and a subset $\{\phi_1, \dots, \phi_n\} \subset \mathcal{H}_2$, linear combinations from the subset define a finite-dimensional linear operator

$$\begin{aligned} \Phi: \mathbb{R}^n &\rightarrow \mathcal{H}_2, \\ \Phi c &:= \sum_{j=1}^n c_j \phi_j. \end{aligned} \quad (29)$$

Obviously, $\operatorname{rank}(\Phi) \leq n$ and $\operatorname{rank}(\Phi) = n$ if and only if $\{\phi_1, \dots, \phi_n\}$ is linearly independent.

Φ is bounded and has the matrix representation

$$\Phi c = (\phi_1 \dots \phi_n) \begin{pmatrix} c_1 \\ \vdots \\ c_n \end{pmatrix}.$$

Apply the definition of the adjoint operator Φ^* and notice that the first scalar product is that of \mathbb{R}^n , $\langle c, d \rangle_{\mathbb{R}^n} = c^T d$; then

$$c^T (\Phi^* f) = \langle c, \Phi^* f \rangle_{\mathbb{R}^n} = \langle \Phi c, f \rangle_2 = \sum_{j=1}^n c_j \langle \phi_j, f \rangle_2 = c^T \begin{pmatrix} \langle \phi_1, f \rangle_2 \\ \vdots \\ \langle \phi_n, f \rangle_2 \end{pmatrix} \text{ for all } c \in \mathbb{R}^n,$$

$$f \in \mathcal{H}_2.$$

Therefore the adjoint operator Φ^* is

$$\begin{aligned} \Phi^* : \mathcal{H}_2 &\rightarrow \mathbb{R}^n, \\ \Phi^* f &= \begin{pmatrix} \langle \phi_1, f \rangle_2 \\ \vdots \\ \langle \phi_n, f \rangle_2 \end{pmatrix}, \end{aligned} \quad (30)$$

and $\text{rank}(\Phi^*) = \text{rank}(\Phi)$.

Substituting $f = \Phi c$ into the preceding equation gives the representation of $\Phi^* \Phi$ as an $n \times n$ matrix (note the scalar products are taken in \mathcal{H}_2):

$$\begin{aligned} \Phi^* \Phi : \mathbb{R}^n &\rightarrow \mathbb{R}^n, \\ \Phi^* \Phi &= \begin{pmatrix} \langle \phi_1, \phi_1 \rangle_2 & \cdots & \langle \phi_1, \phi_n \rangle_2 \\ \vdots & \ddots & \vdots \\ \langle \phi_n, \phi_1 \rangle_2 & \cdots & \langle \phi_n, \phi_n \rangle_2 \end{pmatrix}. \end{aligned} \quad (31)$$

Here $\Phi^* \Phi$ is Hermitian positive semidefinite if \mathcal{H}_2 is over the complex numbers, and symmetric positive semidefinite if \mathcal{H}_2 is over the reals, and

$$\text{rank}(\Phi^* \Phi) = \text{rank}(\Phi).$$

□ Properties of Vandermonde Matrices

A polynomial of degree n has at most n distinct zeros, therefore the set of monomials $\{1, x, \dots, x^{n-1}\}$ is a linearly independent subset of $L^2(a, b)$ and

$$\begin{aligned} \text{rank}(V(a, b, n, x)^T) &= \\ \text{rank}(V(a, b, n, x)^T V(a, b, n, x)) &= \text{rank}(V(a, b, n, x)) = n. \end{aligned} \quad (32)$$

By [5], the determinant of the $n \times n$ Vandermonde matrix is the product

$$\begin{vmatrix} 1 & x_1 & \cdots & x_1^{n-1} \\ 1 & x_2 & \cdots & x_2^{n-1} \\ \vdots & \vdots & \ddots & \vdots \\ 1 & x_n & \cdots & x_n^{n-1} \end{vmatrix} = \prod_{\substack{i,j=1 \\ i>j}}^n (x_i - x_j).$$

Therefore, the rectangular $m \times n$ Vandermonde matrix (15) has full rank $r = \max(m, n)$ if and only if the points x_j are pairwise disjoint.

■ Mathematica Definitions for Polynomial L^2 Approximation

Names are chosen according to previous notation and terminology.

This defines the 2-condition number.

```
ConditionNumber2[mat_?MatrixQ] :=
  With[{σ = SingularValueList[mat]}, First[σ] / Last[σ]]
```

This defines symbolic integration with time constrained to 10 seconds.

```
sIntegrate[f_, a_, b_] :=
  TimeConstrained[Integrate[f[x], {x, a, b},
    Assumptions → a < b], 10] /; Not[NumericQ[a] && NumericQ[b]]
```

This defines numerical integration.

```
nIntegrate[f_, a_, b_] :=
  Module[{x, symbint, resint},
    symbint = TimeConstrained[Integrate[f[x], {x, a, b}],
      5 (*Seconds*)];
    If[symbint === $Aborted || Head[symbint] === Integrate ||
      Head[symbint] === List &&
      Head[First[symbint]] === Integrate,
      resint = NIntegrate[f[x], {x, a, b}], resint = symbint];
    resint] /; NumericQ[a] && NumericQ[b]
```

The function wIntegrate (which is Listable) first attempts symbolic integration; if that is unsuccessful or takes too long, it performs numerical integration.

```
wIntegrate[f_, a_, b_] :=
  Simplify[If[NumericQ[a] && NumericQ[b], nIntegrate[f, a, b],
    sIntegrate[f, a, b]], a < b]
```

This defines the scalar product in $L^2(a, b)$.

```
scalarproductL2[f_, g_, a_, b_] :=
  wIntegrate[f[#] g[#] &, a, b]
```

Since scalarproductL2 is listable in f and g, it also implements the adjoint operator for a set of functions according to equation (32).

This defines the norm in $L^2(a, b)$.

```
normL2[f_, a_, b_] := Sqrt[scalarproductL2[f, f, a, b]]
```

This defines functions for discrete grids.

```
stepsizeh[a_, b_, m_?IntegerQ /; m > 0] := (b - a) / m

grid[a_, b_, m_, s_: 0] :=
  If[0 ≤ s ≤ 1,
    Simplify[a + stepsizeh[a, b, m] (Range[0, m - 1] + s)], {}]

riemannsum[f_, a_, b_, m_, s_: 0] :=
  stepsizeh[a, b, m] Total[Map[f, grid[a, b, m, s]]]
```

```

scalarproductgrid[f_, g_, a_, b_, m_, s_: 0] :=
  riemannsum[f[#] g[#] &, a, b, m, s]

norm2grid[f_, a_, b_, m_, s_: 0] :=
  Sqrt[scalarproductgrid[f, f, a, b, m, s]]

```

If a, b or s is a machine number, the functions `grid`, `riemannsum`, `scalarproductgrid` and `norm2grid` return machine numbers.

This defines the functions `monomials`, `polynomial`, `vandermonde` and `vandermondegrid`.

```

monomials[degree_?IntegerQ, x_] :=
  x^Range[0, degree] /. Indeterminate -> 1

polynomial[coefficients_?VectorQ, x_] :=
  FromDigits[Reverse[coefficients], x]

vandermonde[n_, x_?AtomQ] := {monomials[n - 1, x]}

vandermonde[n_, grid_?VectorQ] :=
  Quiet[Outer[Power, grid, Range[0, n - 1]] /. Indeterminate -> 1]

vandermondegrid[n_, a_, b_, m_, s_: 0] :=
  vandermonde[n, grid[a, b, m, s]]

```

To avoid potential clashes with predefined values for the variables a , b and x , the script letters a , b and x are used for symbolic results.

These sample functions are used in the `Manipulate` commands.

$\sin\left(\frac{2\pi x}{3} + \frac{1}{3}\right)$	1. no even/odd symmetry over the intervals $[-1, 1]$ and $[0, 1]$
$\sin(\pi x)$	2. periodic, smooth and antisymmetric
$\text{sgn}(x)$	3. antisymmetric; discontinuous
$ x $	4. symmetric; discontinuous first derivative
$\begin{cases} -2x - 2 & x < -\frac{1}{2} \\ 2x & -\frac{1}{2} \leq x \leq \frac{1}{2} \\ 2 - 2x & \text{True} \end{cases}$	5. antisymmetric; discontinuous first derivative
$\frac{x x }{2}$	6. antisymmetric; discontinuous second derivative
$\sqrt{ x }$	7. first derivative is singular at 0
$ x ^{3/2}$	8. second derivative is singular at 0
$e^{\sin(\pi x)}$	9. too demanding for symbolic integration
$e^{\sin\left(\frac{2\pi x}{3}\right)}$	10. symbolic integration not possible, uses numerical integration

■ Linear Least-Squares Solution via QR Factorization

Let $m, n \in \mathbb{N}$. Given a rectangular data matrix and an observation vector f ,

$$A \in \mathbb{R}^{m \times n}, \quad m \geq n, \quad \text{and } f \in \mathbb{R}^m, \quad (33)$$

the linear least-squares (LS) problem is to find:

$$\min_{c \in \mathbb{R}^n} \|A c - f\|_2. \quad (34)$$

A comprehensive description of the QR factorization of an $m \times n$ matrix via Householder, Givens, fast Givens, classical Gram–Schmidt and modified Gram–Schmidt methods is given in [4], section 5.2. Here only the essential steps are presented.

□ Principle of QR Factorization

Let Q be an orthogonal matrix. Such a matrix preserves lengths of vectors:

$$\|Q z\|_2 = \sqrt{z^T Q^T Q z} = \sqrt{z^T z} = \|z\|_2.$$

Given the real $m \times n$ matrix A , $m \geq n$, the goal is to construct an orthogonal $m \times m$ matrix Q such that

$$A = Q R,$$

where R is an $m \times n$ upper-triangular matrix of the form

$$R = \begin{pmatrix} \tilde{R} \\ 0 \end{pmatrix}, \quad \text{where } \tilde{R} \text{ has } n \text{ rows and there are } m - n \text{ rows of zeros,}$$

$$\text{and } \tilde{R} = \begin{pmatrix} r_{11} & \cdots & r_{1n} \\ \vdots & \ddots & \vdots \\ 0 & \cdots & r_{nn} \end{pmatrix} \in \mathbb{R}^{n \times n}.$$

Obviously

$$\text{rank}(A) = \text{rank}(R) = \text{rank}(\tilde{R}).$$

□ Linear Least-Squares Solution by “Thin” QR Factorization

The Mathematica function `QRDecomposition` deviates from the full QR factorization as follows:

$$A = \tilde{Q}^T \tilde{R}.$$

\tilde{Q} is output as an $n \times m$ matrix. The rows of \tilde{Q} are orthonormal. Only the upper-triangular $n \times n$ submatrix \tilde{R} is output.

Then the unique solution of the $n \times n$ upper-triangular system is straightforward:

$$\tilde{R} c_{\text{QR}} = \tilde{Q} f.$$

It gives the unique solution of the full-rank linear least-squares problem (34):

$$c_{\text{QR}} = \tilde{R}^{-1} \tilde{Q} f.$$

Since multiplication with an orthogonal matrix does not change the singular values, the condition numbers do not change either:

$$\kappa_2(A) = \kappa_2(R) = \kappa_2(\tilde{R}). \quad (35)$$

This holds in particular for the Vandermonde matrices, both in the discrete and continuous case.

□ Mathematica Definitions for QR Orthonormalization

This defines `qrDecompositionSolution` and `qrPolynomialgrid`.

```
qrDecompositionSolution[amat_?MatrixQ, rhs_?VectorQ] :=
Module[{qTmat, rmat, qrsol},
  {qTmat, rmat} = QRDecomposition[amat];
  qrsol = LinearSolve[rmat, qTmat.rhs];
  {Transpose[qTmat], rmat, qrsol}]

qrPolynomialgrid[f_, a_, b_, n_, x_, m_, s_: 0] :=
polynomial[
  Last[qrDecompositionSolution[
    vandermondegrid[n, a, b, m, s],
    Map[f, grid[a, b, m, s]]]], x]
```

■ Classical Gram–Schmidt Orthonormalization

□ Polynomial Approximation via Classical Gram–Schmidt Orthonormalization

Applying the classical Gram–Schmidt orthonormalization process in a pre-Hilbert space, described in [2], p. 88 ff., to the monomials $\{1, x, \dots, x^{n-1}\}$ in $L^2(a, b)$ gives an orthonormal system of n polynomials

$$\mathcal{P}(a, b, n, x) := \{p(a, b, 0, x), p(a, b, 1, x), \dots, p(a, b, n-1, x)\} \quad (36)$$

that satisfy

$$\text{degree}(p(a, b, j, x)) = j, \quad j = 0, 1, \dots, n-1. \quad (37)$$

The Fourier coefficients of any function $f \in L^2(a, b)$ with respect to this orthonormal system are defined according to [2], p. 86, equation (1) (the dot here denotes the placeholder for the integration argument in the scalar product):

$$f_j := \langle p(a, b, j, \cdot), f \rangle_{[a,b]} = \int_a^b p(a, b, j, t) f(t) dt, \quad j = 0, 1, \dots, n-1.$$

The best L^2 approximation to f is given as the Fourier sum of n terms:

$$s_n(x) = \sum_{j=0}^{n-1} f_j p(a, b, j, x).$$

□ Mathematica Definitions for Classical Gram–Schmidt Orthonormalization

The orthonormal system of polynomials $\mathcal{P}(a, b, n, x)$ is given by the function `orthonormalP`. The functions `fourierCoefficient` and `fourierPolynomial` are also defined.

```

orthonormalP[a_, b_, degree_?IntegerQ, x_] :=
orthonormalP[a, b, degree, x] =
  FullSimplify[Orthogonalize[monomials[degree, x],
    Integrate[#1 #2, {x, a, b}, Assumptions → a < b] &,
    Method → "GramSchmidt"], a < b]

fourierCoefficient[f_, a_, b_, n_] :=
  scalarproductL2[orthonormalP[a, b, n - 1, #] &, f, a, b]

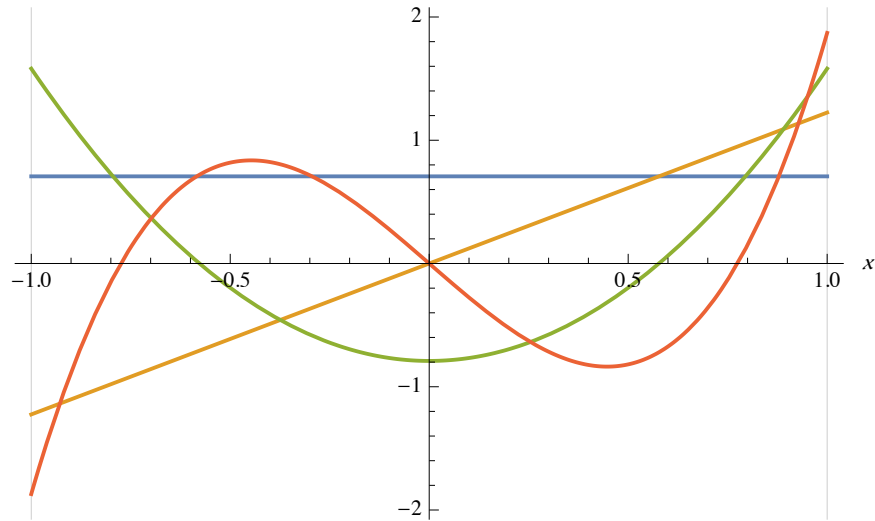
fourierPolynomial[f_, a_, b_, n_, x_] :=
  Simplify[Dot[fourierCoefficient[f, a, b, n],
    orthonormalP[a, b, n - 1, x]], a < b]

```

□ Example Interval $[-1, 1]$: Fourier Approximation by Legendre Polynomials

This defines the polynomials `legendreP`.

```
legendreP[degree_, x_] := orthonormalP[-1, 1, degree, x]
```



$$\text{blue line: } \frac{1}{\sqrt{2}} \quad \text{orange line: } \sqrt{\frac{3}{2}} x$$

$$\text{green line: } \frac{1}{2} \sqrt{\frac{5}{2}} (-1 + 3x^2)$$

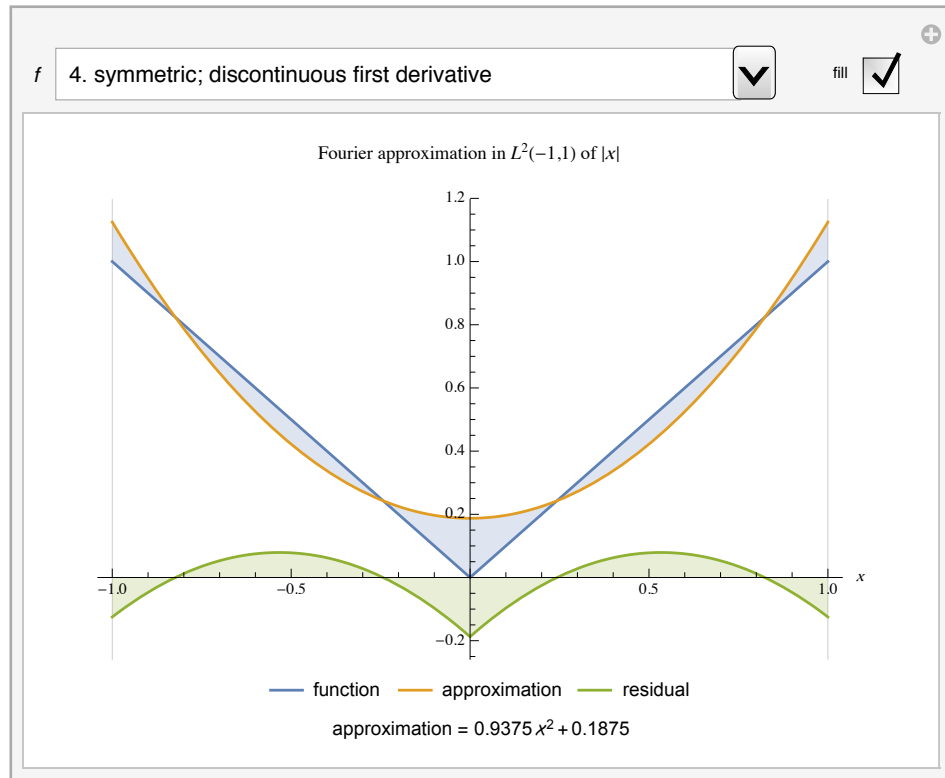
$$\text{red line: } \frac{1}{2} \sqrt{\frac{7}{2}} x (-3 + 5x^2)$$

These polynomials differ from the classical Legendre polynomials `LegendreP` built into Mathematica only by normalization factors.

```
With[{degree = 3},  
  TraditionalForm[Table[LegendreP[j, x], {j, 0, degree}]]]
```

$$\left\{1, x, \frac{1}{2}(3x^2 - 1), \frac{1}{2}(5x^3 - 3x)\right\}$$

This shows the Fourier approximation of a sample set of functions.



□ The Classical Gram–Schmidt Orthonormalization Interpreted as QR Factorization

Proposition 1

Define $C(a, b, n)$ as the matrix of the monomial coefficients of the polynomials $\mathcal{P}(a, b, n, x)$ of equation (36). Then $C(a, b, n)$ is an invertible lower-triangular $n \times n$ matrix and

$$C(a, b, n) \begin{pmatrix} 1 \\ x \\ \vdots \\ x^{n-1} \end{pmatrix} = \begin{pmatrix} p(a, b, 0, x) \\ p(a, b, 1, x) \\ \vdots \\ p(a, b, n-1, x) \end{pmatrix}, \quad (38)$$

$$\{1, x, \dots, x^{n-1}\} C(a, b, n)^T = \{p(a, b, 0, x), p(a, b, 1, x), \dots, p(a, b, n-1, x)\}. \quad (39)$$

Proof

The Gram–Schmidt orthonormalization is an invertible linear mapping of Π_{n-1} from the basis of monomials to the orthonormal basis $\mathcal{P}(a, b, n, x)$, representable by a real $n \times n$ matrix $C(a, b, n)$. By (37), the matrix is lower triangular. Transposing (38) gives (39).

□

Theorem 4

The classical Gram–Schmidt orthonormalization applied to the monomials $\{1, x, \dots, x^{n-1}\}$ is equivalent to the QR factorization of the Vandermonde operator $V(a, b, n, x) : \mathbb{R}^n \rightarrow L^2(a, b)$:

$$V(a, b, n, x) C(a, b, n)^T = Q(a, b, n, x),$$

$$V(a, b, n, x) = Q(a, b, n, x) (C(a, b, n)^T)^{-1} = Q(a, b, n, x) R(a, b, n),$$

where $Q(a, b, n, x)$ is the n -dimensional operator with orthonormal columns

$$Q(a, b, n, x) : L^2(a, b) \rightarrow \mathbb{R}^n,$$

$$Q(a, b, n, x) := (p(a, b, 0, x), p(a, b, 1, x), \dots, p(a, b, n-1, x)).$$

Proof

This follows from Proposition 1 by interpreting matrices as operators.

□

For this QR factorization, the inverse of the upper-triangular $n \times n$ matrix R is already calculated by the Gram–Schmidt process:

$$R(a, b, n)^{-1} = C(a, b, n)^T.$$

Equations (26), (28) and (35) give the relations on the 2-condition numbers of the operators or matrices:

Corollary 1

$$\kappa_2(Q(a, b, n, x)) = 1,$$

$$\kappa_2(V(a, b, n, x)) = \kappa_2(C(a, b, n)) = \kappa_2(R(a, b, n)).$$

The numerical instability of the classical Gram–Schmidt process in machine arithmetic is discussed in [4], section 5.2.8. However, since the Gram–Schmidt orthonormalization of the monomials with respect to the scalar product $\langle f, g \rangle_{[a,b]}$ can be performed by symbolic calculations, numerical algorithm stability is not an issue here, contrary to the m -dimensional scalar product $\langle f, g \rangle_{\text{grid}(a,b,m,s)}$.

□ Mathematica Definitions: Gram–Schmidt Orthonormalization as QR Factorization

This defines the Gram–Schmidt coefficient matrix $C(a, b, n)$.

```
gramschmidtC[a_, b_, n_] :=
  Simplify[
    PadRight[CoefficientList[orthonormalP[a, b, n - 1, x], x],
      {n, n}], a < b]
```

This defines the lower- and upper-triangular matrices of QR decomposition.

```
gramschmidtL[a_, b_, n_] :=
  FullSimplify[Inverse[gramschmidtC[a, b, n]], a < b]

gramschmidtR[a_, b_, n_] := Transpose[gramschmidtL[a, b, n]]
```

This defines the orthogonal matrix $Q(a, b, n, x)$.

```
orthonormalQ[a_, b_, n_, x_] := {orthonormalP[a, b, n - 1, x]}
```

□ Illustrating Proposition 1 and Theorem 4 for an Arbitrary Interval

Here is a set of orthonormal polynomials ($n = 3$, degree = $n - 1 = 2$).

```
orthonormalP[a, b, 2, x]
```

$$\left\{ \frac{1}{\sqrt{-a+b}}, -\frac{\sqrt{3}(a+b-2x)}{(-a+b)^{3/2}}, \frac{\sqrt{5}(a^2+4ab+b^2-6(a+b)x+6x^2)}{(-a+b)^{5/2}} \right\}$$

This gives the Gram–Schmidt coefficient matrix $C(a, b, n)$, with $n = 3$.

```
gramschmidtC[a, b, 3] // MatrixForm
```

$$\begin{pmatrix} \frac{1}{\sqrt{-a+b}} & 0 & 0 \\ -\frac{\sqrt{3}(a+b)}{(-a+b)^{3/2}} & \frac{2\sqrt{3}}{(-a+b)^{3/2}} & 0 \\ \frac{\sqrt{5}(a^2+4ab+b^2)}{(-a+b)^{5/2}} & -\frac{6\sqrt{5}(a+b)}{(-a+b)^{5/2}} & \frac{6\sqrt{5}}{(-a+b)^{5/2}} \end{pmatrix}$$

Multiply the matrix of monomials from the right or left.

Simplify[gramschmidtC[a, b, 3].monomials[2, x]]

$$\left\{ \frac{1}{\sqrt{-a+b}}, -\frac{\sqrt{3} (a+b-2x)}{(-a+b)^{3/2}}, \frac{\sqrt{5} (a^2+4ab+b^2-6ax-6bx+6x^2)}{(-a+b)^{5/2}} \right\}$$

Simplify[monomials[2, x].Transpose[gramschmidtC[a, b, 3]]]

$$\left\{ \frac{1}{\sqrt{-a+b}}, -\frac{\sqrt{3} (a+b-2x)}{(-a+b)^{3/2}}, \frac{\sqrt{5} (a^2+4ab+b^2-6ax-6bx+6x^2)}{(-a+b)^{5/2}} \right\}$$

This reproduces orthogonal matrix $Q(a, b, n, x)$.

MatrixForm[orthonormalQ[a, b, 3, x]]

$$\left(\begin{array}{ccc} \frac{1}{\sqrt{-a+b}} & -\frac{\sqrt{3} (a+b-2x)}{(-a+b)^{3/2}} & \frac{\sqrt{5} (a^2+4ab+b^2-6(a+b)x+6x^2)}{(-a+b)^{5/2}} \end{array} \right)$$

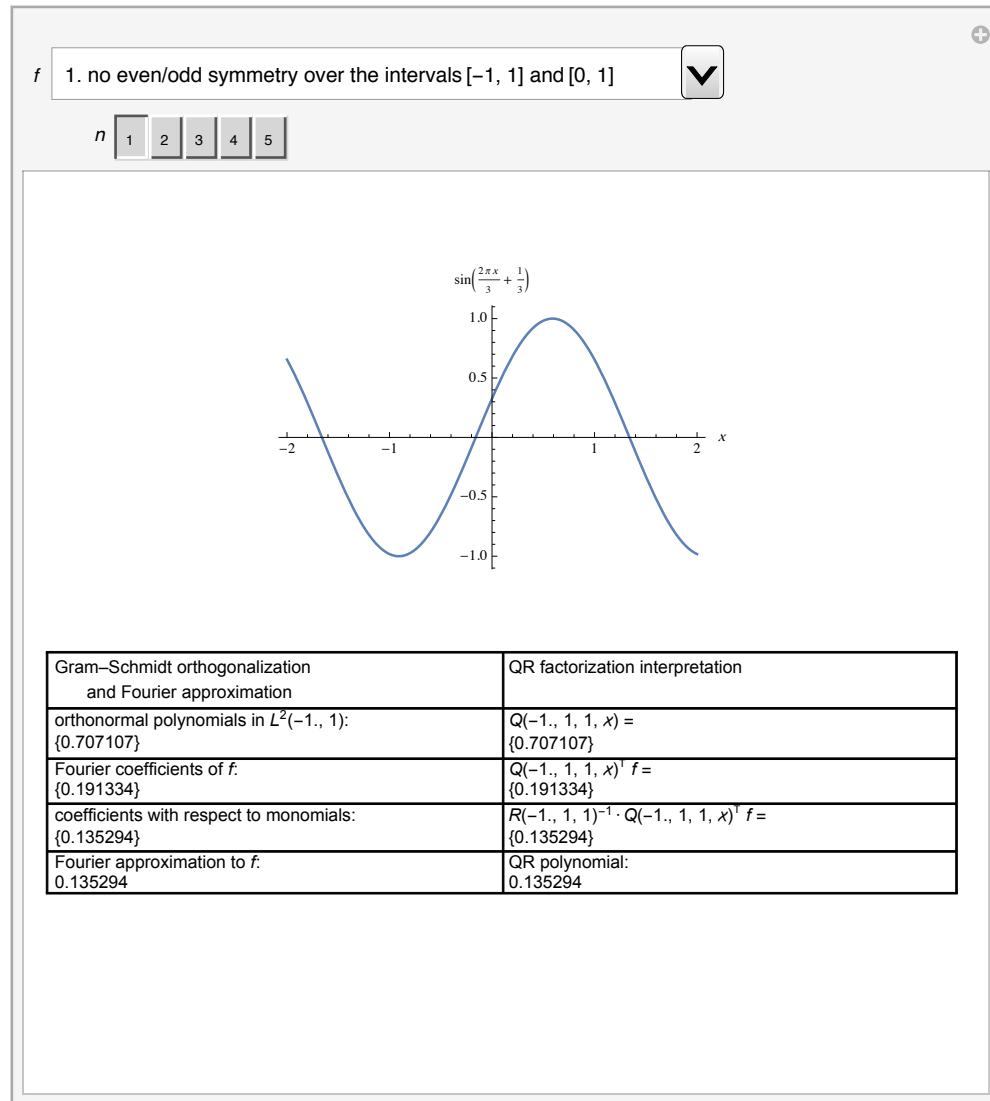
By construction, the polynomials contained in the matrix columns are orthonormal with respect to the scalar product in $L^2(a, b)$.

**MatrixForm[
Simplify[
wIntegrate[
Function[x, Transpose[orthonormalQ[a, b, 3, x]].
orthonormalQ[a, b, 3, x]], a, b]]]**

$$\left(\begin{array}{ccc} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{array} \right)$$

□ Correspondence between Gram–Schmidt Orthonormalization and QR Factorization Interpretation

Select one of the sample functions and compare the results from the Gram–Schmidt orthonormalization and QR factorization interpretation.



□ Examples: Condition Numbers of Vandermonde Matrices

For the continuous case, because $Q(a, b, n, x)^T Q(a, b, n, x) = 1_{n \times n}$, the singular values of $V(a, b, n, x)$ equal the singular values of $R(a, b, n)$.

```
With[{a = 0, b = 1},
  Table[SingularValueList[N[gramschmidtR[a, b, n]]],
    {n, 5}] // TableForm
```

1.				
1.12587	0.256401			
1.18673	0.349753	0.0518396		
1.22483	0.411268	0.082087	0.00983373	
1.25182	0.456655	0.106806	0.0174899	0.00181326

```
With[{a = 0, b = 1},
  Table[ConditionNumber2[N[gramschmidtR[a, b, n]]], {n, 8}]]
```

```
{1., 4.39107, 22.8923, 124.554,
  690.367, 3866.66, 21802.9, 123522.}
```

Here is the case of a discrete grid.

```
With[{a = 0, b = 1, m = 10, s = 1.},
  Table[SingularValueList[N[vandermondegrid[n, a, b, m, s]]],
    {n, 5}] // TableForm
```

3.16228				
3.63679	0.789785			
3.88628	1.12125	0.151463		
4.05784	1.35331	0.251982	0.026508	
4.19169	1.53348	0.338647	0.0499783	0.00432751

```
With[{a = 0, b = 1, m = 10, s = 1.},
  Table[ConditionNumber2[N[vandermondegrid[n, a, b, m, s]]],
    {n, 8}]]
```

```
{1., 4.60479, 25.6582, 153.08,
  968.615, 6561.32, 48523.3, 404846.}
```

□ Example Interval $[0, 1]$: Relation to Hilbert Matrices

Obviously, `gramschmidtL[0, 1, n]` is the lower-triangular Cholesky factor of the `HilbertMatrix[n]`.

```
matrices01 = Module[
  {n = 4, gs01},
  gs01 = gramschmidtL[0, 1, n];
  {gs01, gs01.Transpose[gs01], HilbertMatrix[n],
   Transpose[CholeskyDecomposition[HilbertMatrix[n]]]};
```

```
Map[MatrixForm, matrices01]
```

$$\left\{ \begin{pmatrix} 1 & 0 & 0 & 0 \\ \frac{1}{2} & \frac{1}{2\sqrt{3}} & 0 & 0 \\ \frac{1}{3} & \frac{1}{2\sqrt{3}} & \frac{1}{6\sqrt{5}} & 0 \\ \frac{1}{4} & \frac{3\sqrt{3}}{20} & \frac{1}{4\sqrt{5}} & \frac{1}{20\sqrt{7}} \end{pmatrix}, \begin{pmatrix} 1 & \frac{1}{2} & \frac{1}{3} & \frac{1}{4} \\ \frac{1}{2} & \frac{1}{3} & \frac{1}{4} & \frac{1}{5} \\ \frac{1}{3} & \frac{1}{4} & \frac{1}{5} & \frac{1}{6} \\ \frac{1}{4} & \frac{1}{5} & \frac{1}{6} & \frac{1}{7} \end{pmatrix}, \right.$$

$$\left. \begin{pmatrix} 1 & \frac{1}{2} & \frac{1}{3} & \frac{1}{4} \\ \frac{1}{2} & \frac{1}{3} & \frac{1}{4} & \frac{1}{5} \\ \frac{1}{3} & \frac{1}{4} & \frac{1}{5} & \frac{1}{6} \\ \frac{1}{4} & \frac{1}{5} & \frac{1}{6} & \frac{1}{7} \end{pmatrix}, \begin{pmatrix} 1 & 0 & 0 & 0 \\ \frac{1}{2} & \frac{1}{2\sqrt{3}} & 0 & 0 \\ \frac{1}{3} & \frac{1}{2\sqrt{3}} & \frac{1}{6\sqrt{5}} & 0 \\ \frac{1}{4} & \frac{3\sqrt{3}}{20} & \frac{1}{4\sqrt{5}} & \frac{1}{20\sqrt{7}} \end{pmatrix} \right\}$$

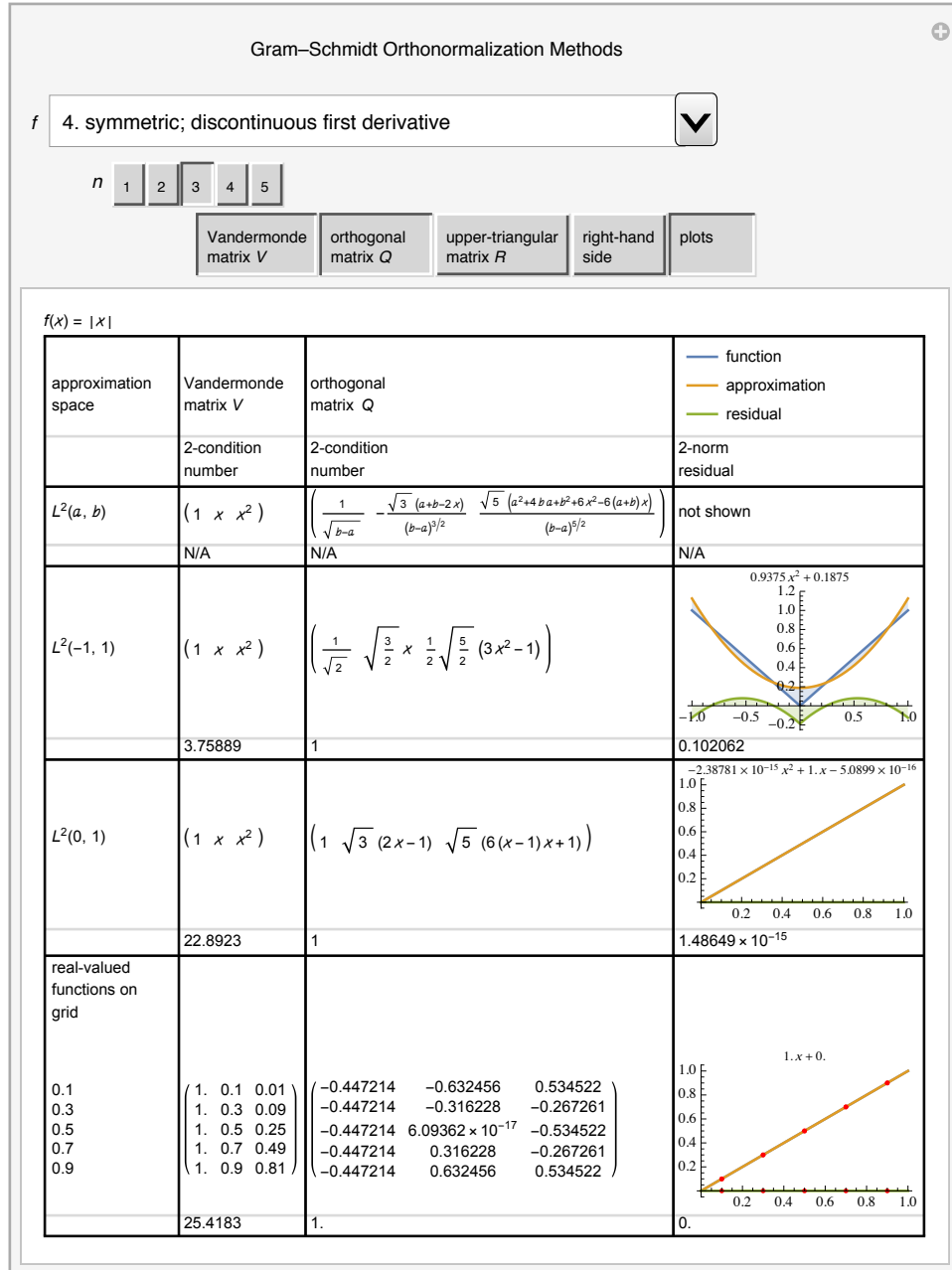
Consequently, the 2-condition number of the Vandermonde matrix of size n on $[0, 1]$ is the square root of the 2-condition number of `HilbertMatrix[n]`.

```
Map[ConditionNumber2, N[matrices01]]
```

```
{124.554, 15 513.7, 15 513.7, 124.554}
```

■ Summary of Results for Orthogonalization Methods

This gives summary results.



□ Characterizing Orthogonalization Solutions

1. The system matrix V is a Vandermonde matrix of dimension $1 \times n$ in the continuous setting and $m \times n$ in the discrete setting.
2. V has full rank in the continuous and discrete settings.
3. The discrete case is solved by the QR factorization of $V_{m \times n}$ into an orthogonal matrix $Q_{m \times n}$ and an upper-triangular matrix $R_{n \times n}$.
4. In the continuous case, the Gram–Schmidt orthonormalization applied to the monomials in $L^2(a, b)$ reads in matrix vector notation as $\{1, x, \dots, x^{n-1}\} C(a, b, n)^T = \{p(a, b, 0, x), p(a, b, 1, x), \dots, p(a, b, n-1, x)\}$.
5. This is reformulated as QR factorization of the $1 \times n$ Vandermonde matrix: $V(a, b, n, x) = (1 \ x \ \dots \ x^{n-1}) = Q(a, b, n, x) R(a, b, n)$.
6. Consequently, the 2-condition number of $V(a, b, n, x)$ equals the 2-condition number of the $n \times n$ upper-triangular matrix $R(a, b, n)$:
 - a. For intervals symmetric around zero, $[-b, b]$, the columns of $V(-b, b, n, x)$ are alternating orthogonal. Therefore the 2-condition number is much smaller than for $[0, b]$.
 - b. The minimum condition number is achieved for interval $[-1, 1]$.
 - c. Therefore, to obtain the most precise numeric results, the approximation interval $[a, b]$ should be linearly mapped onto $[-1, 1]$.
 - d. $V(a, b, n, x)$ is well-conditioned (i.e. less than about 10,000) for the interval $[0, 1]$ and $n \leq 6$.
 - e. $V(a, b, n, x)$ is ill-conditioned for the interval $[0, 1]$, $n > 6$, and generally for intervals $[a, a+1]$, $a \gg 0$.

■ The Normal Equations Method

□ The Normal Equations Method for Finite Dimensions

The approach for deriving the normal equations for the $m \times n$ least-squares problem (34) is described in [4], section 5.3, for example. Define

$$\phi(c) := \frac{1}{2} \langle A c - f, A c - f \rangle,$$

$$\nabla \phi(c) = A^T (A c - f).$$

A necessary condition for a minimum is $\nabla \phi(c) = 0$, or equivalently,

$$A^T A c = A^T f. \tag{40}$$

These are called the *normal equations*. The minimum is unique if the $n \times n$ *Hessian matrix* $H := A^T A$ has full rank n . Then there is a unique solution c_{LS} of the linear least-squares problem (34) or (40):

$$c_{LS} = (A^T A)^{-1} A^T f. \quad (41)$$

For an equally spaced grid (a, b, m, s) , defined as in equation (5), the $m \times n$ Vandermonde matrix $V(\text{grid}(a, b, m, s), n)$ has full rank, so the Hessian matrix for polynomial approximation has full rank as well.

If A is rank deficient ($\text{rank}(A) < \min(m, n)$), then there are an infinite number of solutions to the least-squares problem. There is still a unique solution with minimum 2-norm, which can be written with the pseudo-inverse matrix ([4], section 5.5):

$$c_{LS} = A^+ f.$$

For full rank, $A^+ = (A^T A)^{-1} A^T$.

□ The Normal Equations Algorithm for the $m \times n$ Case

Suppose $A \in \mathbb{R}^{m \times n}$ with $\text{rank}(A) = n$ (see [4] section 5.3.1). Then this algorithm computes the unique solution c to the linear least-squares problem (34):

- Compute the lower-triangular portion of $C = A^T A$.
- Compute $d = A^T f$.
- Compute the Cholesky factorization $C = L L^T$ (or $C = U^T U$ via `CholeskyDecomposition`).
- Solve the triangular systems $L y = d$ and $L^T c = y$ (or $U^T y = d$ and $U c = y$).

□ The Normal Equations Method on $L^2(a, b)$

The approach for deriving the normal equations for the $m \times n$ case applies with one modification to the continuous least-squares approximation equation (12) (see [6]):

$$\phi(c) := \frac{1}{2} \langle V(a, b, n, x) c - f, V(a, b, n, x) c - f \rangle_{[a,b]}.$$

The matrix transpose $A^T: \mathbb{R}^m \rightarrow \mathbb{R}^n$ has to be replaced by the adjoint operator $V(a, b, n, x)^T: L^2(a, b) \rightarrow \mathbb{R}^n$:

$$\nabla \phi(c) = V(a, b, n, x)^T (V(a, b, n, x) c - f).$$

A necessary condition for a minimum is $\nabla \phi(c) = 0$, or equivalently,

$$V(a, b, n, x)^T V(a, b, n, x) c = V(a, b, n, x)^T f. \quad (42)$$

These are called the *normal equations*.

$$H(a, b, n) := V(a, b, n, x)^T V(a, b, n, x) \quad (43)$$

is called the $n \times n$ *Hessian matrix*. The minimum is unique if $H(a, b, n)$ has full rank n .

The elements can be calculated via equation (31):

$$H(a, b, n)_{ij} = \langle x^{i-1}, x^{j-1} \rangle_{[a,b]} = \int_a^b x^{i+j-2} dx = \frac{b^{i+j-1} - a^{i+j-1}}{i+j-1}, \quad i, j = 1, \dots, n,$$

$$H(a, b, n) := \begin{pmatrix} \langle 1, 1 \rangle_{[a,b]} & \langle 1, x \rangle_{[a,b]} & \dots & \langle 1, x^{n-1} \rangle_{[a,b]} \\ \langle x, 1 \rangle_{[a,b]} & \langle x, x \rangle_{[a,b]} & \dots & \langle x, x^{n-1} \rangle_{[a,b]} \\ \vdots & \vdots & \ddots & \vdots \\ \langle x^{n-1}, 1 \rangle_{[a,b]} & \langle x^{n-1}, x \rangle_{[a,b]} & \dots & \langle x^{n-1}, x^{n-1} \rangle_{[a,b]} \end{pmatrix}.$$

Obviously, the Hessian matrix is symmetric and positive semidefinite. Since $V(a, b, n, x)$ has full rank for any nonzero interval $[a, b]$, then $V(a, b, n, x)^T V(a, b, n, x)$ has full rank n (as well by equation (32)) and is therefore positive definite. Then there exists a unique solution c_{LS} of equations (42) and (12).

Finally, calculate the elements on the right-hand side of $V(a, b, n, x)^T f$ via equation (30):

$$(V(a, b, n, x)^T f)_i = \langle x^{i-1}, f \rangle_{[a,b]} = \int_a^b x^{i-1} f(x) dx, \quad i = 1, \dots, n.$$

□ Convergence of Discrete Approximations to Continuous Approximations

This subsection investigates under which conditions and how fast the polynomial approximations on discrete grids converge to the continuous polynomial approximation in $L^2(a, b)$.

For an equally spaced $X := \text{grid}(a, b, m, s)$, the normal equations, multiplied by the step size $h = (b - a) / m$, read

$$h V(n, X)^T V(n, X) c = h V(n, X)^T f(X). \quad (44)$$

Define

$$H(a, b, n, m, s) := \frac{b-a}{m} V(n, X)^T V(n, X); \quad (45)$$

then by equation (7), the matrix elements of $H(a, b, m, n, s)$ are just the Riemann sums for the matrix elements (integrals) of $H(a, b, n)$. Therefore, the Hessian matrices on the discrete grid converge to the continuous Hessian matrix in any matrix norm according to:

$$\lim_{m \rightarrow \infty} \|H(a, b, n) - H(a, b, n, m, s)\| = \begin{cases} O(m^{-1}) & \text{for Riemann sums,} \\ O(m^{-2}) & \text{for the composite midpoint rule.} \end{cases}$$

Define

$$\text{rhs}(a, b, n, m, s) := \frac{b-a}{m} V(n, X)^T f(X); \quad (46)$$

then by equation (7), the elements of $\text{rhs}(a, b, n, m, s)$ are just the Riemann sums for the elements of $V(x, a, b, n)^T f$, the moments of f . Therefore, the right-hand side of the normal equations on the discrete grid converge to the right-hand side of the continuous normal equations in any vector norm according to:

$$\begin{aligned} \lim_{m \rightarrow \infty} \|V(a, b, n, x)^T f - \text{rhs}(a, b, n, m, s)\| = \\ \begin{cases} O(m^{-1}) & \text{for Riemann sums and } f \in C^1[a, b], \\ O(m^{-2}) & \text{for the composite midpoint rule and } f \in C^2[a, b]. \end{cases} \end{aligned} \quad (47)$$

Proposition 2

The polynomial approximations on the discrete grids converge to the continuous polynomial approximation with the same order as the Riemann sums.

Proof

From equation (42), the solution of the polynomial approximation in $L^2(a, b)$ is

$$c_{[a,b]} = H(a, b, n)^{-1} V(a, b, n, x)^T f.$$

From equation (44), the solution for the discrete grid is

$$c_{LS} = H(a, b, n, m, s)^{-1} \text{rhs}(a, b, n, m, s). \quad (48)$$

For the matrix inverses

$$\begin{aligned} \lim_{m \rightarrow \infty} \|H(a, b, n)^{-1} - H(a, b, n, m, s)^{-1}\| = \\ \begin{cases} O(m^{-1}) & \text{for Riemann sums,} \\ O(m^{-2}) & \text{for the composite midpoint rule.} \end{cases} \end{aligned}$$

Expanding the difference of the solution coefficient vectors completes the proof:

$$\begin{aligned} c_{[a,b]} - c_{LS} = [H(a, b, n)^{-1} - H(a, b, n, m, s)^{-1}] V(a, b, n, x)^T f + \\ H(a, b, n, m, s)^{-1} [V(a, b, n, x)^T f - \text{rhs}(a, b, n, m, s)]. \end{aligned}$$

□

□ The Relation of the Normal Equations Method to Gram–Schmidt Orthonormalization on $L^2(a, b)$

Theorem 5

The $n \times n$ Hessian matrix of the normal equations on $L^2(a, b)$ is related to the lower-triangular Gram–Schmidt coefficient matrix $C(a, b, n)$ from Proposition 1 and the upper-triangular matrix $R(a, b, n)$ from Theorem 4 by

$$H(a, b, n) = C(a, b, n)^{-1} (C(a, b, n)^{-1})^T = R(a, b, n)^T R(a, b, n).$$

Equivalently, $C(a, b, n)^{-1}$ is the lower-triangular Cholesky factor and $R(a, b, n)$ the upper-triangular Cholesky factor of the symmetric positive-definite Hessian matrix of the normal equations method.

Proof

From Theorem 4 and because $Q(a, b, n, x)^T Q(a, b, n, x) = 1_{n \times n}$,

$$\begin{aligned} H(a, b, n) &= V(a, b, n, x)^T V(a, b, n, x) = \\ &C(a, b, n)^{-1} Q(a, b, n, x)^T Q(a, b, n, x) (C(a, b, n)^{-1})^T = \\ &C(a, b, n)^{-1} (C(a, b, n)^{-1})^T = R(a, b, n)^T R(a, b, n). \end{aligned}$$

By [4], Theorem 4.2.7, the Cholesky decomposition of a symmetric positive-definite square matrix is unique.

□

Equations (24) and (26) give the relation for the 2-condition numbers of the matrices.

Corollary 2

$$\kappa_2(H(a, b, n)) = \kappa_2(C(a, b, n))^2 = \kappa_2(R(a, b, n))^2.$$

□ Definitions to Solve Normal Equations

This defines the continuous and discrete Hessian matrices.

```
hessian[a_, b_, n_] :=
  Simplify[
    wIntegrate[
      Function[x, Outer[Times, monomials[n - 1, x],
        monomials[n - 1, x]]], a, b], a < b]

hessiangrid[a_, b_, n_, m_, s_: 0] :=
  stepsizeh[a, b, m] With[{v = vandermondegrid[n, a, b, m, s]},
    Transpose[v].v]
```

These are the right-hand sides of the continuous and discrete normal equations.

```
normalrhs[f_, a_, b_, n_] :=
  scalarproductL2[monomials[n - 1, #] &, f, a, b]

normalrhsgrid[f_, a_, b_, n_, m_, s_: 0] :=
  stepsizeh[a, b, m] With[{gr = grid[a, b, m, s]},
    Transpose[vandermonde[n, gr]].Map[f, gr]]
```

This gives the solution of the normal equations system.

```
normalDecompositionSolution[ata_?MatrixQ, rhs_?VectorQ] :=
Module[
  {umat, y, normalcoeff},
  umat = Simplify[CholeskyDecomposition[ata]];
  y = Simplify[LinearSolve[Transpose[umat], rhs]];
  normalcoeff = Simplify[LinearSolve[umat, y]];
  {umat, normalcoeff}
]
```

This gives the approximation polynomials for the continuous and discrete cases.

```
normalPolynomial[f_, a_, b_, n_, x_] :=
  Simplify[
    polynomial[
      Last[normalDecompositionSolution[hessian[a, b, n],
        normalrhs[f, a, b, n]]], x], a < b]

normalPolynomialgrid[f_, a_, b_, n_, x_, m_, s_: 0] :=
  polynomial[
    Last[normalDecompositionSolution[
      hessiangrid[a, b, n, m, s],
      normalrhsgrid[f, a, b, n, m, s]]], x]
```

□ **Illustrating Theorem 4 for the General Interval $[a, b]$**

This gives the Gram–Schmidt coefficient matrix, its inverse and inverse transpose.

```
With[{n = 3},
  TraditionalForm@
  Column[
    Riffle[Map[MatrixForm,
      {gramschmidtC[a, b, n], gramschmidtL[a, b, n],
       gramschmidtR[a, b, n]}], ""]]]
```

$$\begin{pmatrix} \frac{1}{\sqrt{b-a}} & 0 & 0 \\ -\frac{\sqrt{3}(a+b)}{(b-a)^{3/2}} & \frac{2\sqrt{3}}{(b-a)^{3/2}} & 0 \\ \frac{\sqrt{5}(a^2+4ab+b^2)}{(b-a)^{5/2}} & -\frac{6\sqrt{5}(a+b)}{(b-a)^{5/2}} & \frac{6\sqrt{5}}{(b-a)^{5/2}} \end{pmatrix}$$

$$\begin{pmatrix} \sqrt{b-a} & 0 & 0 \\ \frac{1}{2}\sqrt{b-a}(a+b) & \frac{(b-a)^{3/2}}{2\sqrt{3}} & 0 \\ \frac{1}{3}\sqrt{b-a}(a^2+ab+b^2) & \frac{(b-a)^{3/2}(a+b)}{2\sqrt{3}} & \frac{(b-a)^{5/2}}{6\sqrt{5}} \end{pmatrix}$$

$$\begin{pmatrix} \sqrt{b-a} & \frac{1}{2}\sqrt{b-a}(a+b) & \frac{1}{3}\sqrt{b-a}(a^2+ab+b^2) \\ 0 & \frac{(b-a)^{3/2}}{2\sqrt{3}} & \frac{(b-a)^{3/2}(a+b)}{2\sqrt{3}} \\ 0 & 0 & \frac{(b-a)^{5/2}}{6\sqrt{5}} \end{pmatrix}$$

The matrix `gramschmidtL[a, b, n]` times its transpose equals the Hessian matrix of the normal equations.

```
Module[{n = 3, gsL, hess},
  gsL = gramschmidtL[a, b, n];
  hess = hessian[a, b, n];
  TraditionalForm@
  Column[
    Riffle[Map[MatrixForm,
      {FullSimplify[gsL.Transpose[gsL]], hess,
       Simplify[gsL.Transpose[gsL] - hess]}], ""]]]
```

$$\begin{pmatrix} b-a & \frac{1}{2}(b-a)(a+b) & \frac{1}{3}(b^3-a^3) \\ \frac{1}{2}(b-a)(a+b) & \frac{1}{3}(b^3-a^3) & \frac{1}{4}(b^4-a^4) \\ \frac{1}{3}(b^3-a^3) & \frac{1}{4}(b^4-a^4) & \frac{1}{5}(b^5-a^5) \end{pmatrix}$$

$$\begin{pmatrix} b-a & \frac{1}{2}(b^2-a^2) & \frac{1}{3}(b^3-a^3) \\ \frac{1}{2}(b^2-a^2) & \frac{1}{3}(b^3-a^3) & \frac{1}{4}(b^4-a^4) \\ \frac{1}{3}(b^3-a^3) & \frac{1}{4}(b^4-a^4) & \frac{1}{5}(b^5-a^5) \end{pmatrix}$$

$$\begin{pmatrix} 0 & 0 & 0 \\ 0 & 0 & 0 \\ 0 & 0 & 0 \end{pmatrix}$$

Equivalently, `gramschmidtL[a, b, n]` is the lower-triangular and `gramschmidtR[a, b, n]` is the upper-triangular Cholesky factor of the Hessian matrix $H(a, b, n)$.

```
Module[{n = 3, gsR, umat},
  gsR = gramschmidtR[a, b, n];
  umat = FullSimplify[CholeskyDecomposition[hessian[a, b, n]],
    a < b];
  TraditionalForm@
  Column[
    Riffle[Map[MatrixForm,
      {gsR, umat, Simplify[gsR - umat, a < b]}], ""]]]
```

$$\begin{pmatrix} \sqrt{b-a} & \frac{1}{2}\sqrt{b-a}(a+b) & \frac{1}{3}\sqrt{b-a}(a^2+ab+b^2) \\ 0 & \frac{(b-a)^{3/2}}{2\sqrt{3}} & \frac{(b-a)^{3/2}(a+b)}{2\sqrt{3}} \\ 0 & 0 & \frac{(b-a)^{5/2}}{6\sqrt{5}} \end{pmatrix}$$

$$\begin{pmatrix} \sqrt{b-a} & \frac{b^2-a^2}{2\sqrt{b-a}} & \frac{b^3-a^3}{3\sqrt{b-a}} \\ 0 & \frac{(b-a)^{3/2}}{2\sqrt{3}} & \frac{(b-a)^{3/2}(a+b)}{2\sqrt{3}} \\ 0 & 0 & \frac{(b-a)^{5/2}}{6\sqrt{5}} \end{pmatrix}$$

$$\begin{pmatrix} 0 & 0 & 0 \\ 0 & 0 & 0 \\ 0 & 0 & 0 \end{pmatrix}$$

□ Special Case: The Interval $[0, 1]$ for Hilbert Matrices

For $[a, b] = [0, 1]$, the Hessian matrix $H(a, b, n)$ is identical to the Hilbert matrix H_n of dimension $n \times n$:

$$H(0, 1, n)_{ij} = (H_n)_{ij} = \frac{1}{i+j-1}, \quad i, j = 1, \dots, n,$$

$$H_n := \begin{pmatrix} 1 & 1/2 & \dots & 1/(n-1) \\ 1/2 & 1/3 & \dots & 1/n \\ \vdots & \vdots & \ddots & \vdots \\ 1/(n-1) & 1/n & \dots & 1/(2n-1) \end{pmatrix}.$$

**With[{n = 3}, {MatrixForm[hessian[0, 1, n]],
MatrixForm[HilbertMatrix[n]]}]**

$$\left\{ \begin{pmatrix} 1 & \frac{1}{2} & \frac{1}{3} \\ \frac{1}{2} & \frac{1}{3} & \frac{1}{4} \\ \frac{1}{3} & \frac{1}{4} & \frac{1}{5} \end{pmatrix}, \begin{pmatrix} 1 & \frac{1}{2} & \frac{1}{3} \\ \frac{1}{2} & \frac{1}{3} & \frac{1}{4} \\ \frac{1}{3} & \frac{1}{4} & \frac{1}{5} \end{pmatrix} \right\}$$

Hilbert matrices are ill-conditioned already for dimensions $n \geq 4$ (i.e. have condition numbers greater than about 10,000) and soon reach the limits of 64-bit IEEE arithmetic.

Table[SingularValueList[N[HilbertMatrix[n]]], {n, 5}] //
TableForm

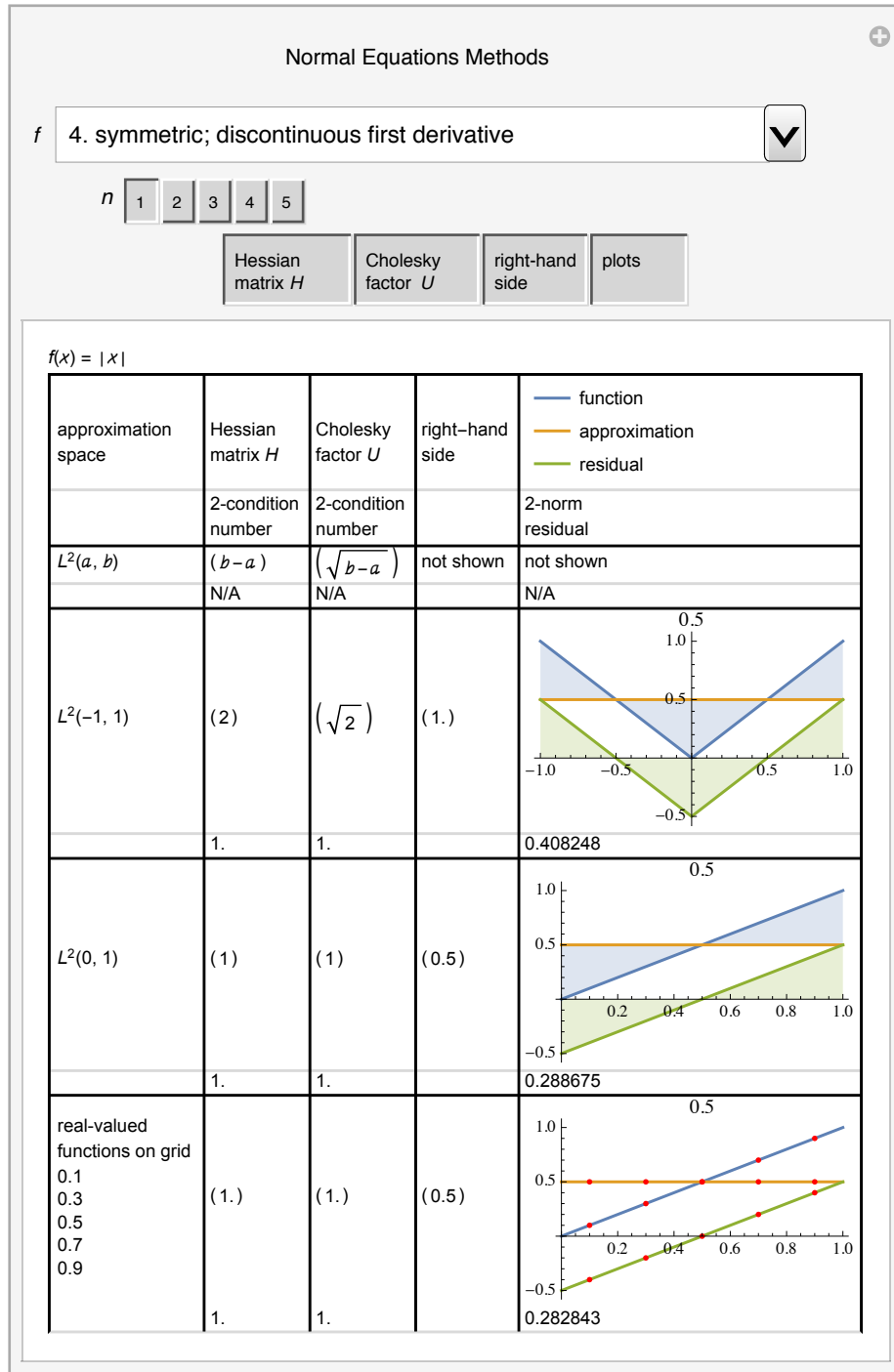
```
1.
1.26759      0.0657415
1.40832      0.122327      0.00268734
1.50021      0.169141      0.00673827      0.0000967023
1.56705      0.208534      0.0114075      0.000305898      3.28793 × 10-6
```

Table[ConditionNumber2[N[HilbertMatrix[n]]], {n, 8}]

```
{1., 19.2815, 524.057, 15513.7, 476607.,
 1.49511 × 107, 4.75367 × 108, 1.52576 × 1010}
```


□ Summary of Results for the Normal Equations Method

Here are the summary results. It takes some time for $n > 2$.



□ Characterization of the Normal Equations Solution

1. Continuous case: The $n \times n$ Hessian matrix $H(a, b, n)$ originates from the full-rank Vandermonde matrix $V(a, b, n, x)$ of the orthogonalization solutions by taking $V(a, b, n, x)^T V(a, b, n, x)$.
2. Discrete case: The $n \times n$ Hessian matrix $H_{\text{grid}}(a, b, n, m, s)$ comes from the full-rank Vandermonde matrix $V_{\text{grid}}(n, a, b, m, s)$ of the orthogonalization solutions by taking $h V_{\text{grid}}(n, a, b, m, s)^T V_{\text{grid}}(n, a, b, m, s)$.
3. Consequently,
 - a. H has full rank.
 - b. H is positive definite.
 - c. The 2-condition number of H is the square of the 2-condition number of the Vandermonde matrix V . This is the root cause of the inherent ill-conditioning of the normal equations.
4. $[a, b] = [0, 1]$ gives the exact Hilbert matrix of dimension n .
5. For other intervals $[a, b]$, the Hessian matrix $H(a, b, n)$ is close to a Hilbert matrix (“Hilbert-type matrix”).
6. Hilbert matrices and Hilbert-type matrices are very ill-conditioned already for dimensions greater than four and return extremely inaccurate numerical solutions.
7. For intervals symmetric around zero, $[-b, b]$, $H(-b, b, n)$ has a chessboard pattern of zeros, since the columns of $V(-b, b, n, x)$ are alternating orthogonal.
8. The normal equations are solved by Cholesky decomposition in the continuous and discrete settings.

■ Comparison of Solution Methods

□ Gram–Schmidt Orthonormalization versus Normal Equations

This performs Gram–Schmidt orthonormalization for a special case. For other cases, change the 6 in `samplefunctions`[[6, 1]] to an integer between 1 and 10.

```
f1 = samplefunctions[[6, 1]];
f1[x]
```

$$\frac{1}{2} \times \text{Abs}[x]$$

$$\mathbf{fp} = \mathbf{With}[\{\mathbf{n} = 2\}, \mathbf{fourierPolynomial}[\mathbf{f1}, \mathbf{a}, \mathbf{b}, \mathbf{n}, \mathbf{x}]]$$

$$\left[\begin{array}{ll} \frac{1}{12} (a^2 + 4 a b + b^2 - 6 a x - 6 b x) & a < 0 \ \&\& b \leq 0 \\ \frac{1}{12} (-a^2 - 4 a b - b^2 + 6 a x + 6 b x) & a \geq 0 \ \&\& b > 0 \\ \frac{a^5 - 8 a^2 b^3 + a^4 (b - 6 x) + b^4 (b - 6 x) + 4 a^3 b (-2 b + 3 x) + a b^3 (b + 12 x)}{12 (a - b)^3} & \mathbf{True} \end{array} \right.$$

Here are the results from the normal equations.

$$\mathbf{nep} = \mathbf{With}[\{\mathbf{n} = 2\}, \mathbf{normalPolynomial}[\mathbf{f1}, \mathbf{a}, \mathbf{b}, \mathbf{n}, \mathbf{x}]]$$

$$\left[\begin{array}{ll} \frac{1}{12} (a^2 + 4 a b + b^2 - 6 a x - 6 b x) & a < 0 \ \&\& b \leq 0 \\ \frac{1}{12} (-a^2 - 4 a b - b^2 + 6 a x + 6 b x) & a \geq 0 \ \&\& b > 0 \\ \frac{a^5 - 8 a^2 b^3 + a^4 (b - 6 x) + b^4 (b - 6 x) + 4 a^3 b (-2 b + 3 x) + a b^3 (b + 12 x)}{12 (a - b)^3} & \mathbf{True} \end{array} \right.$$

The two solutions agree both symbolically and in exact arithmetic.

$$\mathbf{FullSimplify}[\mathbf{fp} - \mathbf{nep}, a < b]$$

$$0$$

But there are numeric differences in IEEE 758 machine arithmetic.

$$\mathbf{With}[\{\mathbf{a} = 0, \mathbf{b} = 1, \mathbf{n} = 3\},$$

$$\mathbf{Simplify}[\mathbf{fourierPolynomial}[\mathbf{f1}, \mathbf{a}, \mathbf{b}, \mathbf{n}, \mathbf{x}] -$$

$$\mathbf{fourierPolynomial}[\mathbf{f1}, \mathbf{N@a}, \mathbf{N@b}, \mathbf{n}, \mathbf{x}]]]$$

$$5.68989 \times 10^{-16} - 3.66374 \times 10^{-15} x + 3.71925 \times 10^{-15} x^2$$

$$\mathbf{With}[\{\mathbf{a} = 0, \mathbf{b} = 1, \mathbf{n} = 3\},$$

$$\mathbf{Simplify}[\mathbf{fourierPolynomial}[\mathbf{f1}, \mathbf{a}, \mathbf{b}, \mathbf{n}, \mathbf{x}] -$$

$$\mathbf{normalPolynomial}[\mathbf{f1}, \mathbf{N@a}, \mathbf{N@b}, \mathbf{n}, \mathbf{x}]]]$$

$$7.40149 \times 10^{-17} - 4.44089 \times 10^{-16} x + 4.44089 \times 10^{-16} x^2$$

These differences come from the lower error amplification expressed in a lower 2-condition number,

```
Module[{cvandermonde, chessian},
  With[{a = 0, b = 1, n = 3},
    cvandermonde = ConditionNumber2[N[gramschmidtR[a, b, n]]];
    chessian = ConditionNumber2[N[hessian[a, b, n]]];
    {cvandermonde, chessian, Chop[chessian - cvandermonde^2]}
  ]
]

{22.8923, 524.057, 0}
```

The numerical solution via the Gram–Schmidt orthonormalization solution is usually more accurate than the normal equations solution.

□ QR Factorization versus Normal Equations on an Equally Spaced Grid

Here is the QR factorization. Again, for other cases, change the 3 in `samplefunctions` to an integer between 1 and 10.

```
f2 = samplefunctions[3, 1];
f2[x]

Sign[x]

qrpg = With[{a = 0, b = 1, m = 5, s = 0.5, n = 3},
  qrPolynomialgrid[f2, a, b, n, x, m, s]]

1. + 5.26625 × 10-16 x
```

Normal equations.

```
nepg = With[{a = 0, b = 1, m = 5, s = 0.5, n = 3},
  normalPolynomialgrid[f2, a, b, n, x, m, s]]

1.
```

The difference between the numerical solutions is due to the difference in how their rounding errors propagate.

```
qrpg - nepg

- 4.44089 × 10-16 + 5.26625 × 10-16 x
```

Because of the lower error amplification expressed in a lower 2-condition number, the numerical solution via QR factorization is more accurate than the normal equations solution.

```
Module[{cvandermonde, chessian},
  With[{a = 0, b = 1, m = 5, s = 0.5, n = 3},
    {
      cvandermonde = ConditionNumber2[
        N[vandermondegrid[n, a, b, m, s]]],
      chessian = ConditionNumber2[
        N[hessiagrid[a, b, n, m, s]]],
      Chop[chessian - cvandermonde^2]}
    ]
  ]

{25.4183, 646.092, 0}
```

□ Convergence of the Right-Hand Sides of Normal Equations

This calculates the convergence order for grids of powers of 2.

```
convergenceorder[errors_?MatrixQ] := If[
  Last[Dimensions[errors]] == 2,
  Transpose[{errors[[2 ;;, 1]],
    -Log[2, Quiet[Ratios[errors[[ ;;, 2]]]]]}]
]
```

Choose problem parameters. Yet again, for other cases, change the 7 in samplefunctions[[7, 1]] to an integer between 1 and 10.

```
f3 = samplefunctions[[7, 1]];
f3[x]
```

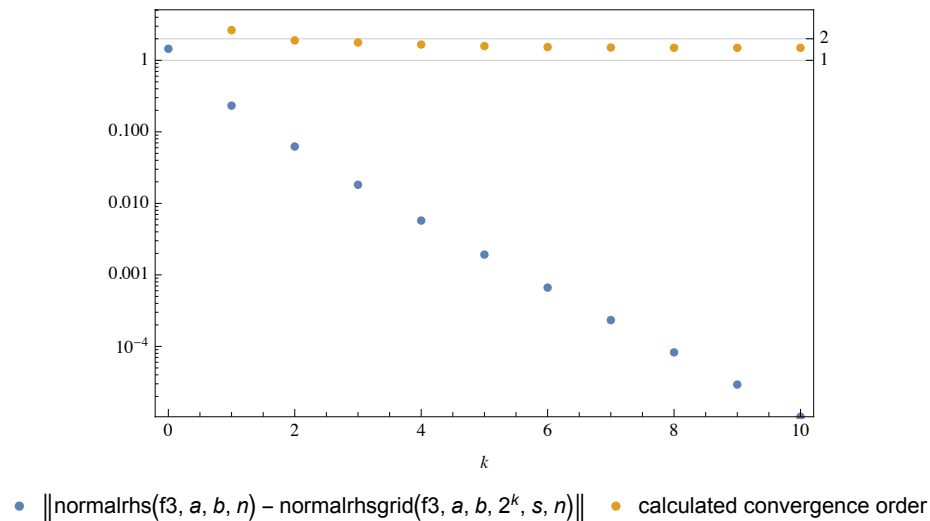
$$\sqrt{\text{Abs}[x]}$$

Here are the approximation errors for grids of powers of 2.

```
errors3 = With[{a = -1, b = 1, n = 3, s = 0.5},
  Table[
    {k, Norm[normalrhs[f3, a, b, n] -
      normalrhsgrid[f3, a, b, n, 2^k, s]]}, {k, 0, 10}]]

{{0, 1.45062}, {1, 0.232403}, {2, 0.0623052},
 {3, 0.0182}, {4, 0.00574758}, {5, 0.00192125},
 {6, 0.000663999}, {7, 0.000233287}, {8, 0.0000825335},
 {9, 0.0000292681}, {10, 0.0000103829}}
```

Here is the convergence order. To see the result for another function, select another sample function in `errors3`. Zero errors are suppressed.



For case 3, all but the first two elements of `errors3` are zero. For case 5, all but the first element of `errors3` are zero; therefore the logarithmic plots look incomplete.

Sample functions 3, 4, 5, 6 have discontinuities in the zeroth, first or second derivative; 7 and 8 have singularities in the first or second derivative. These sample functions and $x^j f(x)$ do not satisfy all the assumptions of Theorem 1. Therefore the convergence order of the right-hand side can be lower than 1 (respectively 2) as predicted by equation (47). Sample functions 1, 2, 9, 10 are infinitely often continuously differentiable; therefore they have maximum convergence order 1 (respectively 2) according to equation (47).

□ Convergence of the Normal Equations Solution

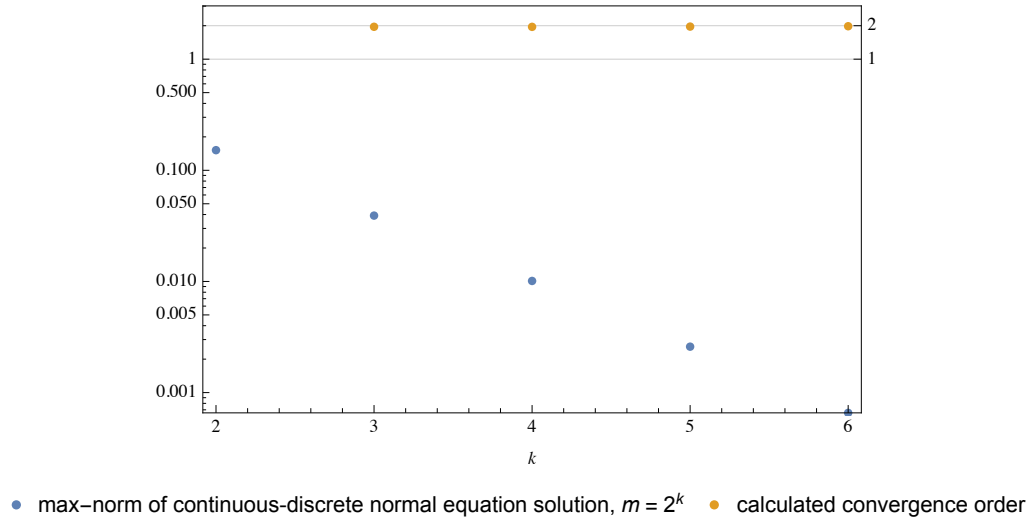
These are the approximation errors for grids of powers of 2.

```
f4 = samplefunctions[[9, 1]];
f4[x]
@Sin[π x]
```

This takes a few minutes.

```
errors4 = With[{a = -1., b = 1., n = 3, s = 0.5},
  Table[{k, Max[Abs[
    Map[
      normalPolynomial[f4, a, b, n, #] -
      normalPolynomialgrid[f4, a, b, n, #, 2^k, s] &,
      grid[a, b, 2^k, s]
    ]
  ]]}, {k, 2, 6}]]];
```

This plots the convergence order.



■ Relations between Solution Methods

Analyzing polynomial L^2 approximation, this article has systematically worked out the close relations between the solutions obtained by:

1. Gram–Schmidt orthonormalization and Fourier approximation
2. QR factorization
3. Normal equations

The interrelations are:

1. The Gram–Schmidt orthonormalization applied to the monomials $\{1, x, \dots, x^{n-1}\}$ on $[a, b]$ is reformulated as QR factorization of the $1 \times n$ Vandermonde matrix $V(a, b, n, x) = (1 \ x \ \dots \ x^{n-1})|_{[a,b]} = Q(a, b, n, x) R(a, b, n)$ with the following one-to-one correspondences.
 - a. The n columns of the matrix $Q(a, b, n, x)$ are the orthonormal polynomials on $[a, b]$.
 - b. The $n \times n$ upper-triangular matrix $R(a, b, n)$ is the transpose inverse of the coefficient matrix of the orthonormal polynomials $C(a, b, n)$.
 - c. The numerical condition number of $V(a, b, n, x)$ equals the numerical condition number of $C(a, b, n)$ with respect to the 2-norm.
 - d. The elements of $Q(a, b, n, x)^T f$ are the Fourier coefficients f_j , where $Q(a, b, n, x)^T : L^2(a, b) \rightarrow \mathbb{R}^n$ is the transpose operator.
 - e. The elements of $c = R(a, b, n)^{-1} Q(a, b, n, x)^T f$ are the coefficients of the Fourier polynomial expanded into monomials.

2. For $[a, b] = [-1, 1]$, Gram–Schmidt orthonormalization returns scalar multiples of the classical Legendre polynomials.
3. The $n \times n$ Hessian matrix H in the normal equations originates from the full-rank Vandermonde matrix V of the orthogonalization solutions by taking $V^T V$.
4. The 2-condition number of H is the square of the 2-condition number of the Vandermonde matrix V .
5. The upper Cholesky factor of H is identical to the upper-triangular $n \times n$ matrix R of the QR factorization of V in both the continuous and discrete cases.
6. H is identical to $R^T R$ of the QR factorization in both the continuous and discrete cases.
7. The QR and normal equations solutions on an equally spaced grid of m points in the interval $[a, b]$ converge to the solution in $L^2(a, b)$ with the same order as the error of the quadrature formulas.

■ Summary

This article has analyzed the polynomial approximation of a real-valued function f with respect to the least-squares norm in different settings:

- $L^2(a, b)$, $-\infty < a < b < \infty$
- $L^2(-1, 1)$
- $L^2(0, 1)$
- 2-norm on an equally spaced grid of m points in the interval $[a, b]$

Three different solution methods for this least-squares problem have been compared:

1. The orthogonalization solutions
 - a. The Gram–Schmidt orthonormalization in the continuous setting
 - b. The matrix QR factorization in the discrete setting
2. The normal equations solution in the continuous and discrete settings

All definitions and solution methods were implemented in Mathematica 11.1.

All solution characteristics and their relations were illustrated by symbolic or numeric examples and graphs.

■ Acknowledgments

The author thanks Alexandra Herzog and Jonas Gienger for critical proofreading and the anonymous referees for valuable improvements of the paper. The paper evolved from a lecture series in numerical analysis, given by the author for ESOC staff and contractors from 2012 to 2016, using Mathematica:

- Lecture #4: Linear Least-Squares Systems
- Lecture #6: Numerical Quadrature
- Lecture #9: Interpolation and Approximation of Functions, Curve and Surface Fitting

■ References

- [1] E. Hewitt and K. Stromberg, *Real and Abstract Analysis*, New York: Springer-Verlag, 1975.
- [2] K. Yosida, *Functional Analysis*, 5th ed., New York: Springer-Verlag, 1978.
- [3] P. J. Davis and P. Rabinowitz, *Methods of Numerical Integration*, 2nd ed., London: Academic Press, 1984.
- [4] G. H. Golub and C. F. Van Loan, *Matrix Computations*, 4th ed., Baltimore: The John Hopkins University Press, 2013.
- [5] E. W. Weisstein. "Vandermonde Determinant" from Wolfram MathWorld—A Wolfram Web Resource. mathworld.wolfram.com/VandermondeDeterminant.html.
- [6] J. D. Faires and R. Burden, *Numerical Methods*, 3rd ed., Pacific Grove, CA: Thomson/Brooks/Cole, 2003.

G. Gienger, "Polynomial L^2 Approximation," *The Mathematica Journal*, 2017. [dx.doi.org/doi:10.3888/tmj.19-2](https://doi.org/10.3888/tmj.19-2).

About the Author

2011–2016 ESA/ESOC research and technology management office
 2010–2011 ESOC navigation support office
 1987–2009 Mathematical analyst in ESOC Flight Dynamics division; supported more than 20 space missions
 1988 Received Dr. rer. nat. from Heidelberg University, Faculty of Science and Mathematics
 1984–1987 Teaching assistant at University of Giessen
 1982–1984 Research scientist at Heidelberg University
 1975–1981 Studied mathematics and physics at Heidelberg University

Gottlob Gienger

*European Space Operations Centre ESOC
 Research and Technology Management Office OPS-GX
 Robert-Bosch-Strasse 5, 64293 Darmstadt, Germany
 Retired staff
 Gottlob.Gienger@gmail.com*