Numerical Methods in Economics

MIT Press, 1998

Chapter 11 Notes: Projection Methods for Functional Equations

November 3, 2010

Functional Problems

- Many problems involve solving for some unknown function
  - Dynamic programming
  - Consumption and investment policy functions
  - Pricing functions in asset pricing models
  - Strategies in dynamic games
- The projection method is a robust method for solving such problems

An Ordinary Differential Equation Example

- Consider the differential equation
$$y' - y = 0, \quad y(0) = 1 \tag{11.1.1}$$
  - Solution is $y = e^x$.
  - We use projection methods to solve it for $0 \le x \le 3$.
- Key Distinction:
  - Finite difference methods solve a finite set of equations on a grid - they replace the continuous domain for x with a discrete set of x values
  - Projection methods find a *function* that approximately solves the *functional* equation (11.1.1) - they approximate the unknown function $y(x)$ with a function from a finite-dimensional space of functions.
- Define L
$$Ly \equiv y' - y \ . \tag{11.1.2}$$
  - L is an operator mapping functions to functions; domain is C^1 functions and range is C^0.
  - Define $Y = \{y(x) | y \in C^1, \ y(0) = 1\}$
  - (11.1.1) wants to find a $y \in Y$ such that $Ly = 0$.

- Approximate functions: consider family

$$\hat{y}(x; a) = 1 + \sum_{j=1}^{n} a_j x^j. \tag{11.1.3}$$

  - An affine subset of the vector space of polynomials.
  - Note that $\hat{y}(0; a) = 1$ for any choice of a, so $\hat{y}(0; a) \in Y$ for any a.

- Objective: find a s.t. $\hat{y}(x;a)$ "nearly" solves differential equation (11.1.1).
- Define *residual function*

$$R\,(x;a) \equiv L\hat{y} = -1 + \sum_{j=1}^{n} a_j(jx^{j-1} - x^j) \tag{11.1.4}$$

  - $R\,(x;a)$ is deviation of $L\hat{y}$ from zero, the target value.
  - A projection method adjusts a until it finds a "good" a that makes $R(x;a)$ "nearly" the zero function.
  - Different projection methods use different notions of "good" and "nearly."

Example:

- Consider

$$y' - y = 0, \quad y(0) = 1 \tag{11.1.1}$$

  for $x \in [0, 3]$ with

$$\hat{y}(x; a) = 1 + \sum_{j=1}^{3} a_j x^j$$

- Least Squares:
  - Find a that minimizes the total squared residual

$$\min_a \int_0^3 R(x; a)^2 \, dx. \tag{11.1.5}$$

  - Objective is quadratic in the a's with f.o.c.'s

$$\begin{pmatrix} 6 & \frac{9}{2} & \frac{-54}{5} \\ \frac{9}{2} & \frac{36}{5} & 0 \\ \frac{54}{5} & 0 & 41\frac{23}{35} \end{pmatrix} \begin{pmatrix} a_1 \\ a_2 \\ a_3 \end{pmatrix} = \begin{pmatrix} -3 \\ 0 \\ \frac{27}{2} \end{pmatrix}. \tag{11.1.6}$$

- Method of moments:
  - Idea: If $R(x;a)$ were zero, then $\int_0^3 R(x;a)\, f(x)\, dx = 0$ for all $f(x)$.
  - Use low powers of x to identify a via projection conditions

$$0 = \int_0^3 R(x;a)\, x^j\, dx \ , \quad j = 0, 1, 2. \tag{11.1.9}$$

  - Conditions reduce to linear system in a:

$$\begin{pmatrix} -3/2 & 0 & 27/4 \\ -9/2 & -9/4 & 243/20 \\ -45/4 & 81/10 & 243/10 \end{pmatrix} \begin{pmatrix} a_1 \\ a_2 \\ a_3 \end{pmatrix} = \begin{pmatrix} 3 \\ \frac{9}{2} \\ 6 \end{pmatrix} \tag{11.1.10}$$

- Galerkin
  - Idea: use basis elements, x, x^2, and x^3 in projection conditions
  - Form projections of R against the basis elements

$$0 = \int_0^3 R(x;a)\, x^j\, dx \ , \quad j = 1, 2, 3.$$

  - Another linear equation

- Collocation
  - Idea: If $R(x;a) = 0$ then it is zero at all x.
  - Specify a finite set of X and choose a so that $R(x;a)$ is zero $x \in X$. If $X = \{0, 3/2, 3\}$, the uniform grid, this reduces to linear equations

$$\begin{aligned} &R(0;a) = 0 &&= -1 + a_1 \\ &R(1.5;a) = 0 &&= -1 - \tfrac{1}{2}a_1 + \tfrac{3}{4}\, a_2 + \tfrac{27}{8}\, a_3 \\ &R(3;a)0 &&= -1 - 2a_1 \; - 3a_2 \end{aligned} \tag{11.1.11}$$

- Chebyshev Collocation
  - Idea: interpolation at Chebyshev points is best
  - Let

$$X = \left\{ \frac{3}{2}\,(\cos\frac{\pi}{6} + 1),\ \frac{3}{2},\ \frac{3}{2}\,(\cos\frac{5\pi}{6} + 1) \right\}$$

    the zeroes of $T_3(x)$ adapted to [0,3]
  - Reduces to linear equations $R(x_i;a) = 0$, $x_i \in X$.

Table 11.1: Solutions for Coefficients in (11.1.3)

Scheme:	a_1	a_2	a_3
Least Squares	1.290	-.806	.659
Galerkin	2.286	-1.429	.952
Chebyshev Collocation	1.692	-1.231	.821
Uniform Collocation	1.000	-1.000	.667
Optimal L_2	1.754	-.838	.779

Table 11.2: Projection Methods Applied to (11.1.2): L_2 errors of solutions

	Uniform	Chebyshev	Least		
n	Collocation	Collocation	Squares	Galerkin	Best poly.
3	5.3(0)	2.2(0)	3.2(0)	5.3(-1)	1.7(-1)
4	1.3(0)	2.9(-1)	1.5(-1)	3.6(-2)	2.4(-2)
5	1.5(-1)	2.5(-2)	4.9(-3)	4.1(-3)	2.9(-3)
6	2.0(-2)	1.9(-3)	4.2(-4)	4.2(-4)	3.0(-4)
7	2.2(-3)	1.4(-4)	3.8(-5)	3.9(-5)	2.8(-5)
8	2.4(-4)	9.9(-6)	3.2(-6)	3.2(-6)	2.3(-6)
9	2.2(-5)	6.6(-7)	2.3(-7)	2.4(-7)	1.7(-7)
10	2.1(-6)	4.0(-8)	1.6(-8)	1.6(-8)	1.2(-8)

Continuous-Time Life-Cycle Consumption Models

- Consider life-cycle problem

$$\begin{aligned} &\max_c \int_0^T e^{-\rho t} u(c)\, dt, \\ &\dot{A} \quad = rA + w(t) - c(t) \\ &A(0) \ = A(T) = 0 \end{aligned} \tag{10.6.10}$$

- Parameters
  - $u(c) = c^{1+\gamma}/(1+\gamma)$
  - $\rho = 0.05, r = 0.10, \gamma = -2$
  - $w(t) = 0.5 + t/10 - 4(t/50)^2$, and $T = 50$.

- The functions $c(t)$ and $A(t)$ must approximately solve the two point BVP

$$\begin{aligned} \dot{c}(t) &= -\tfrac{1}{2} c(t)(0.05 - 0.10), \\ \dot{A}(t) &= 0.1A(t) + w(t) - c(t), \\ A(0) &= A(T) = 0. \end{aligned} \tag{11.4.7}$$

- Approximation: degree 10 Chebyshev polys for $c(t)$ and $A(T)$:

$$\begin{aligned} A(t) &= \sum_{i=0}^{10} a_i T_i\left(\tfrac{t-25}{25}\right), \\ c(t) &= \sum_{i=0}^{10} c_i T_i\left(\tfrac{t-25}{25}\right), \end{aligned} \tag{11.4.6}$$

- Define the two residual functions

$$\begin{aligned} R_1(t) &= \dot{c}(t) - 0.025c(t) \\ R_2(t) &= \dot{A}(t) - \left(.1A(t) + \left(.5 + \tfrac{t}{10} - 4(\tfrac{t}{50})^2\right) - c(t)\right). \end{aligned} \tag{11.4.8}$$

- Choose a_i and c_i so that $R_1(t)$ and $R_2(t)$ are nearly zero and $A(0) = A(T) = 0$ hold.
  - Boundary conditions impose two conditions
  - Need 20 more conditions to determine the 22 unknown coefficients.
  - Use 10 collocation points on $[0, 50]$: the 10 zeros of $T_{10}(t - 25/25)$

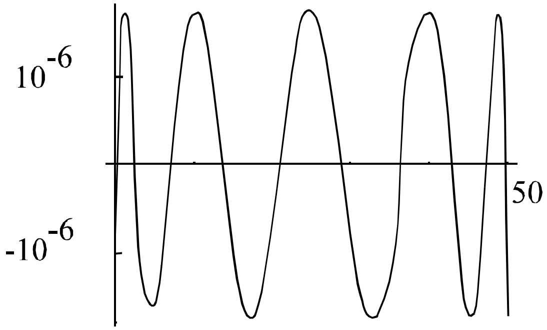
$$\mathcal{C} \equiv \{0.31, 2.72, 7.32, 13.65, 21.09, 28.91, 36.35, 42.68, 47.28, 49.69\}$$

  - Choose the a_i and c_i coefficients, which solve

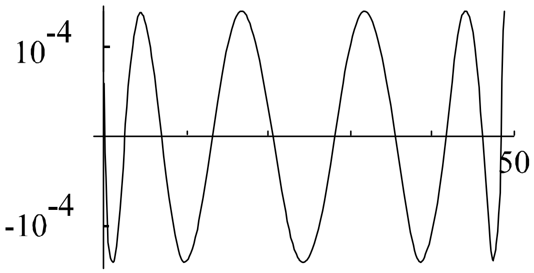
$$\begin{aligned} R_1(t_i) &= 0, \;\; t_i \in \mathcal{C}, i = 1, ..., 10, \\ R_2(t_i) &= 0, \;\; t_i \in \mathcal{C}, i = 1, ..., 10, \\ A(0) \;\; &= \textstyle\sum_{i=1}^{10} a_i(-1)^i = 0, \\ A(50) &= \textstyle\sum_{i=1}^{10} a_i = 0. \end{aligned} \tag{11.4.9}$$

  - 22 linear equations in 22 unknowns.
  - The system is nonsingular; therefore there is a unique solution.
- The true solution to the system (11.4.7) can be solved since it is a linear problem.

- Residuals:



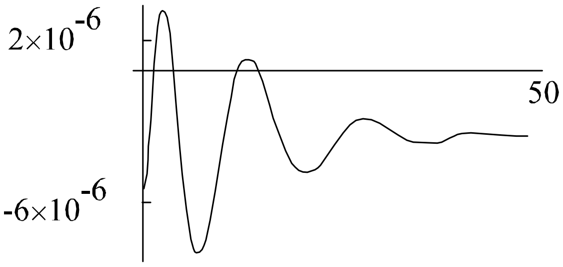
$\dot{c}$ equation residuals



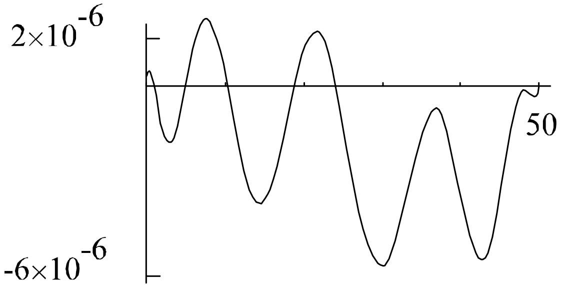
$\dot{A}$ equation residuals

- Note:
  - Equioscillation in residuals
  - Small size of residuals

- Errors



relative consumption errors



relative asset errors

- Note:
  - Lack of equioscillation in errors
  - Small size of errors
  - Errors are roughly same size as residuals

Simple Example: One-Sector Growth

- Consider

$$\max_{c_t} \sum_{t=1}^{\infty} \beta^t u(c_t)$$
$$k_{t+1} = f(k_t) - c_t$$

- Optimality implies that c_t satisfies

$$u'(c_t) = \beta u'(c_{t+1}) f'(k_{t+1})$$

- Problem: The number of unknowns $c_t, \ t = 1, 2, ...$ is infinite.
- **Step 0:** Express solution in terms of an unknown function

$$c_t = C(k_t) : \text{consumption function}$$

  - Consumption function $C(k)$ must satisfy the functional equation:

$$\begin{aligned} 0 &= u'(C(k)) - \beta u'(C(f(k) - C(k))) f'(f(k) - C(k)) \\ &\equiv (\mathcal{N}(C))(k) \end{aligned}$$

  - This defines the operator

$$\mathcal{N} : C^0_+ \to C^0_+$$

  - Equilibrium solves the operator equation

$$0 = \mathcal{N}(C)$$

- **Step 1:** Create approximation:
  - Find
    $$\widehat{C} \equiv \sum_{i=0}^{n} a_i k^i$$
    which "nearly" solves
    $$\mathcal{N}\ (\widehat{C}) = 0$$
  - Convert an infinite-dimensional problem to a finite-dimensional problem in R^n
    - No discretization of state space
    - A form of discretization, but in spectral domain
- **Step 2:** Compute Euler equation error function:
  $$R\ (k;\ \vec{a}) = u'(\widehat{C}(k)) - \beta u'(\widehat{C}(f(k) - \widehat{C}(k)))f'(f(k) - \widehat{C}(k))$$

$\cdots$

- **Step 3**: Choose $\vec{a}$ to make $R(\cdot;\vec{a})$ "small" in some sense**:**
  - Least-Squares: minimize sum of squared Euler equation errors

    $$\min_{\vec{a}} \int R(\cdot;\vec{a})^2 dk$$

  - Galerkin: zero out weighted averages of Euler equation errors

    $$P_i(\vec{a}) \equiv \int R(k;\vec{a})\psi_i(k)dk = 0, \;\; i = 1, \cdots, n$$

    for n weighting functions $\psi_i(k)$.
  - Collocation: zero out Euler equation errors at $k \in \{k_1, k_2, \cdots, k_n\}$:

    $$P_i(\vec{a}) \equiv R(k_i;\vec{a}) = 0 \;, \;\; i = 1, \cdots, n$$

- Details of $\int ...dk$ computation:
  - Exact integration seldom possible in nonlinear problems.
  - Use quadrature formulas – they tell us what are *good* points.
  - Monte Carlo – often mistakenly used for high–dimension integrals
  - Number Theoretic methods – best for large dimension
- Details of solving $\vec{a}$:
  - Jacobian, $\vec{P}_{\vec{a}}(\vec{a})$, should be well-conditioned
  - Newton's method is quadratically convergent since it uses Jacobian
  - Functional iteration and time iteration ignore Jacobian and are linearly convergent.
  - Homotopy methods are almost surely globally convergent
  - Least squares may be ill-conditioned (that is, be flat in some directions).

Bounded Rationality Accuracy Measure

Consider the one-period relative Euler equation error:

$$E(k) = 1 - \frac{(u')^{-1}\left(\beta u'\left(C\left(f(k) - C(k)\right)\right) f'\left(f(k) - C(k)\right)\right)}{C(k)}$$

- Equilibrium requires it to be zero.
- $E(k)$ is measure of optimization error
  - 1 is unacceptably large
  - Values such as .00001 is a limit for people.
  - $E(k)$ is unit-free.
- Define the L^p, $1 \leq p < \infty$, *bounded rationality accuracy* to be

$$\log_{10} \parallel E(k) \parallel_p$$

- The L^∞ error is the maximum value of $E(k)$.

Numerical Results

- Machine: Compaq 386/20 w/ Weitek 1167
- Speed: Deterministic case: < 15 seconds
- Accuracy: Deterministic case: 8th order polynomial agrees with 250,000–point discretization to within 1/100,000.

General Projection Method

- **Step 0:** Express solution in terms of unknown functions

$$0 = \mathcal{N}(h)$$

- The $h(x)$ are decision and price rules expressing the dependence on the state x
  - consumption as a function of wealth
  - aggregate investment as a function of current capital stock and productivity
  - an individual's asset trading as a function of public and his private information
  - equilibrium price as a function of.all information
  - firm investment as a function of his and rivals' current capital stock
- The functions h express
  - agents on demand curve
  - firms on their product supply and factor demand curve
  - market clearing
  - value functions from dynamic programming problems
  - value functions in dynamic games
  - laws of motion
  - Bayesian updating and\or regression learning rules
- The collection of conditions $0 = \mathcal{N}(h)$ express equilibrium.

- **Step 1:** Choose space for approximation:
  - Basis for approximation for h:

    $$\{\varphi_i\}_{i=1}^{\infty} \equiv \Phi$$

  - Norm:

    $$\langle \cdot, \cdot \rangle : C_+^0 \times C_+^0 \to R$$

    basis should be complete in space of C_+^0 functions basis should be orthogonal w.r.t. $\langle \cdot, \cdot \rangle$ norm and basis should be easy to compute norm and basis should be "appropriate" for problem norms are often of form $\langle f, g \rangle = \int_D f(x)g(x)w(x)dx$ for some $w(x) > 0$
  - Goal: Find $\widehat{h}$ which "nearly" solves $\mathcal{N}\ (\widehat{h}) = 0$

    $$\widehat{h} \equiv \sum_{i=1}^{n} a_i\ \varphi_i$$

  - We have converted an infinite-dimensional problem to a problem in R^n
    - * No discretization of state space.
    - * Instead, discretize in a functional (spectral) domain.

- Example Bases:
  - $\Phi = \{1, k, k^2, k^3, \cdots\}$
  - $\Phi = \{\sin\ k,\ \sin\ 2k, \cdots\}$: Fourier – (periodic problems)
  - $\varphi_n = T_n(x)$: Chebyshev polynomials – (for smooth, nonperiodic problems)
  - Legendre polynomials
  - Step functions
  - Tent functions
  - B-Splines (smooth generalizations of step and tent functions)
  - Step functions are also finite element methods, but seldom used outside of economics.
- Nonlinear generalization
  - For some parametric form, $\Phi(x; a)$

    $$\widehat{h}(x; a) \equiv \Phi(x; a)$$

  - Examples:
    - Neural networks
    - Rational functions
  - Goal: Find an

    $$\widehat{h} \equiv \Phi(x; a)$$

    which "nearly" solves $\mathcal{N}\left(\widehat{h}\right) = 0$
  - Promising direction but tools of linear functional analysis and approximation theory are not available.

- **Step 2:** Compute residual function:

$$R(\cdot, a) = \widehat{\mathcal{N}}(\widehat{h}) \doteq \mathcal{N}(\widehat{h}) \doteq \mathcal{N}(h)$$

- **Step 3:** Choose $\vec{a}$ so that $R(\cdot;\vec{a})$ is “small” in $\langle \cdot, \cdot \rangle$.
  - Alternative Criteria:
    - Least-Squares

$$\min_{\vec{a}} \ \langle R(\cdot;\vec{a}), \ R(\cdot;\vec{a}) \rangle$$

    - Galerkin

$$P_i(\vec{a}) \equiv \langle R(\cdot;\vec{a}), \ \varphi_i \rangle \ = 0, i = 1, \cdots, n$$

    - Method of Moments

$$P_i(\vec{a}) \equiv \left\langle R(\cdot;\vec{a}), k^{i-1} \right\rangle = 0 \ , \ i = 1, \cdots, n$$

    - Collocation

$$P_i(\vec{a}) \equiv R(k_i;\vec{a}) = 0 \ , \ i = 1, \cdots, n, \ k_i \in \{k_1, k_2, \cdots, k_n\}$$

    - Orthogonal Collocation (a.k.a. Pseudospectral)

$$P_i(\vec{a}) \equiv R(k_i;\vec{a}) = 0 \ , \ i = 1, \cdots, n, \ k_i \in \{k : \varphi_n(k) = 0\}$$

- Details of $\langle \cdot, \cdot \rangle$ computation:
  - Exact integration seldom possible in nonlinear problems.
  - Use quadrature formulas – they tell us what are *good* points.
  - Monte Carlo – often mistakenly used for high–dimension integrals
  - Number Theoretic methods – best for large dimension
- Details of solving $\vec{a}$:
  - Jacobian, $\vec{P}_{\vec{a}}(\vec{a})$, should be well-conditioned.
  - Newton's method is quadratically convergent since it uses Jacobian; functional iteration (e.g., parameterized expectations) and time iteration ignore Jacobian and are linearly convergent.
  - If Φ is orthogonal w.r.t. $\langle \cdot, \cdot \rangle$, then Galerkin method uses orthogonal projections, helping with conditioning.
  - Least squares uses

    $$\left\langle R, \frac{\partial R}{\partial a_i} \right\rangle = 0$$

    projection conditions, which may lead to ill-conditioning.

New Method: L1 Minimization

- Recent work of Judd, Lontzek, and Michelangeli.
- Given residual function $R(x, a)$, choose a by solving

$$\min_a \int |R(x; a)| \, dx.$$

  - Operationalize by solving

$$\begin{aligned} \min_a \sum_i \lambda_i \, dx \\ -\lambda_i \leq R(x_i; a) \leq \lambda_i \\ 0 \leq \lambda_i \end{aligned}$$

  - Can impose conditions on a, such as imposing a shape on the unknown function.
  - Initial guess is less important.
  - There will always be a solution; most solvers should be able to find one since the λ parameters enter the constraints and objective linearly

Convergence Properties of Galerkin Methods

- Zeidler (1989): If the nonlinear operator $\mathcal{N}$ is monotone, coercive, and satisfies a growth condition then Galerkin method proves existence and works numerically.

- Krasnosel'skii and Zabreiko (1984): If $\mathcal{N}$ satisfies certain degree conditions, then a large set of projection methods (e.g., Galerkin methods with numerical quadrature) converge.

- Convergence is neither sufficient nor necessary
  - Usually only locally valid
  - Convergence theorems don't tell you when to stop.
  - Non-convergent methods are no worse if they satisfy stopping rules

A Partial Differential Equation Example

- Consider the simple heat equation

$$\theta_t - \theta_{xx} = 0$$

  - Domain $0 \leq x \leq 1, \quad 0 \leq t \leq 1.$
  - Initial condition $\theta(x, 0) = \sin \pi x$
  - Boundary conditions $\theta(0, t) = 0\ , \quad \theta(1, t) = 0, 0 \leq t \leq 1.$

- Unique solution is $\theta(x, t) = e^{-\pi^2 t} \sin \pi x.$

- Projection approach.
  - Form polynomial approximation

$$\hat{\theta}(x,t) = \theta_0(x) + \sum_{i=1}^{n} \sum_{j=1}^{m} a_{ij}\,(x - x^i)\,t^j .$$

    - Initial condition is absorbed in

$$\theta_0(x) = \sin \pi x$$

    - Boundary condition is automatically true since approximation is weighted sum of $x - x^j$ terms, which is zero at $x = 0, 1$.
    - A better choice may be to use orthogonal polynomials ϕ and ψ in $\sum_{i=1}^{n} \sum_{j=1}^{m} a_{ij}\,\phi_i(x)\,\psi_j\,(t)$ in x and t - e.g., Legendre polynomials adapted to $[0, 1]$.
  - Residual is a function of both space and time

$$R(x,t) = -\theta_{0xx}(x) + \sum_{i=1}^{n} \sum_{j=1}^{m} (a_{ij}\,(x - x^i)\,jt^{j-1} - a_{ij}\,(-i)(i-1)x^{i-2}\,t^j). \tag{1}$$

  - The nm unknown coefficients, a_{ij}, are fixed by the nm projection conditions

$$\big\langle R(x,t),\ \psi_{ij}(x,t)\big\rangle = 0, \qquad i = 1, \cdots, n,\ \ j = 1, \cdots, m, \tag{2}$$

    where $\psi_{ij}(x,t) = (x - x^i)t^j$ is a collection of nm basis functions.
  - Equations (2 form a system of linear algebraic equations in the unknown coefficients a_{ij}. System is better conditioned if we use orthogonal polynomials.

Computing Conditional Expectations

- Many economics problems need to compute conditional expectation functions.
- The *conditional expectation of* Y *given* X, denoted $E\{Y|X\}$, is a function of X, $\psi(X)$, such that

$$E\left\{\left(Y-\psi\left(X\right)\right)\,g\left(X\right)\right\}=0 \tag{11.6.1}$$

  for all continuous functions g.

  - Prediction error $Y-\psi(X)$ is uncorrelated with all functions of X.
  - We seek a function $\widehat{\psi}(X)$ which approximates $E\{Y|X\}$.

- Use projection method to approximate $\widehat{\psi}(X)$

  - Construct approximation scheme

$$\widehat{\psi}(X;a)=\sum_{i=0}^{n}a_i\varphi_i(X), \tag{11.6.2}$$

  - We now need to find the a coefficients in $\widehat{\psi}$.
  - Assume (WLOG) there is a r. v. Z such that $Y=g(Z)$ and $X=h(Z)$.
  - The least squares coefficients a solve

$$\min_a E\left\{\left(\psi\left(h\left(Z\right);a\right)-g(Z)\right)^2\right\}. \tag{11.6.3}$$

- Monte Carlo approach
  - Generate a sample of (Y, X) pairs, $\{(y_i, x_i) \mid i = 1, \cdots, N\}$
  - Regress the values of Y on X, solving the least squares problem

$$\min_a \sum_i \left(\psi\left(x_i; a\right) - y_i\right)^2 . \tag{11.6.4}$$

- Projection method
  - For all i, the projection condition $E\{(g\left(Z\right) - \psi(h\left(Z\right)))\, \varphi_i(h\left(Z\right))\} = 0$.
  - Fix coefficients a by imposing $n + 1$ projection conditions

$$E\left\{(g(Z) - \widehat{\psi}(h(Z); a))\, \varphi_i(h(Z))\right\} = 0,\ \ i = 0, ..., n. \tag{11.6.5}$$

  - (11.6.5) is a linear equation in the a coefficients.
  - Use deterministic methods to compute each integral

- Example:
  - Let $Y, W \sim U[0,1]$, $X = \varphi\left(Y, W\right) = \left(Y + W + 1\right)^2$
  - $E\{Y|X\} = (X^{1/2} - 1)/2$.
  - Monte Carlo
    - Produce 1,000 (y, w) pairs, and compute $x_i = \varphi(y_i, w_i)$.
    - Regress y on $1, x, x^2, x^3$, and x^4, producing

      $$\widehat{\psi}_{MC}(x) = -0.1760 + 0.2114x - 0.0075x^2 - 0.0012x^3 + 0.0001x^4.$$

    - The L^2 norm of $\widehat{\psi}_{MC} - \psi$ is 0.0431.
  - Projection method
    - Project prediction error $\widehat{\psi}(\varphi(y, w); a) - y$ against moments of x:

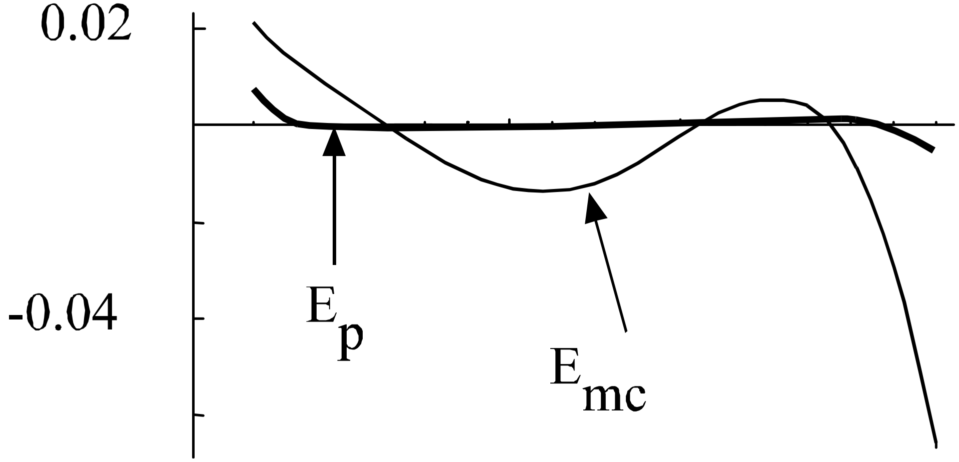
      $$\int_0^1 \int_0^1 (\widehat{\psi}(\varphi(y, w); a) - y)\ \varphi(y, w)^k\ dw\ dy = 0,\ \ k = 0, 1, 2, 3, 4$$

    - Linear system of equations in the unknown coefficients a.
    - Use quadrature for integrals; don't need 1000 points.
    - The solution implies

      $$\widehat{\psi}_P = -0.2471 + 0.2878x - 0.0370x^2 + 0.0035x^3 - 0.0001x^4.$$

    - The L^2 norm of $\widehat{\psi}_P - \psi$ is 0.0039

- Comparison:
  - $\widehat{\psi}_P$ error is ten times less than the L^2 error of the $\widehat{\psi}_{MC}$
  - $\widehat{\psi}_P$ is faster to compute than $\widehat{\psi}_{MC}$



- Conditional expectations are linear operators
  - Projection method reduces conditional expectations to linear problems combined with quadrature
  - No need to resort to Monte Carlo