Dynamic Stochastic General Equilibrium Models

Projection Method Very Short

Willi Mutschler

EBERHARD KARLS UNIVERSITÄT TÜBINGEN

Dynare Team

simple 1-sector model

Lifetime utility:

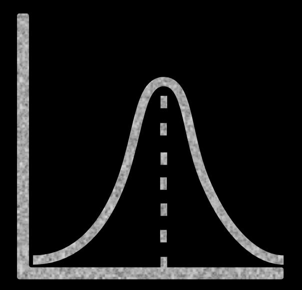
$$E_0 \sum_{t=0}^{\infty} \beta^t \frac{c_t^{1-\tau}}{1-\tau}$$

Budget restriction:

$$c_t + \underbrace{k_t - (1-\delta)k_{t-1}}_{i_t} = z_t k_{t-1}^{\alpha}$$

Productivity:

$$\ln(z_t) = \rho \ln(z_{t-1}) + \varepsilon_t$$



Euler-equation:

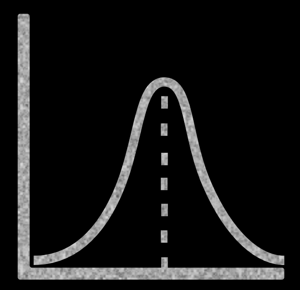
$$c_t^{-\tau} = \beta E_t \left[c_{t+1}^{-\tau}(1 - \delta + \alpha z_{t+1} k_t^{\alpha-1}) \right]$$

solution concept

Point of departure: initial states k_0, z_0

Goal: Find (recursive) decision rules, so-called *policy functions*, for the optimal paths of next periods decisions given shocks:

$$\ln(z_{t+1}) = \rho \ln(z_t) + \varepsilon_{t+1}$$



$$k_{t+1} = (1-\delta)k_t + z_{t+1}k_t^{\alpha} - c_{t+1}$$

$$c_{t+1} = C(k_t, z_t) = \ ?$$

special case: $\tau = \delta = 1$:

$$c_{t+1} = (1-\alpha\beta) z_t^{\rho} k_t^{\alpha}$$

$$k_{t+1} = \alpha\beta z_t^{\rho} k_t^{\alpha}$$

In general: how can we approximate C?

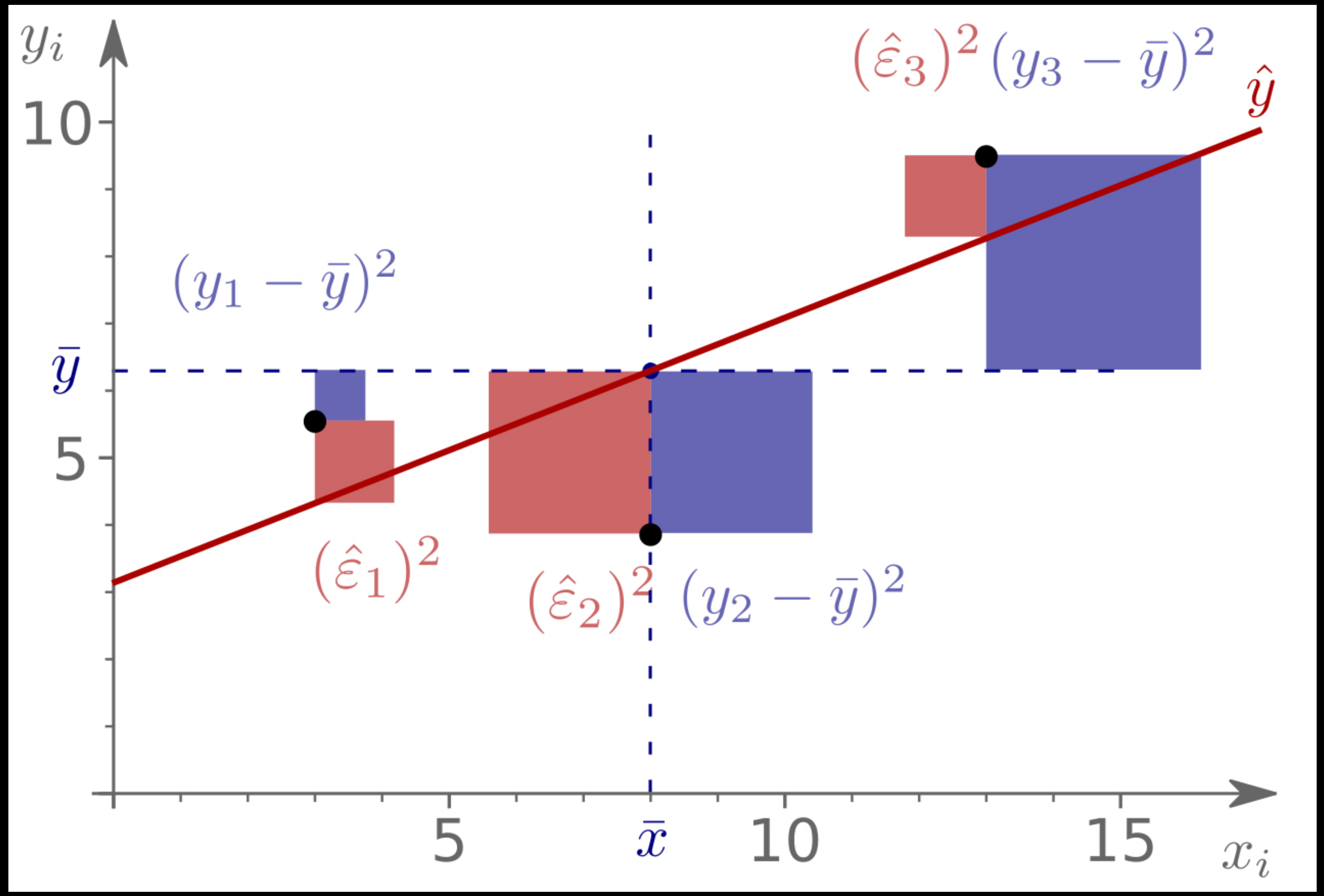
What is a *good* approximation?

- Define Euler-residual:
  $R = -c_t^{-\tau} + E_t[\beta c_{t+1}^{-\tau}(1 - \delta + \alpha z_{t+1} k_t^{\alpha-1})]$

- Exact solution: $R = 0$

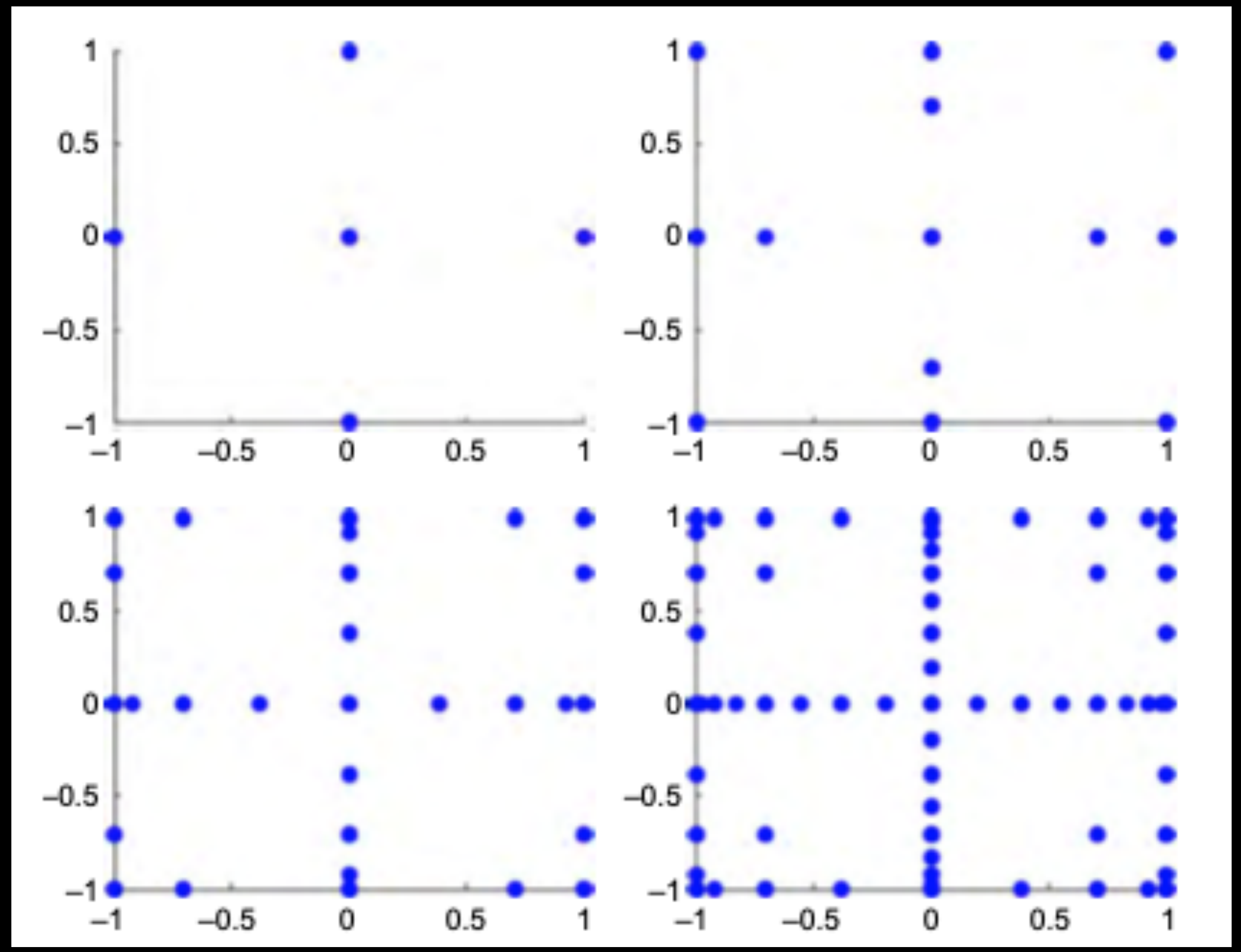
- Approximate solution $\hat{C}$: $R \lesseqgtr 0$

- Construct $\hat{C}(k_t, z_t; \theta)$ by choosing θ, such that we minimize *sum-of-squared-residuals*

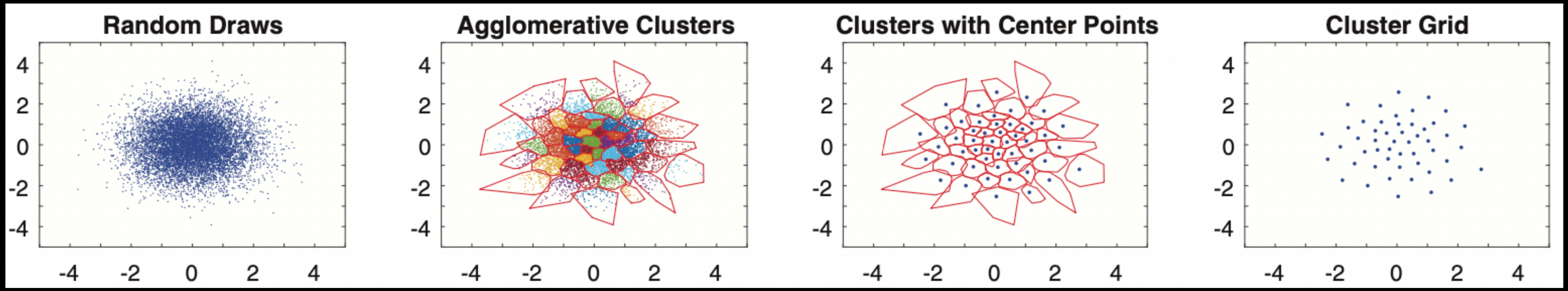


projection method

1. Choose grid $\{k_m, z_m\}_{m=1,\ldots,M}$ to construct $\hat{C}$



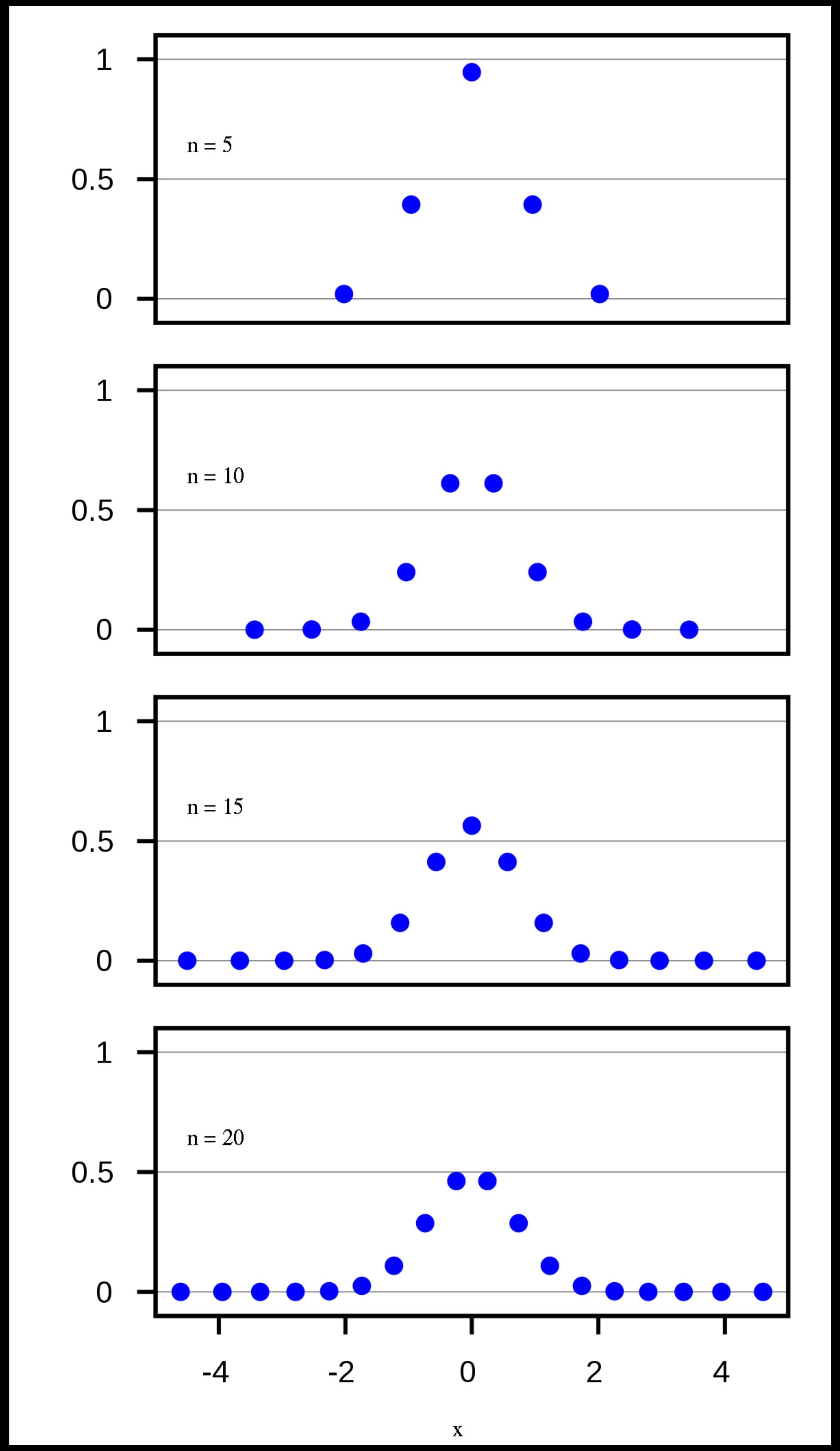
Traditional



Cluster Grid

projection method

2. Choose J nodes and weights for conditional expectation (*Gauss-Hermite-Quadrature*)

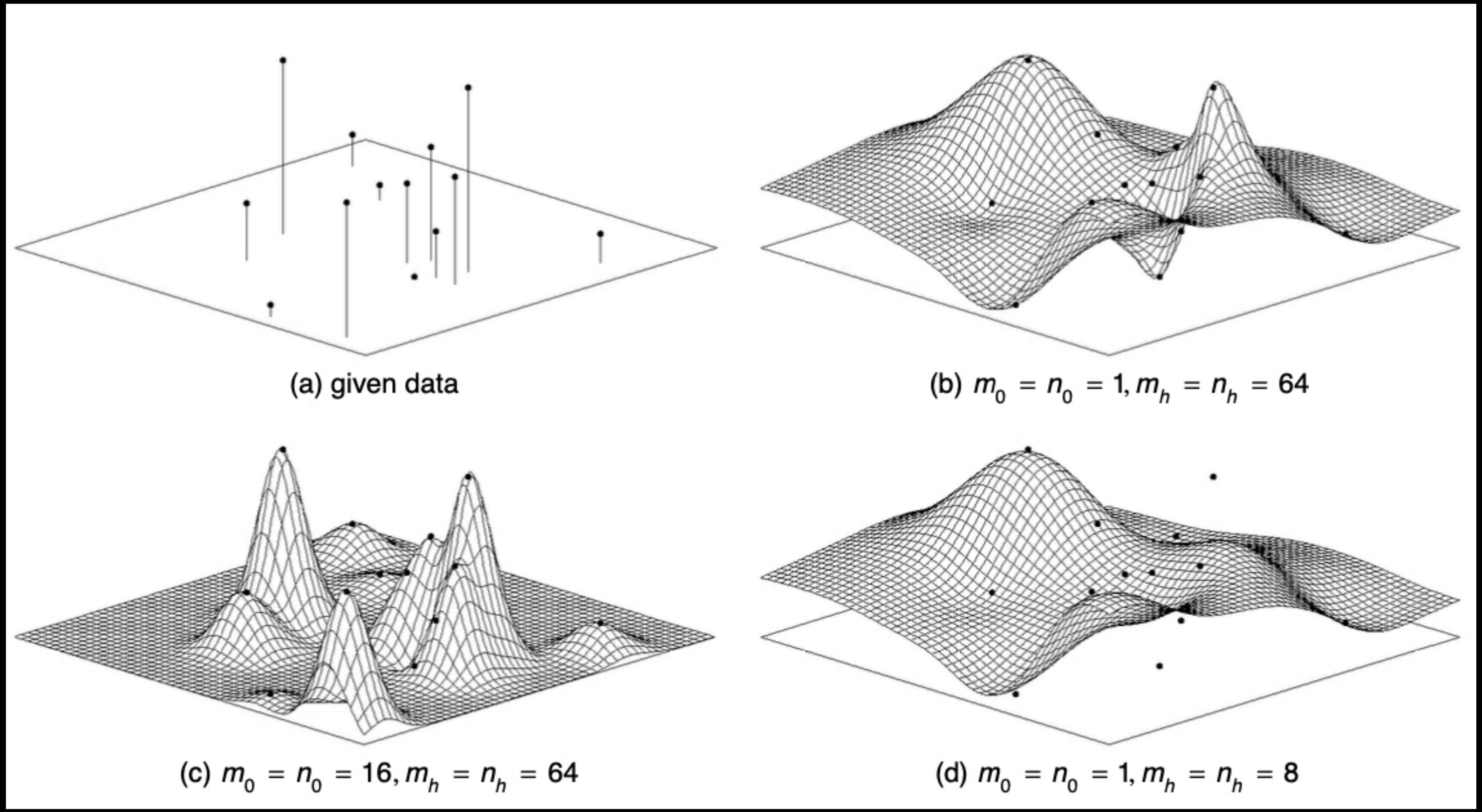


projection method

3. Choose functional form for $\hat{C}$ with coefficients θ, for example:

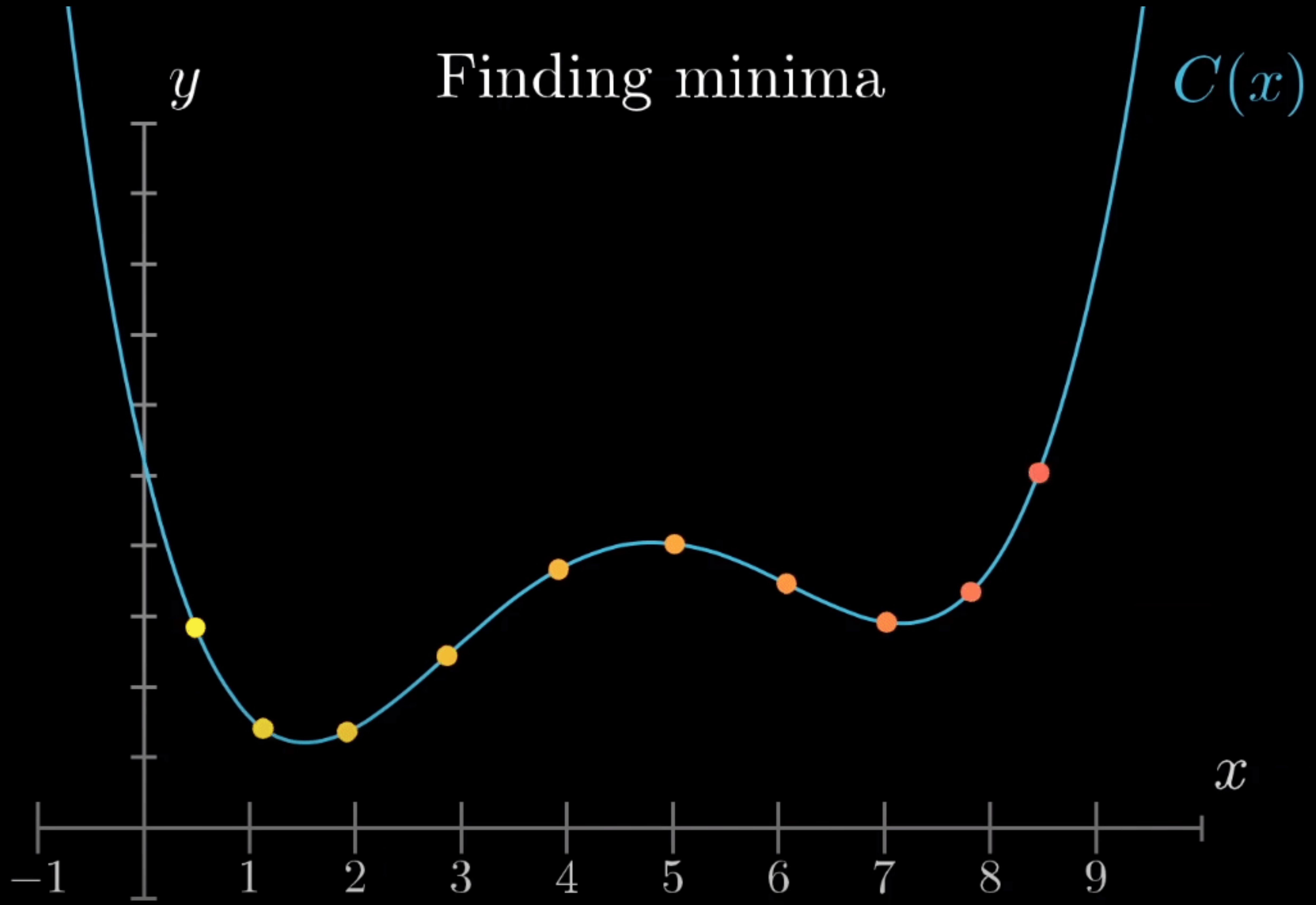
- polynomials: $\hat{C}(k,z) = \theta_0 + \theta_k(k-\bar{k}) + \theta_z(z-\bar{z}) + \frac{1}{2}\left(\theta_{kk}(k-\bar{k})^2 + 2\theta_{kz}(k-\bar{k})(z-\bar{z}) + \theta_{zz}(z-\bar{z})^2\right) + \frac{1}{6}\ldots$

- B-Spline functions:

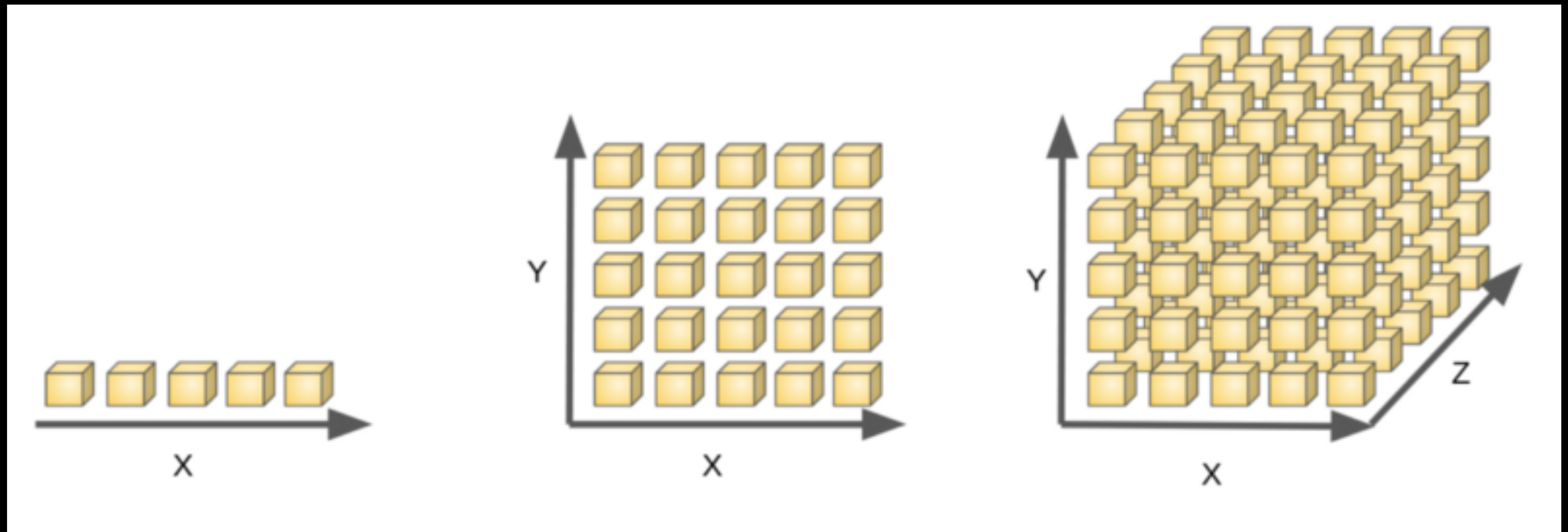


projection method

4. optimization problem: find coefficients θ that minimize sum of squared residuals



Curse of Dimensionality



- Number of grid points increases exponentially with number of state variables
- Number of integration nodes and weights increases exponentially with number of shocks
- Spline interpolation does not work with more than two variables
- Costs (runtime, memory, reliability) in the optimization algorithm explode

Perturbation

Sparse Polynomial Interpolation

Smolyak Sparse Grids

Derivative Free Fixed Point Iteration

Non Product Monomial Integration

Low Discrepancy Sequences