Package ‘capn’

August 22, 2026

Type Package

Title Capital Asset Pricing for Nature

Version 2.0.0

Date 2026-8-19

Author Seong D. Yun [aut, cre],
Eli P. Fenichel [aut, ctb],
Joshua K. Abbott [aut, ctb]

Maintainer Seong D. Yun <yunsd2004@gmail.com>

Description Implements the natural capital asset pricing for nature (CAPN) approach, using a collocation or iteration method and associated functions for dynamic programs. Development of this package was supported by the Knobloch Family Foundation and Lenfest Ocean Program Contract ID 00029728.

Depends R (>= 4.5.0)

License GPL (>= 2)

LazyData False

Suggests knitr, rmarkdown

VignetteBuilder knitr

NeedsCompilation no

Repository CRAN

Date/Publication 2026-08-22 15:10:46 UTC

Contents

`AFY` *Abbott–Fenichel–Yun Data (1-D Stochastic Example)*

Description

`AFY` contains approximation and simulation data used to replicate the one-dimensional stochastic example in Abbott, Fenichel, and Yun (2026), which extends the classic renewable resource model of Pindyck (1984). The dataset illustrates the natural capital asset pricing (CAPN) framework under uncertainty.

Usage

```
data("AFY")
```

Format

A data.frame containing simulation and approximation data for a one-dimensional stochastic renewable resource model:

simData A data.frame with the following components:

- `stock`: Stock levels.
- `mu.d`: Deterministic stock growth rate.
- `profit.d`: Deterministic profit.
- `mu.s`: Drift term of the stochastic stock process.
- `profit.s`: Profit under stochastic dynamics.
- `sigs`: Variance term of the stochastic process.

param A data.frame of model and approximation parameters:

- `r`: Intrinsic growth rate (=0.5)
- `K`: Carrying capacity (=100)
- `b`: Demand parameter (=1)
- `eta`: elasticity of demand (=1/2)
- `c`: cost parameter (=5)

- `gamma`: elasticity of marginal cost (=2)
- `delta`: discount rate (=0.05)
- `order`: Chebyshev polynomial order (=30)
- `upperK`: Upper bound of Chebyshev nodes (= 114.3)
- `lowerK`: Lower bound of Chebyshev nodes (= 20)
- `nodes`: Number of Chebyshev nodes (= 30)

Details

Following Abbott, Fenichel, and Yun (2026) and Pindyck (1984), the model is defined as follows.

Demand function:

$$q(p) = bp^{-\eta}$$

Cost function:

$$c(s) = cs^{-\gamma}$$

Biological growth function:

$$f(s) = rs\left(1 - \frac{s}{K}\right)$$

Stock dynamics:

$$ds = [f(s) - q(s)]\,dt + \sigma s\,dz$$

Net benefit function:

$$\int_0^q p(z)\,dz - c(s)q(s) = -\frac{b^2}{q} - \frac{c}{s^2}q$$

Closed-form value function:

$$V(s) = -\frac{\phi}{s} - \frac{\phi r}{\delta K}$$

where

$$\phi = \frac{2b^2 + 2b\sqrt{b^2 + c(r + \delta - \sigma^2)}}{(r + \delta - \sigma^2)^2}$$

Optimal harvest (catch) function:

$$q(s) = b(\phi + c)^{-1/2}s$$

Parameter values:

- $r = 0.5$
- $K = 1$ (=100% in the percentage replication)
- $b = 1$
- $\eta = 0.5$
- $c = 5$
- $\gamma = 2$
- $\delta = 0.05$

References

Abbott, Joshua K., Eli P. Fenichel, and Seong D. Yun. (2026). Risky (Natural) Assets: Stochasticity, Nonconvexity, and the Value of Natural Capital. *Journal of the Association of Environmental and Resource Economists*, 13(5), 1269-1309. doi:10.1086/741689

Pindyck, Robert S. (1984). Uncertainty in the Theory of Renewable Resource Markets. *Review of Economic Studies*, 51(2), 289–303. doi:10.2307/2297693

See Also

`vaprox`, `vaprox.pindyck`, `vsim`

`aproxdef` *Define Approximation Space*

Description

Define an approximation space for the value function and price function approximation methods (`V`, `P`, and `Pdot`).

Usage

```
aproxdef(deg, lb, ub, delta)
```

Arguments

`deg`	A vector specifying the number of polynomials (degrees of Chebyshev polynomials) in each dimension.
`lb`	A vector of lower bounds for each dimension.
`ub`	A vector of upper bounds for each dimension.
`delta`	A numeric scalar specifying the discount rate.

Details

For the i-th dimension, $i = 1, 2, \ldots, d$, suppose a polynomial approximant over a bounded interval $[a_i, b_i]$ is defined using Chebyshev polynomials. Then, a d-dimensional approximation domain is defined as:

$$\mathbf{S} = \{(s_1, s_2, \ldots, s_d) \mid a_i \le s_i \le b_i,\ i = 1, 2, \ldots, d\}.$$

Suppose n_i Chebyshev polynomials (i.e., degree $n_i - 1$) are used for the i-th dimension. The approximation space is defined by:

`deg` = `c(`$n_1, n_2, \ldots, n_d$`)`,
`lb` = `c(`$a_1, a_2, \ldots, a_d$`)`, and
`ub` = `c(`$b_1, b_2, \ldots, b_d$`)`.

The argument `delta` is the constant discount rate used in the approximation.

Value

A list defining the approximation space, containing polynomial degrees, bounds, and discount rate.

References

Fenichel, Eli P. and Joshua K. Abbott. (2014). Natural Capital: From Metaphor to Measurement. *Journal of the Association of Environmental Economists*, 1(1/2), 1–27. doi:10.1086/676034

Examples

```
## Reef-fish example: see Fenichel and Abbott (2014)
delta <- 0.02          # discount rate
upper <- 359016000     # upper bound on approximation space
lower <- 5e+6          # lower bound on approximation space

myspace <- aproxdef(50, lower, upper, delta)

## Two-dimensional example
ub <- c(1.5, 1.5)
lb <- c(0.1, 0.1)
deg <- c(20, 20)
delta <- 0.03

myspace <- aproxdef(deg, lb, ub, delta)
```

`chebbasisgen` *Generate Unidimensional Chebyshev Polynomial Basis*

Description

Compute the monomial basis of Chebyshev polynomials for given unidimensional nodes s_i over a bounded interval $[a, b]$.

Usage

```
chebbasisgen(nodes, npol, a, b, dorder = NULL)
```

Arguments

`nodes`	A numeric vector of Chebyshev nodes s_i (an array of nodes in the `capn` package).
`npol`	An integer specifying the number of polynomials (n polynomials correspond to degree $n-1$).
`a`	The lower bound of the interval $[a, b]$.
`b`	The upper bound of the interval $[a, b]$.
`dorder`	Order of the partial derivative of the basis. The default `NULL` returns the basis itself. If `dorder = 1`, the first derivative is returned. Higher-order derivatives can be requested with `dorder >= 2`.

Details

Suppose there are m Chebyshev nodes over a bounded interval $[a, b]$:

$s_i \in [a, b]$, for $i = 1, 2, \ldots, m$.

These nodes are normalized to the standard Chebyshev domain $[-1, 1]$ as:

$z_i = \frac{2(s_i - a)}{b-a} - 1$.

With normalized Chebyshev nodes, the recurrence relations for Chebyshev polynomials are:

$T_0(z_i) = 1$,
$T_1(z_i) = z_i$, and
$T_n(z_i) = 2z_i T_{n-1}(z_i) - T_{n-2}(z_i)$.

The interpolation (Vandermonde) matrix of Chebyshev polynomials of degree up to $n - 1$ with m nodes, Φ_{mn}, is:

$$\Phi_{mn} = \begin{bmatrix} 1 & T_1(z_1) & \cdots & T_{n-1}(z_1) \\ 1 & T_1(z_2) & \cdots & T_{n-1}(z_2) \\ \vdots & \vdots & \ddots & \vdots \\ 1 & T_1(z_m) & \cdots & T_{n-1}(z_m) \end{bmatrix}.$$

Derivatives of the basis are computed using the identity:

$(1 - z_i^2) T_n'(z_i) = n \left[T_{n-1}(z_i) - z_i T_n(z_i) \right].$

Further technical details on Chebyshev polynomial bases can be found in Amparo et al. (2007) and Miranda and Fackler (2002).

Value

A numeric matrix of dimension $m \times n$, where m is the number of nodes and $n = npol$, containing the Chebyshev polynomial basis (Vandermonde matrix) or its derivatives.

References

Amparo, Gil, Javier Segura, and Nico Temme (2007). *Numerical Methods for Special Functions*. Cambridge: Cambridge University Press.

Fenichel, Eli P. and Joshua K. Abbott. (2014). Natural Capital: From Metaphor to Measurement. *Journal of the Association of Environmental Economists*, 1(1/2), 1–27. doi:10.1086/676034

Miranda, Mario J. and Paul L. Fackler (2002). *Applied Computational Economics and Finance*. Cambridge: MIT Press.

See Also

chebnodegen

Examples

```
## Reef-fish example: see Fenichel and Abbott (2014)
data("GOM")
param <- GOM$param
nodes <- chebnodegen(
  param$nodes,
  param$lowerK,
  param$upperK
)

## Chebyshev polynomial basis
chebbasisgen(nodes, 20, 0.1, 1.5)

## First derivative of the Chebyshev polynomial basis
chebbasisgen(nodes, 20, 0.1, 1.5, dorder = 1)
```

---

chebgrids *Generate Chebyshev Grids*

Description

Generate a multi-dimensional Chebyshev grid over bounded intervals.

Usage

```
chebgrids(nnodes, lb, ub, rtype = NULL)
```

Arguments

nnodes	A vector specifying the number of nodes in each dimension.
lb	A vector of lower bounds for each dimension.
ub	A vector of upper bounds for each dimension.
rtype	Type of return value. The default `NULL` returns a list. If `rtype = "list"`, a list is returned. If `rtype = "grid"`, a matrix is returned.

Details

For the i-th dimension, $i = 1, 2, \ldots, d$, suppose a polynomial approximant s_i over a bounded interval $[a_i, b_i]$ is defined using Chebyshev nodes. Then, a d-dimensional Chebyshev grid is defined as:

$$\mathbf{S} = \{(s_1, s_2, \ldots, s_d) \mid a_i \leq s_i \leq b_i,\ i = 1, 2, \ldots, d\}.$$

This corresponds to all combinations of the marginal Chebyshev grids s_i. Two types of return values are provided. `rtype = "list"` returns a list of length d containing the marginal grids, whereas `rtype = "grid"` returns a $\left(\prod_{i=1}^{d} n_i\right) \times d$ matrix of all grid points.

Value

A list with d elements of Chebyshev nodes, or a $\left(\prod_{i=1}^{d} n_i\right) \times d$ matrix of Chebyshev grid points.

See Also

chebnodegen

Examples

```
## Chebyshev grid with two dimensions
chebgrids(c(5, 3), c(1, 1), c(2, 3))

## Returns the same result explicitly as a list
chebgrids(c(5, 3), c(1, 1), c(2, 3), rtype = "list")

## Returns a matrix of grid points over the same domain
chebgrids(c(5, 3), c(1, 1), c(2, 3), rtype = "grid")

## Chebyshev grid with one dimension
chebgrids(5, 1, 2)
chebnodegen(5, 1, 2)

## Chebyshev grid with three dimensions
chebgrids(c(3, 4, 5), c(1, 1, 1), c(2, 3, 4), rtype = "grid")
```

---

chebnodegen *Unidimensional Chebyshev Nodes*

Description

Generate unidimensional Chebyshev nodes over a bounded interval.

Usage

```
chebnodegen(n, a, b)
```

Arguments

n	An integer specifying the number of nodes.
a	The lower bound of the interval $[a, b]$.
b	The upper bound of the interval $[a, b]$.

Details

A polynomial approximant s_i over a bounded interval $[a, b]$ is constructed using Chebyshev nodes as:

$$s_i = \frac{b+a}{2} + \frac{b-a}{2} \cos\left(\frac{n-i+0.5}{n}\pi\right) \text{ for } i = 1, 2, \ldots, n.$$

Further details can be found in Miranda and Fackler (2002, p.~119).

Value

A numeric vector of length n containing Chebyshev nodes.

References

Miranda, Mario J. and Paul L. Fackler (2002). *Applied Computational Economics and Finance*. Cambridge: MIT Press.

Examples

```
## 10 Chebyshev nodes in [-1, 1]
chebnodegen(10, -1, 1)

## 5 Chebyshev nodes in [1, 5]
chebnodegen(5, 1, 5)
```

`forest` *Douglas-Fir Forest Data in Western Oregon (1-D Example)*

Description

`forest` contains parameters and simulated data used to replicate the Douglas-fir forest in Western Oregon example (industrial siteclass 3) from Hashida and Fenichel (2022). The dataset illustrates the natural capital asset pricing (CAPN) framework in a one-dimensional deterministic setting. See `vignette("forestDemo")` for an example demonstrating the use of this data.

Usage

```
data("forest")
```

Format

A list with two elements:

param A data.frame of model and approximation parameters:

- `delta`: Discount rate (= 0.07)
- `order`: Chebyshev polynomial order (= 400)
- `lowerK`: lower bound of Chebyshev nodes (= 1)

- `upperK`: Upper bound of Chebyshev nodes (= 80)
- `nodes`: Number of Chebyshev nodes (= 400)
- `crit.vol`: Boundary (Moratorium or Maximum Standing) Volume (= 18.6592)
- `margp`: Marginal use value $\frac{d\pi}{ds}$ at `crit.vol` (= 743.439)
- `splittime`: Split time $t(s)$ at `crit.vol` (= 57.21486)

simData A data.frame of simulated values evaluated at approximation nodes:

- `vol`: Forest volume (MBF) at Chevyshev nodes in $s \in$ [1,80]
- `growth`: Evaluated stock dynamics $\frac{ds}{dt}$ at `vol`
- `profit`: Profit evaluated at `vol`
- `times`: Evaluated time with $t(s)$ at `vol`

Details

See Hashida and Fenichel (2022).

References

Hashida, Yukiko and Eli P. Fenichel. (2022). Valuing Natural Capital When Management Is Dominated by Periods of Inaction. *American Journal of Agricultural Economics*, 104(2), 791–811. doi:10.1111/ajae.12250

`GOM` *Gulf of Mexico Reef Fish Example Data (1-D Deterministic)*

Description

`GOM` contains parameters and simulated data used to replicate the Gulf of Mexico reef fish example from Fenichel and Abbott (2014). The dataset illustrates the natural capital asset pricing (CAPN) framework in a one-dimensional deterministic setting.

Usage

```
data("GOM")
```

Format

A list with two elements:

param A data.frame of model and approximation parameters:

- `r`: Intrinsic growth rate (= 0.3847)
- `k`: Carrying capacity (= 359016000)
- `q`: Catchability coefficient (= 0.0003172934415731126)
- `price`: Output price (= 2.70)
- `cost`: Unit cost (= 153.0)
- `alpha`: Technology parameter (= 0.5436459179063678)

- `gamma`: Pre-ITQ management parameter (= 0.7882)
- `y`: System equivalence parameter (= 0.15745573410462155)
- `delta`: Discount rate (= 0.02)
- `order`: Chebyshev polynomial order (= 50)
- `upperK`: Upper bound of Chebyshev nodes (= `k`)
- `lowerK`: Lower bound of Chebyshev nodes (= 5×10^6)
- `nodes`: Number of Chebyshev nodes (= 50)

simData A data.frame of simulated values evaluated at approximation nodes:

- `stock`: Approximation nodes
- `profit`: Profit evaluated at `stock`
- `sdot`: Evaluated stock dynamics $\frac{ds}{dt}$
- `dsdotds`: Evaluated $\frac{d}{ds}\left(\frac{ds}{dt}\right)$
- `dsdotdss`: Evaluated $\frac{d^2}{ds^2}\left(\frac{ds}{dt}\right)$
- `dwds`: Evaluated $\frac{dw}{ds}$
- `dwdss`: Evaluated $\frac{d^2w}{ds^2}$

Details

From Fenichel and Abboott(2014),

catch effort: $x(s) = ys^{\gamma}$,

harvest: $h(s,x) = q((ys^{\gamma})^{\alpha})s = q(y^{\alpha})(s^{\gamma\alpha})$,

profit: $w(s,x) = price \cdot h(s,x) - cost \cdot x(s)$, and

sdot: $\dot{s} = rs\left(1 - \frac{s}{k}\right) - q(y^{\alpha})(s^{\gamma\alpha+1})$.

The biological parameters are:

- $r = 0.3847$: Intrinsic growth rate
- $K = 359016000$: Carrying capacity

The economic parameters are:

- $q = 0.00031729344157311126$: Catchability coefficient
- $price = 2.70$: Output price
- $cost = 153.0$: Unit cost
- $alpha = 0.5436459179063678$: Technology parameter
- $gamma = 0.7882$: Pre-ITQ management parameter
- $y = 0.15745573410462155$: System equivalence parameter
- $delta = 0.02$: Discount rate

References

Fenichel, Eli P. and Joshua K. Abbott. (2014). Natural Capital: From Metaphor to Measurement. *Journal of the Association of Environmental Economists*, 1(1/2), 1–27. doi:10.1086/676034

LV *Prey–Predator (Lotka–Volterra) Example Data (2-D Deterministic)*

Description

`LV` contains simulated data for a prey–predator (Lotka–Volterra) bioeconomic model used to illustrate multi-dimensional natural capital asset pricing (CAPN) methods. This example corresponds to a two-dimensional deterministic setting. The dataset includes both approximation data on a Chebyshev grid and time-series simulation output from the underlying dynamic system. See `vignette("LVDemo")` for an example demonstrating the use of this data.

Usage

```
data("LV")
```

Format

A list with two elements:

lvaproxdata A data.frame of approximation data evaluated on a 20×20 Chebyshev grid:

- `xs`: Prey stock
- `ys`: Predator stock
- `xdot`: Evaluated prey dynamics $\frac{dx}{dt}$
- `ydot`: Evaluated predator dynamics $\frac{dy}{dt}$
- `wval`: Profit (objective value W in Fenichel and Abbott (2014))

lvsimdata A data.frame of time-series simulation output from solving the ODE system:

- `tseq`: Time sequence from 0 to 100
- `xs`: Prey stock
- `ys`: Predator stock

Details

The prey–predator system is given by:

Prey (X): $\dot{X} = rX\left(1 - \frac{X}{K}\right) - aXY - \theta X$

Predator (Y): $\dot{Y} = bXY - mY - \gamma Y$

The biological parameters are:

- $r = 0.025$: Intrinsic growth rate of prey
- $K = 1$: Carrying capacity of prey
- $a = 0.08$: Predation effect on prey
- $b = 0.05$: Prey-to-predator conversion parameter
- $m = 0.01$: Natural mortality rate of predator
- $\gamma = 0.005$: Predator harvest control slope

- $\theta = 0.005$: Prey harvest control slope

The economic objective is defined as: $W = \text{harv.prey}\,(p_{\text{prey}} - c_{\text{prey}}/X)\,\theta X + \text{harv.pred}\,(p_{\text{pred}} - c_{\text{pred}}/Y)\,\gamma Y$

The economic parameters are:

- $p_{\text{pred}} = 0$: Price per unit harvest of predator
- $p_{\text{prey}} = 25$: Price per unit harvest of prey
- $c_{\text{prey}} = 0.1\,p_{\text{prey}}$: Cost per unit of prey effort (Schaefer model, with $q = 1$)
- $c_{\text{pred}} = c_{\text{prey}}$: Cost per unit of predator effort (Schaefer model, with $q = 1$)

`paprox` *Calculate P-Approximation Coefficients*

Description

Computes the P-approximation coefficients for the shadow price function using Chebyshev polynomial approximation as defined by `aproxdef`. Currently, only the one-dimensional deterministic case is supported.

Usage

```
paprox(aproxspace, stock, sdot, dsdotds, dwds)
```

Arguments

`aproxspace`	An approximation space defined by the `aproxdef` function.
`stock`	A vector or array of stock values, s.
`sdot`	A vector or array of stock dynamics, $\dot{s} = \frac{ds}{dt}$.
`dsdotds`	A vector or array of derivatives of stock dynamics with respect to stock, $\frac{d\dot{s}}{ds}$.
`dwds`	A vector or array of marginal value of the flow payoff with respect to stock, $\frac{dW}{ds}$.

Details

The P-approximation solves for the shadow price of a stock, $p(s)$, from:

$p(s) = \frac{W_s(s) + \dot{p}(s)}{\delta - \dot{s}_s(s)}$,

where $W_s = \frac{dW}{ds}$, $\dot{p}(s) = \frac{dp}{ds}$, $\dot{s}_s = \frac{d\dot{s}}{ds}$, and δ is the discount rate.

The shadow price function is approximated as:
$p(s) = \mathbf{\Phi}(s)\boldsymbol{\beta}$,

where $\mathbf{\Phi}(s)$ is a vector of Chebyshev basis functions and $\boldsymbol{\beta}$ is the vector of unknown coefficients.

Using the chain rule and Chebyshev basis properties:
$\dot{p}(s) = \mathrm{diag}(\dot{s})\, \boldsymbol{\Phi}_s(s)\boldsymbol{\beta}$.

Substituting into the pricing equation yields:
$\boldsymbol{\beta} = \left(\mathrm{diag}(\delta - \dot{s}_s)\boldsymbol{\Phi} - \mathrm{diag}(\dot{s})\boldsymbol{\Phi}_s\right)^{-1} W_s$.

In the over-determined case (more nodes than approximation degrees), the function uses a least-squares solution:
$\boldsymbol{\beta} = \left(A^T A\right)^{-1} A^T W_s$,
where:
$A = \mathrm{diag}(\delta - \dot{s}_s)\boldsymbol{\mu} - \mathrm{diag}(\dot{s})\boldsymbol{\mu}_s$.

See Fenichel et al. (2016) for additional theoretical background.

Value

A list containing the P-approximation results. Components can be accessed using `results$item` or `results[["item"]]`. The list includes:

`degree`	Degree of the Chebyshev polynomial approximation.
`lowerB`	Lower bound of the approximation domain.
`upperB`	Upper bound of the approximation domain.
`delta`	Discount rate.
`coefficient`	Estimated Chebyshev polynomial coefficients for the shadow price function.

References

Fenichel, Eli P. and Joshua K. Abbott. (2014). Natural Capital: From Metaphor to Measurement. *Journal of the Association of Environmental Economists*, 1(1/2), 1–27. doi:10.1086/676034

Fenichel, Eli P., Joshua K. Abbott, Jude Bayham, Whitney Boone, Erin M. K. Haacker, and Lisa Pfeiffer. (2016) Measuring the Value of Groundwater and Other Forms of Natural Capital. *Proceedings of the National Academy of Sciences*. 113: 2382–2387. doi:10.1073/pnas.1513779113

See Also

`aproxdef`, `psim`, `GOM`

Examples

```
## 1-D Reef-fish example: see Fenichel and Abbott (2014)
data("GOM")

param <- GOM$param
simData <- GOM$simData

Aspace <- aproxdef(param$order, param$lowerK, param$upperK, param$delta)

pC <- paprox(
```

```
  Aspace,
  simData$stock,
  simData$sdot,
  simData$dsdotds,
  simData$dwds
)
```

---

pdotaprox *Compute Pdot-Approximation Coefficients*

Description

Computes Pdot-approximation coefficients for the Chebyshev polynomial representation of the shadow price derivative using the approximation space defined by `aproxdef`. Currently, only the one-dimensional case is supported.

Usage

```
pdotaprox(aproxspace, stock, sdot, dsdotds, dsdotdss, dwds, dwdss)
```

Arguments

`aproxspace`	An approximation space defined by the `aproxdef` function.
`stock`	A vector of stock values, s.
`sdot`	A vector of stock growth rates, $\dot{s} = \frac{ds}{dt}$.
`dsdotds`	A vector of derivatives of stock growth with respect to stock, $\frac{d\dot{s}}{ds}$.
`dsdotdss`	A vector of second derivatives of stock growth with respect to stock, $\frac{d}{ds}\left(\frac{d\dot{s}}{ds}\right)$.
`dwds`	A vector of marginal welfare with respect to stock, $\frac{dW}{ds}$.
`dwdss`	A vector of second derivatives of marginal welfare with respect to stock, $\frac{d}{ds}\left(\frac{dW}{ds}\right)$.

Details

The Pdot-approximation solves for the shadow price of a stock, $p(s)$, using the relationship:

$p(s) = \frac{W_s(s)+\dot{p}(s)}{\delta-\dot{s}_s}$,

where $W_s = \frac{dW}{ds}$, $\dot{p}(s) = \frac{dp}{dt}$, $\dot{s}_s = \frac{d\dot{s}}{ds}$, and δ is the discount rate.

Taking the time derivative of this expression yields:

$\dot{p} = \frac{(W_{ss}\dot{s}+\ddot{p})(\delta-\dot{s}_s)+(W_s+\dot{p})(\dot{s}_{ss}\dot{s})}{(\delta-\dot{s}_s)^2}$.

Let the approximation be $\dot{p}(s) = \mathbf{\Phi}(s)\beta$, where $\mathbf{\Phi}(s)$ is a vector of Chebyshev basis functions and β is the coefficient vector. Then,

$\ddot{p} = \frac{d\dot{p}}{ds}\frac{ds}{dt} = \text{diag}(\dot{s})\mathbf{\Phi}_s(s)\beta.$

Substituting and rearranging yields the linear system:

$\beta = A^{-1}B,$
where

$A = \text{diag}\left(\delta - \dot{s}_s\right)^2 \mu - \text{diag}\left(\dot{s}(\delta - \dot{s}_s)\right)\mu_s - \text{diag}\left(\dot{s}_{ss}\dot{s}\right)\mu,$

and

$B = W_{ss}\dot{s}(\delta - \dot{s}_s) + W_s\dot{s}_{ss}\dot{s}.$

In the over-determined case, coefficients are obtained via least squares:

$\beta = \left(A^T A\right)^{-1} A^T B.$

For additional theoretical background, see Fenichel and Abbott (2014).

Value

A list containing the approximation results. Individual elements can be accessed using `results$item` or `results[["item"]]`. The list includes:

`degree`	Degree of the Chebyshev polynomial.
`lowerB`	Lower bound of the Chebyshev domain.
`upperB`	Upper bound of the Chebyshev domain.
`delta`	Discount rate.
`coefficient`	Estimated Chebyshev polynomial coefficients.

References

Fenichel, Eli P. and Joshua K. Abbott. (2014). Natural Capital: From Metaphor to Measurement. *Journal of the Association of Environmental Economists*, 1(1/2), 1–27. doi:10.1086/676034

See Also

`aproxdef`, `pdotsim`

Examples

```
## 1-D Reef-fish example: see Fenichel and Abbott (2014)
data("GOM")

param <- GOM$param
simData <- GOM$simData

Aspace <- aproxdef(param$order, param$lowerK, param$upperK, param$delta)
```

```
pdotC <- pdotaprox(
  Aspace,
  simData$stock,
  simData$sdot,
  simData$dsdotds,
  simData$dsdotdss,
  simData$dwds,
  simData$dwdss
)
```

---

pdotsim *Simulation of Pdot-Approximation*

Description

Simulates shadow prices and value functions using coefficients from the Pdot-approximation obtained via pdotaprox.

Usage

```
pdotsim(pdotcoeff, stock, sdot, dsdotds, wval, dwds)
```

Arguments

pdotcoeff	An approximation result from the pdotaprox function.
stock	A numeric vector of stock values, s.
sdot	A numeric vector of stock growth rates, $\dot{s} = \frac{ds}{dt}$.
dsdotds	A numeric vector of derivatives of stock growth with respect to stock, $\frac{d\dot{s}}{ds}$.
wval	A numeric vector of W-values.
dwds	A numeric vector of marginal W-values, $\frac{dW}{ds}$.

Details

Let $\hat{\beta}$ denote the vector of approximation coefficients from pdotaprox. Over the approximation interval $s \in [a, b]$, the estimated shadow (accounting) price is given by:

$\hat{p}(s) = \frac{W_s(s)+\mu(s)\hat{\beta}}{\delta-\dot{s}_s(s)}$.

The corresponding estimated value function is:

$\hat{V}(s) = \frac{1}{\delta}\left(W(s) + \hat{p}(s)\,\dot{s}(s)\right)$.

These expressions follow directly from the Pdot-approximation framework developed in Fenichel and Abbott (2014) and Fenichel et al. (2016).

Value

A list of approximation results. Use `results$item` (or `results[["item"]]`) to extract individual components:

`shadowp`	Estimated shadow (accounting) prices, $\hat{p}(s)$.
`iw`	Inclusive wealth, $\hat{p}(s)\,s$.
`vfun`	Estimated value function, $\hat{V}(s)$.
`stock`	Stock values used in the simulation.
`wval`	W-values used in the simulation.

References

Fenichel, Eli P. and Joshua K. Abbott. (2014). Natural Capital: From Metaphor to Measurement. *Journal of the Association of Environmental Economists*, 1(1/2), 1–27. doi:10.1086/676034

Fenichel, Eli P., Joshua K. Abbott, Jude Bayham, Whitney Boone, Erin M. K. Haacker, and Lisa Pfeiffer. (2016) Measuring the Value of Groundwater and Other Forms of Natural Capital. *Proceedings of the National Academy of Sciences*. 113: 2382–2387. doi:10.1073/pnas.1513779113

See Also

`aproxdef`, `pdotaprox`, `plotgen`, `GOM`

Examples

```
## 1-D Reef-fish example: see Fenichel and Abbott (2014)
data("GOM")

param <- GOM$param
simData <- GOM$simData

Aspace <- aproxdef(param$order, param$lowerK, param$upperK, param$delta)

pdotC <- pdotaprox(
  Aspace,
  simData$stock,
  simData$sdot,
  simData$dsdotds,
  simData$dsdotdss,
  simData$dwds,
  simData$dwdss
)

GOMSimPdot <- pdotsim(
  pdotC,
  simData$stock,
  simData$sdot,
  simData$dsdotds,
  simData$profit,
```

```
  simData$dwds
)

## Shadow price
plotgen(GOMSimPdot,
        xlabel = "Stock size, s",
        ylabel = "Shadow price")

## Value function and profit
plotgen(GOMSimPdot,
        ftype  = "vw",
        xlabel = "Stock size, s",
        ylabel = c("Value Function", "Profit"))
```

---

plotgen *Plot Generator for Shadow Price and Value Function*

Description

Generate plots of shadow prices and value functions based on simulation results from `vsim`, `psim`, or `pdotsim`.

Usage

```
plotgen(simres, ftype = NULL, whichs = NULL, tvar = NULL,
        xlabel = NULL, ylabel = NULL)
```

Arguments

`simres`	A simulation result from `vsim`, `psim`, or `pdotsim`.
`ftype`	Plot type. If `NULL` (default) or `"p"`, plot shadow price. If `"vw"`, plot value function (and W-value if available).
`whichs`	An integer specifying which stock to plot in multi-stock cases. Must satisfy `1 <= whichs <= number of stocks`. If `NULL` (default), the first stock is used.
`tvar`	Optional time variable. If provided, the x-axis is time instead of stock.
`xlabel`	Optional character string for the x-axis label. Defaults to `"Stock"` or `"Time"` depending on `tvar`.
`ylabel`	Optional character vector for y-axis labels. For `ftype = "p"`, provide one label. For `ftype = "vw"`, provide one or two labels for the value function and W-value, respectively.

Details

This function produces one-dimensional plots for:

- Shadow price versus stock
- Shadow price versus time

- Value function versus stock
- Value function versus time
- Value function and W-value versus stock
- Value function and W-value versus time

The specific plot depends on the combination of `ftype`, `whichs`, and `tvar`.

Value

Produces a plot of shadow prices, value function, and optionally W-values. The function is called for its side effect of producing a plot.

See Also

GOM pdotsim psim, vsim, vaprox.pindyck

Examples

```
## 1-D Reef-fish example: see Fenichel and Abbott (2014)
data("GOM")

param <- GOM$param
simData <- GOM$simData

Aspace <- aproxdef(param$order, param$lowerK, param$upperK, param$delta)

pC <- paprox(
  Aspace,
  simData$stock,
  simData$sdot,
  simData$dsdotds,
  simData$dwds
)

## Without providing W-value
GOMSimP <- psim(
  pC,
  simData$stock
)

## With W-value
GOMSimP2 <- psim(pC,
  simData$stock,
  simData$profit,
  simData$sdot)

## Shadow price vs stock
plotgen(GOMSimP)
plotgen(GOMSimP, ftype = "p")
plotgen(GOMSimP,
        xlabel = "Stock Size, S",
```

```
        ylabel = "Shadow Price (USD/kg)")

## Value function and W-value vs stock
plotgen(GOMSimP2, ftype = "vw")
plotgen(GOMSimP2,
        ftype  = "vw",
        xlabel = "Stock Size, S",
        ylabel = c("Value Function", "Profit"))
```

---

psim *Simulation of P-Approximation*

Description

Simulates the P-approximation by evaluating the estimated shadow price function and, optionally, the associated value function over a given set of stock values.

Usage

```
psim(pcoeff, stock, wval = NULL, sdot = NULL)
```

Arguments

pcoeff	An approximation result object returned by paprox.
stock	A numeric vector of stock values at which to evaluate the P-approximation.
wval	(Optional) A numeric vector of flow payoff values, $W(s)$. Must be provided together with sdot in order to compute the value function.
sdot	(Optional) A numeric vector of stock dynamics, $\dot{s} = \frac{ds}{dt}$. Must be provided together with wval in order to compute the value function.

Details

Let $\hat{\beta}$ denote the vector of approximation coefficients returned by paprox. The estimated shadow (accounting) price over the approximation interval $s \in [a, b]$ is given by:

$\hat{p}(s) = \mathbf{\Phi}(s)\,\hat{\boldsymbol{\beta}}$,

where $\mathbf{\Phi}(s)$ denotes the vector of Chebyshev basis functions.

If both wval and sdot are provided, the estimated value function is computed as:

$\hat{V}(s) = \frac{1}{\delta}\left(W(s) + \hat{p}(s)\,\dot{s}(s)\right)$.

For additional theoretical details, see Fenichel and Abbott (2014) and Fenichel et al. (2016).

Value

A list containing simulated P-approximation outputs. Use `result$item` (or `result[["item"]]`) to extract individual components.

`shadowp`	Estimated shadow (accounting) prices, $\hat{p}(s)$.
`iw`	Inclusive wealth, defined as $\hat{p}(s) \cdot s$.
`vfun`	Estimated value function, $\hat{V}(s)$. Returned only if both `wval` and `sdot` are provided; otherwise `NULL`.
`stock`	Stock values used for evaluation, returned as a column matrix.
`wval`	Flow payoff values, $W(s)$, if provided; otherwise `NULL`.

References

Fenichel, Eli P. and Joshua K. Abbott. (2014). Natural Capital: From Metaphor to Measurement. *Journal of the Association of Environmental Economists*, 1(1/2), 1–27. doi:10.1086/676034

Fenichel, Eli P., Joshua K. Abbott, Jude Bayham, Whitney Boone, Erin M. K. Haacker, and Lisa Pfeiffer. (2016) Measuring the Value of Groundwater and Other Forms of Natural Capital. *Proceedings of the National Academy of Sciences*. 113: 2382–2387. doi:10.1073/pnas.1513779113

See Also

`aproxdef`, `paprox`, `plotgen`, `GOM`

Examples

```
## 1-D Reef-fish example: see Fenichel and Abbott (2014)
data("GOM")

param <- GOM$param
simData <- GOM$simData

Aspace <- aproxdef(param$order, param$lowerK, param$upperK, param$delta)

pC <- paprox(
  Aspace,
  simData$stock,
  simData$sdot,
  simData$dsdotds,
  simData$dwds
)

GOMSimP <- psim(
  pC,
  simData$stock,
  simData$profit,
  simData$sdot
)
```

```
## Shadow price
plotgen(GOMSimP, xlabel = "Stock size, s", ylabel = "Shadow price")

## Value function and profit
plotgen(
  GOMSimP,
  ftype  = "vw",
  xlabel = "Stock size, s",
  ylabel = c("Value Function", "Profit")
)
```

---

unigrids *Generate Uniform Grids*

Description

Generate a multi-dimensional uniform grid over bounded intervals.

Usage

```
unigrids(nnodes, lb, ub, rtype = NULL)
```

Arguments

`nnodes`	A vector specifying the number of nodes in each dimension.
`lb`	A vector of lower bounds for each dimension.
`ub`	A vector of upper bounds for each dimension.
`rtype`	Type of return value. The default `NULL` returns a list. If `rtype = "list"`, a list is returned. If `rtype = "grid"`, a matrix is returned.

Details

For the i-th dimension, $i = 1, 2, \ldots, d$, suppose a polynomial approximant s_i over a bounded interval $[a_i, b_i]$ is defined by evenly spaced nodes. Then, a d-dimensional uniform grid is defined as:

$$\mathbf{S} = \{(s_1, s_2, \ldots, s_d) \mid a_i \le s_i \le b_i,\ i = 1, 2, \ldots, d\}\,.$$

This corresponds to all combinations of the marginal grids s_i. Two types of return values are provided. `rtype = "list"` returns a list of length d containing the marginal grids, whereas `rtype = "grid"` returns a $\left(\prod_{i=1}^{d} n_i\right) \times d$ matrix of all grid points.

Value

A list with d elements of uniform nodes, or a $\left(\prod_{i=1}^{d} n_i\right) \times d$ matrix of uniform grid points.

Examples

```
## Uniform grid with two dimensions
unigrids(c(5, 3), c(1, 1), c(2, 3))

## Returns the same result explicitly as a list
unigrids(c(5, 3), c(1, 1), c(2, 3), rtype = "list")

## Returns a matrix of grid points over the same domain
unigrids(c(5, 3), c(1, 1), c(2, 3), rtype = "grid")

## Uniform grid with one dimension
unigrids(5, 1, 2)

## Uniform grid with three dimensions
unigrids(c(3, 4, 5), c(1, 1, 1), c(2, 3, 4), rtype = "grid")
```

---

vaprox *Computing Value Function (V) Approximation Coefficients*

Description

Computes Chebyshev polynomial coefficients for approximating the value function $V(s)$ over a bounded state space. The function supports multi-dimensional deterministic and stochastic dynamic optimization problems within the natural capital asset pricing (CAPN) framework.

Usage

```
vaprox(aproxspace, stock, sdot, w, covmat = NULL)
```

Arguments

`aproxspace`	An approximation space defined by the `aproxdef` function.
`stock`	A vector or matrix of stock states s. For multi-dimensional problems, each column corresponds to one state variable.
`sdot`	A vector or matrix of stock dynamics $\dot{s} = ds/dt$ (deterministic case) or drift terms $\mu(s)$ (stochastic case), with the same dimension as `stock`.
`w`	A vector of net benefits (profits or economic program values), $W(s)$.
`covmat`	An optional variance–covariance matrix of the state dynamics. Columns must correspond to the unique (i, j) combinations of state variables. Set to `NULL` for deterministic problems.

Details

The V-approximation solves for the value function $V(s)$ from the Hamilton–Jacobi–Bellman equation.

Deterministic case:

$$\delta V(s) = W(s) + p(s)\dot{s}$$

Stochastic case:

$$\delta V(s) = W(s) + p(s)\mu(s) + \frac{1}{2}\sigma^2(s)p_s(s)$$

where δ is the discount rate, $W(s)$ denotes net benefits, $p(s) = \partial V(s)/\partial s$ is the shadow (accounting) price, $\mu(s)$ is the drift term, and $\sigma^2(s)$ is the variance of the stochastic process.

The value function is approximated as

$$V(s) = \Phi(s)\boldsymbol{\beta},$$

where $\Phi(s)$ denotes the Chebyshev basis evaluated at s and $\boldsymbol{\beta}$ is the vector of unknown coefficients.

Substituting this approximation into the HJB equation yields a linear system:

Deterministic:

$$\delta\Phi(s)\boldsymbol{\beta} = W(s) + \mathrm{diag}(\dot{s})\Phi_s(s)\boldsymbol{\beta}$$

Stochastic:

$$\delta\Phi(s)\boldsymbol{\beta} = W(s) + \mathrm{diag}(\mu(s))\Phi_s(s)\boldsymbol{\beta} + \frac{1}{2}\sigma^2(s)\Phi_{ss}(s)\boldsymbol{\beta}$$

In exactly identified cases, the system is solved directly. In over-determined cases (more nodes than basis terms), the coefficients are obtained via least squares:

$$\boldsymbol{\beta} = (A^\top A)^{-1}A^\top W,$$

where A collects the terms multiplying $\boldsymbol{\beta}$.

The formulation extends naturally to multi-dimensional state spaces using tensor products of Chebyshev bases.

Value

A list containing the approximation results:

- `degree`: Degree of the Chebyshev polynomial in each dimension.
- `lowerB`: Lower bounds of the approximation domain.
- `upperB`: Upper bounds of the approximation domain.
- `delta`: Discount rate.
- `coefficient`: Estimated Chebyshev polynomial coefficients.
- `model.type`: Either `"deterministic"` or `"stochastic"`.

References

Abbott, Joshua K., Eli P. Fenichel, and Seong D. Yun. (2026). Risky (Natural) Assets: Stochasticity, Nonconvexity, and the Value of Natural Capital. *Journal of the Association of Environmental and Resource Economists*, 13(5), 1269-1309. doi:10.1086/741689

Fenichel, Eli P. and Joshua K. Abbott. (2014). Natural Capital: From Metaphor to Measurement. *Journal of the Association of Environmental Economists*, 1(1/2), 1–27. doi:10.1086/676034

Yun, Seong D., Barbara Hutniczak, Joshua K. Abbott, and Eli P. Fenichel. (2017). Ecosystem-Based Management and the Wealth of Ecosystems. *Proceedings of the National Academy of Sciences*, 114(25), 6539–6544. doi:10.1073/pnas.1617666114

See Also

aproxdef, vsim, GOM, LV, AFY

Examples

```
## 1-D Deterministic: Reef-fish example (Fenichel and Abbott, 2014)
data("GOM")

param <- GOM$param
simData <- GOM$simData

Aspace <- aproxdef(param$order, param$lowerK, param$upperK, param$delta)

vC <- vaprox(
  Aspace,
  simData$stock,
  simData$sdot,
  simData$profit
)

## 2-D Deterministic: Prey--Predator example
data("LV")

lvspace <- aproxdef(
  deg = c(20, 20),
  lb = c(0.1, 0.1),
  ub = c(1.5, 1.5),
  delta  = 0.03
)

vCLV <- vaprox(
  lvspace,
  LV$lvaproxdata[, c("xs", "ys")],
  LV$lvaproxdata[, c("xdot", "ydot")],
  LV$lvaproxdata[, "wval"]
)

## 1-D Stochastic: Pindyck (1984) extension (Abbott et al., 2026)
data("AFY")

Aspace <- aproxdef(35, 0.2, 1.4, 0.05)

## deterministic
vCd <- vaprox(
  Aspace,
  AFY$simData$stock,
  AFY$simData$mus.d,
  AFY$simData$profit.d
)

## stochastic
vCs <- vaprox(
```

```
  Aspace,
  AFY$simData$stock,
  AFY$simData$mus.s,
  AFY$simData$profit.s,
  AFY$simData$sigs
)
```

---

vaprox.pindyck *Computing Value Function (V) Approximation Coefficients for examples in Pindyck (1984)*

Description

Computes Chebyshev polynomial coefficients for approximating the value function $V(s)$ over a bounded state space using value function iteration described in Pindyck (1984) and Abbott, Fenicehl, and Yun (2026). The function supports one-dimensional deterministic and stochastic dynamic optimization problems within the natural capital asset pricing (CAPN) framework.

Usage

```
vaprox.pindyck(param,
               growthfun = c("logistic", "gompertz", "sqroot"),
               covmat    = NULL,
               itermax   = 300,
               tol       = 1e-8)
```

Arguments

`param`	A data.frame of parameters adopted in three examples in pindyck (1984)
`growthfun`	A net growth function: one of the three net growth function in Pindyck (1984) `"logistic"` (Logistic: example 1), `"gompertz"` (Gompertz: example 2), and `"sqroot"` (Sqaure-root: example 3).
`covmat`	An optional variance–covariance matrix of the state dynamics. Columns must correspond to the unique (i, j) combinations of state variables. Set to `NULL` for deterministic problems.
`itermax`	Maximum number of iterations. Default is 300.
`tol`	Tolerance of the error size. Default is 1e-8.

Details

The V-approximation with value function iteration solves for the value function $V(s)$ in Pindyck (1984) and Abbott, Fenichel, and Yun (2026).

A state equation in the examples in Pindyck (1984) is defined as:

$$ds = [f(s) - q(t)]dt + \theta s dz$$

where dz is a Weiner process.

The net growth functions are:

Logistic net growth function:

$$f(s) = rs\left(1 - \frac{s}{K}\right)$$

Gompertz net gorwth function:

$$f(s) = rsln\left(\frac{K}{s}\right)$$

Square-root net gorwth function:

$$f(s) = rs^{1/2} - \frac{rs}{K^{1/2}}$$

where r is the instrinsic growth rate, K the carrying capacity $p(s) = \partial V(s)/\partial s$ is the shadow (accounting) price.

The catch (harvest) function:

$$q(s, V_s) = b\left(V_s + cs^{-\gamma}\right)^{-\eta}$$

The profit (net benefit) function:

If $\eta = 1$: $bln(q) - cs^{-\gamma}q$

if $\eta \neq 1$: $\frac{b^{1/\eta}}{(1-1/\eta)}q^{(1-1/\eta)} - cs^{-\gamma}q$

where b is the demand parameter, c is the cost parameter, η is the elasticity of demand, and γ is the elasticity of marginal cost.

More details are available in Pindyck (1984).

Value

A list containing the approximation results:

- `degree`: Degree of the Chebyshev polynomial in each dimension.
- `lowerB`: Lower bounds of the approximation domain.
- `upperB`: Upper bounds of the approximation domain.
- `delta`: Discount rate.
- `coefficient`: Estimated Chebyshev polynomial coefficients.
- `model.type`: Either `"deterministic"` or `"stochastic"`.
- `gfun`: net growth function decleared in the argument
- `wfun`: profit (net benefit) function with the decleared `"gfun"`
- `qfun`: catch (harvest) function with the decleared `"gfun"`

References

Abbott, Joshua K., Eli P. Fenichel, and Seong D. Yun. (2026). Risky (Natural) Assets: Stochasticity, Nonconvexity, and the Value of Natural Capital. *Journal of the Association of Environmental and Resource Economists*, 13(5), 1269-1309. doi:10.1086/741689

Pindyck, Robert S. (1984). Uncertainty in the Theory of Renewable Resource Markets. *Review of Economic Studies*, 51(2), 289–303. doi:10.2307/2297693

See Also

AFY aproxdef, plotgen vaprox, vsim

Examples

```
####################################################################################
## Logistic net growth function: Example 1 in Pindyck (1984)
data("AFY")

## parameters used in Abbott, Fenichel, and Yun (2026)
param <- AFY$param

## approximation and simulation range
stock <- chebnodegen(param$nodes,param$lowerK,param$upperK)

## volatility term
theta <- 0.1
sigsGBM <- as.matrix((theta*stock)^2,col=1)

cvlogistic <- vaprox.pindyck(param,'logistic',sigsGBM)
vlogistic <- vsim(cvlogistic,stock)

## plot value function
plotgen(vlogistic,ftype="vw",xlabel="Stock Size", ylabel="Value Function")

## plot shadow (accounting) prices
plotgen(vlogistic,ftype="p",xlabel="Stock Size", ylabel="Shadow Price")
```

---

vaprox.pjump *Value Function Approximation with Poisson Jump Risk*

Description

Computes Chebyshev polynomial coefficients for approximating the value function $V(s)$ in a dynamic optimization problem with Poisson jump risk. The function is designed for one-dimensional natural capital asset pricing applications.

Usage

```
vaprox.pjump(aproxspace, stock, sdot, wb, hs, zs)
```

Arguments

`aproxspace`	An approximation space defined by the aproxdef function.
`stock`	A vector or matrix of stock states s.
`sdot`	A vector or matrix of stock dynamics $\dot{s} = ds/dt$.
`wb`	A vector of net benefits (profits or economic program values) without a shock, $W(s)$.

`hs`	A vector of stock-dependent hazard rates, $h(s)$.
`zs`	A vector of infinite-horizon net benefits following a jump event, $Z(s)$.

Details

The value function $V(s)$ with Poisson jump risk satisfies the Hamilton–Jacobi–Bellman equation derived in Abbott, Fenichel, and Yun (2026), building on Reed and Heras (1992):

$$[\delta + h(s)]\, V(s) - h(s)Z(s) = W(s) + V_s(s)\, \dot{s},$$

where δ is the discount rate, $W(s)$ denotes net benefits in the absence of a jump, $V_s(s) = \partial V(s)/\partial s = p(s)$ is the shadow (accounting) price, $h(s)$ is the stock-dependent hazard rate, and $Z(s)$ is the post-jump infinite-horizon value.

Rearranging terms yields:

$$[\delta + h(s)]\, V(s) - V_s(s)\, \dot{s} = W(s) + h(s)Z(s).$$

The value function is approximated as:

$$V(s) = \Phi(s)\boldsymbol{\beta},$$

where $\Phi(s)$ is the Chebyshev basis evaluated at s and $\boldsymbol{\beta}$ is the vector of unknown coefficients.

Substituting this approximation into the HJB equation produces the linear system:

$$[(\delta + h(s))\Phi(s) - \mathrm{diag}(\dot{s})\Phi_s(s)]\, \boldsymbol{\beta} = W(s) + h(s)Z(s).$$

In exactly identified cases, the system is solved directly. In over-determined cases (more nodes than basis terms), the coefficients are obtained via least squares:

$$\boldsymbol{\beta} = (A^\top A)^{-1} A^\top (W + hZ),$$

where A collects the terms multiplying $\boldsymbol{\beta}$.

Value

A list containing the approximation results:

- `degree`: Degree of the Chebyshev polynomial in each dimension.
- `lowerB`: Lower bounds of the approximation domain.
- `upperB`: Upper bounds of the approximation domain.
- `delta`: Discount rate.
- `coefficient`: Estimated Chebyshev polynomial coefficients.
- `model.type`: `"Poisson Jump"`.

References

Abbott, Joshua K., Eli P. Fenichel, and Seong D. Yun. (2026). Risky (Natural) Assets: Stochasticity, Nonconvexity, and the Value of Natural Capital. *Journal of the Association of Environmental and Resource Economists*, 13(5), 1269-1309. doi:10.1086/741689

Reed, William J. and Hector E. Heras. (1992). The Conservation and Exploitation of Vulnerable Resources. *Bulletin of Mathematical Biology*, 54, 185–207. doi:10.1007/BF02464829

See Also

AFY, aproxdef, vaprox, vsim

Examples

```
## Example from Abbott, Fenichel, and Yun (2026)
data("AFY")

param <- AFY$param

## Deterministic benchmark
cvDET <- vaprox.pindyck(param, "logistic")

slow <- 0.1
vlow <- vsim(cvDET, slow)
qlow <- cvDET$qfun(param, slow, vlow$shadowp)
zs   <- cvDET$wfun(param, slow, qlow) / param$delta

s <- chebnodegen(param$nodes, param$lowerK, param$upperK)

qopt <- cvDET$qfun(param, s, vsim(cvDET, s)$shadowp)
mus  <- cvDET$gfun(param, s) - qopt

alpha <- 0.5
hs <- alpha / (alpha + s)

wb <- cvDET$wfun(param, s, qopt)

Aspace <- aproxdef(param$order, param$lowerK, param$upperK, param$delta)

cvPSS <- vaprox.pjump(Aspace, s, mus, wb, hs, zs)

vPSS <- vsim(cvPSS, s)

## Plot value function
plotgen(vPSS, ftype = "vw",
        xlabel = "Stock Size",
        ylabel = "Value Function")

## Plot shadow (accounting) price
plotgen(vPSS, ftype = "p",
        xlabel = "Stock Size",
```

```
ylabel = "Shadow Price")
```

vaprox.split *Value Function Approximation with Inaction (Split) Management*

Description

Computes Chebyshev polynomial coefficients for approximating the value function $V(s)$ in a one-dimensional natural capital asset pricing model with inaction (split) management.

The function allows for a boundary stock level at which management switches from waiting (inaction) to active use, and estimates both the value function and the associated passive value parameter.

Usage

```
vaprox.split(aproxspace, stock, sdot, profit,
             crit.stock, split.margp, split.time = NULL)
```

Arguments

aproxspace	An approximation space defined by the aproxdef function.
stock	A vector or matrix of stock states s.
sdot	A vector or matrix of stock dynamics $\dot{s} = ds/dt$ evaluated at stock (without split management).
profit	A vector of net benefits (profit or economic program value) evaluated at stock (without split management).
crit.stock	Critical (boundary) stock level, $\bar{s}$, representing a moratorium or maximum waiting threshold.
split.margp	Marginal use value at the boundary stock, $\frac{d\pi}{ds}\Big\|_{s=\bar{s}}$.
split.time	(Optional) Time of action $t(\bar{s})$ at the boundary stock. If NULL, the standard boundary condition is used.

Details

Following Hashida and Fenichel (2022), management is characterized by an indicator function:

$$x(s) = \begin{cases} 1 & \text{if } s \geq \bar{s}, \\ 0 & \text{if } s < \bar{s}. \end{cases}$$

The benefit function can be decomposed as:

$$W(s, x(s)) = \pi(s, x(s)) + \alpha(s, x(s)),$$

where $\pi(s, x(s))$ represents realized income when harvesting occurs ($x(s) = 1$), and $\alpha(s, x(s))$ represents passive (amenity) value when management remains inactive ($x(s) = 0$).

At the boundary $s = \bar{s}$, smooth pasting implies:

$$V_s(\bar{s}) = \pi_s(\bar{s}).$$

The Hamilton–Jacobi–Bellman equation becomes:

$$\delta V(s) = \pi(s) + \alpha(s) + \dot{s}\, V_s(s).$$

The value function is approximated by Chebyshev polynomials:

$$V(s) = \Phi(s)\boldsymbol{\beta},$$

where $\Phi(s)$ is the Chebyshev basis evaluated at s, and $\boldsymbol{\beta}$ is the vector of unknown coefficients.

Substituting into the HJB equation yields the linear system:

$$\left[\delta\Phi(s) - \mathrm{diag}(\dot{s})\Phi_s(s)\right]\boldsymbol{\beta} = \pi(s) + \alpha(s).$$

The passive value is approximated by a first-order form $\alpha(s) = \alpha s$, and $\boldsymbol{\beta}$ and α are jointly determined using the boundary condition $V_s(\bar{s}) = \pi_s(\bar{s})$.

In exactly identified cases, the system is solved directly. In over-determined cases, a least-squares solution is used:

$$\boldsymbol{\beta} = (A^\top A)^{-1} A^\top \left(\pi(s) + \alpha(s)\right),$$

where $A = \delta\Phi(s) - \mathrm{diag}(\dot{s})\Phi_s(s)$.

A complete application is provided in the `forestDemo` vignette. See `vignette("forestDemo")`.

Value

A list containing the approximation results:

- `degree`: Degree of the Chebyshev polynomial.
- `lowerB`: Lower bound of the approximation interval.
- `upperB`: Upper bound of the approximation interval.
- `delta`: Discount rate.
- `coefficient`: Estimated Chebyshev polynomial coefficients.
- `shadow.stop`: Shadow price evaluated at the boundary stock $\bar{s}$.
- `alpha`: Estimated passive value parameter at $\bar{s}$.
- `model.type`: `"Split (Economic Program)"`.

References

Hashida, Yukiko and Eli P. Fenichel. (2022). Valuing Natural Capital When Management Is Dominated by Periods of Inaction. *American Journal of Agricultural Economics*, 104(2), 791–811. doi:10.1111/ajae.12250

See Also

`aproxdef`, `forest`, `vsim`

`vsim` *Simulation of Value Function Approximation*

Description

Simulates the value function and shadow (accounting) prices using the Chebyshev polynomial approximation obtained from `vaprox`, `vaprox.pindyck`, and `vaprox.pjump`.

Usage

```
vsim(vcoeff, stock, wval = NULL)
```

Arguments

`vcoeff`	An approximation object returned by the `vaprox` function.
`stock`	A numeric vector, matrix, or `data.frame` containing stock values $\mathbf{S}$ at which the value function and shadow prices are evaluated.
`wval`	(Optional; used by `plotgen`) A vector or array of instantaneous welfare values $W(\mathbf{S})$.

Details

Let $\hat{\boldsymbol{\beta}}$ denote the vector of approximation coefficients obtained from `vaprox`. For stock vector s, the estimated shadow (accounting) price of stock is given by

$$\hat{p}(s) = \frac{\partial \Phi(s)}{\partial s}\hat{\beta},$$

where $\Phi(s)$ denotes the Chebyshev polynomial basis.

The approximated value function is

$$\hat{V}(s) = \Phi(s)\hat{\beta}.$$

Inclusive wealth is computed as the inner product of shadow prices and stock levels.

For further theoretical details, see Abbott, Fenichel, and Yun (2026), Fenichel and Abbott (2014), and Yun et al. (2017).

Value

A list containing simulation results. Individual elements can be accessed using `results$item` or `results[["item"]]`.

`shadowp`	Matrix of shadow (accounting) prices for each stock.
`iweach`	Inclusive wealth contribution of each stock (multi-stock case).
`iw`	Inclusive wealth.
`vfun`	Approximated value function.
`stock`	Stock values used in the simulation.
`wval`	Welfare values if `wval` is provided.
`model.type`	Either `"deterministic"` or `"stochastic"`.

References

Abbott, Joshua K., Eli P. Fenichel, and Seong D. Yun. (2026). Risky (Natural) Assets: Stochasticity, Nonconvexity, and the Value of Natural Capital. *Journal of the Association of Environmental and Resource Economists*, 13(5), 1269-1309. doi:10.1086/741689

Fenichel, Eli P. and Joshua K. Abbott. (2014). Natural Capital: From Metaphor to Measurement. *Journal of the Association of Environmental Economists*, 1(1/2), 1–27. doi:10.1086/676034

Yun, Seong D., Barbara Hutniczak, Joshua K. Abbott, and Eli P. Fenichel. (2017). Ecosystem-Based Management and the Wealth of Ecosystems. *Proceedings of the National Academy of Sciences*, 114(25), 6539–6544. doi:10.1073/pnas.1617666114

See Also

`AFY` `aproxdef`, `GOM`, `LV`, `plotgen`, `vaprox`, `vaprox.pindyck`, `vaprox.pjump`, `vaprox.split`

Examples

```
# 1-D Deterministic: Reef-fish example (Fenichel and Abbott, 2014)
data("GOM")

param <- GOM$param
simData <- GOM$simData

Aspace <- aproxdef(param$order, param$lowerK, param$upperK, param$delta)

vC <- vaprox(
  Aspace,
  simData$stock,
  simData$sdot,
  simData$profit
)

GOMSimV <- vsim(vC,
                simData$stock,
                simData$profit)

# plot shadow (accounting) price: Figure 4 in Fenichel and Abbott (2014)
plotgen(GOMSimV, xlabel="Stock size, s", ylabel="Shadow price")

## 1-D Stochastic: Pindyck (1984) extension (Abbott et al., 2026)
data("AFY")

Aspace <- aproxdef(35, 0.2, 1.4, 0.05)

## deterministic
vCd <- vaprox(
  Aspace,
  AFY$simData$stock,
  AFY$simData$mus.d,
```

```
  AFY$simData$profit.d
)

simVd <- vsim(vCd,AFY$simData$stock)

# plot value function
plotgen(simVd,ftype="vw",xlabel="Stock Size", ylabel="Value Function")

# plot shadow (accounting) prices
plotgen(simVd,ftype="p",xlabel="Stock Size", ylabel="Shadow Price")

## stochastic
vCs <- vaprox(
  Aspace,
  AFY$simData$stock,
  AFY$simData$mus.s,
  AFY$simData$profit.s,
  AFY$simData$sigs
)

simVs <- vsim(vCs,AFY$simData$stock)

# plot value function
plotgen(simVs,ftype="vw",xlabel="Stock Size", ylabel="Value Function")

# plot shadow (accounting) prices
plotgen(simVs,ftype="p",xlabel="Stock Size", ylabel="Shadow Price")
```

Index