SCILAB *à l'École nationale des ponts et chaussées*

http://cermics.enpc.fr/scilab

Halmstad 2006
Dynamic programming and Markov chains

Jean-Philippe Chancelier

23 novembre 2006 (dernière date de mise à jour)

Table des matières

1 Finite horizon problem

Let $(X_n)_{n\in\mathbb{N}}$ be a finite Markov chain with transition matrix $M^{(n)}$. We want to recursively compute :

$$v_n(x) = \mathbb{E}\left[\sum_{k=n}^{T-1} \frac{1}{(1+r)^{k-n}} c_k(X_k) + \frac{1}{(1+r)^{T-n}} f(T, X_T) | X_n = x\right],$$

Question 1 *Write a first routine which returns a random stochastic matrix M of size* `NxN`*. In scilab there is a function called* `genmarkov` *that you can also check if you want.*

```
function M=my_gen_markov(n);
  P=rand(n,n,'u');
  y=sum(P,'c'); y = ones(y)./y ; y=y*ones(1,N);
  M=y.* P;
endfuncion
```

Question 2 *Choose a state size, generate a stochastic matrix M, then generate and plot some samples of the homogeneous Markov chain with states in $[1, N]$ described by M (using* `grand` *in Scilab).*

```
n=10;
M=my_gen_markov(n)
// generate and draw some typical samples
T=4; // horizon time
m=20; // number of samples
// generate the m-samples
// each sample is of length T
Y=grand(T,'markov',M,ones(m,1));
// just add the initial state to each sample
Y=[ones(m,1),Y];
// you can check the size of Y with size
size(Y)
// draw the first sample
plot(Y(1,:))
// draw the second sample
plot(Y(2,:))
```

Question 3 *Choosing an instantaneous cost c and a final cost f recusively compute the value function $v^n(x)$ and draw the result as a surface (i.e $v(t,x)$). The computation can be stored in a matrix.*

```
function y=c(x); y=x; endfunction
function y=f(x);K=2; y=max(x-K,0); endfunction

states=(1:n)' // possible states
Cv=c(states) // vector of c possible values
fv=f(states) // vector of f possible values
// here you should check that Cv and fv are column vectors

r=0.05; // a discount factor
// I compute the value function and store the
// results in a matrix V
V=zeros(n,T);
V(:,T) = fv;
for i=T-1:-1:1
  V(:,i) = (Cv + M*V(:,i+1))/(1+r);
end
// plot the surface

plot3d(1:n,1:T,V)
```

Question 4 *Using samples of the markov chain for a given starting state evaluate by Monte Carlo the cost function $V(1,1)$ and compare the results with previous question.*

```
// test by monte carlo
// Compute V(1,1) by monte carlo
m=10000;
X0=ones(m,1);
X=grand(T-1,'markov',M,X0);
X=[X0,X];

// fix n
// Approximate the cost by Monte Carlo

n=1;
```

```
Cm= mean((1/(1+r).^(T-n))*f(X(:,T)));
for i=(T-1):-1:n;
  Cm = mean((1/(1+r).^(i-n))*c(X(:,i)))+ Cm;
end

// test

Cm -V(1,1)
```