```
%Clinton Tepper
%Macro Finance
%Model from Eisfeldt 2007. All errors are mine.

clear();

analyzeSigma(1.5);
analyzeSigma(2);
analyzeSigma(5);

function stat=analyzeSigma(sigma)
%set parameter values
    disp(strcat("RESULTS FOR SIGMA = ", num2str(sigma)));

    beta=.96^(.25);
    rb=0;
    tau=.01;

    %markov process for income (from AG 1991)
    y=[.31;.7524;1.3470];  %average income=1
    piy=[.34 .33 .33;.035 .4825 .4825;.035 .4825 .4825];

    %finite dimensional state space
    step=.1; %should be .1, set to 1 for debugging
    bmax=5.1; %should be 5.1
    b=(0:step:bmax)';     %grid steps for bonds are .1*avg. income

    %%%%%%%%%%%%%%begin: My Code
    %Note
    tol=10^-10;
    n = length(b);
    m = 3;

    %first we create our operator
    R = ones(n,n,m);
    I = ones(n,m);

    for j = 1:3
        R(:,:,j)=ones(n,1)*b'.*(1+rb) + y(j)-tau- b*ones(1,n);
    end

    R=(max(R,10^-10000).^(1-sigma))./(1-sigma);

    err = 1;
    iter = 1;
    V = ones(n,m);
    Vm1 = V;

    while err> tol && iter < 10000

        %to get policy and V
        for j = 1:3
            [V(:,j),I(:,j)]=max(R(:,:,j)+...
                piy(j,1)*beta*Vm1(:,1)*ones(1,n)+...
                piy(j,2)*beta*Vm1(:,2)*ones(1,n)+...
                piy(j,3)*beta*Vm1(:,3)*ones(1,n));
        end

        err = max(max(abs(V-Vm1)));
        Vm1=V;

        iter = iter + 1;

    end

    figure();
    hold on;
    plot(b,b(I(:,1)),'+');
    plot(b,b(I(:,2)),'o');
    plot(b,b(I(:,3)),'*');
    title(strcat('b+1 vs b sigma=', num2str(sigma)));
    xlabel('b');
    ylabel('b+1');
    line([0 max(b)],[0 max(b)],'LineStyle','--');
    legend('Low','Med','High','45 deg');
    legend('location','southeast');
    hold off;


    %Calc our stats by MC
    nSims=10^8;
    burnIn = nSims/10; %Just to be safe
    consumption = ones(nSims,1);
    savings = ones(nSims,1);
    state=2;
    bCur=round(n/2);
    bNext=1;
    luck=rand(nSims,1);
    stateHist = ones(nSims,1);
    bHist = ones(nSims,1);

    for i = 1:nSims
        %update the state
        if luck(i) < piy(state,1)
            state = 1;
        elseif luck(i) < piy(state,2)+piy(state,1)
            state = 2;
        else
            state = 3;
        end

        stateHist(i) = state;
        bNext = I(bCur,state);
        consumption(i) = b(bCur) + y(state)- tau - b(bNext);
        savings(i) = b(bNext)-b(bCur);

        bHist(i) = b(bNext);
        bCur = bNext;
    end

    consumption = consumption(burnIn:end);
    %savings = savings(burnIn:end);
    bHist = bHist(burnIn:end);
    stateHist = stateHist(burnIn:end);

    disp(strcat("Mean Consumption:", num2str(mean(consumption))));

    disp(strcat("Consumption Variance:", num2str(var(consumption))));

    bMean = [mean(bHist(stateHist==1))...
        mean(bHist(stateHist==2))...
        mean(bHist(stateHist==3))];


    disp("Average Bond Holdings by Income Level");
    disp(["Income" 1:3;"b" bMean]);

    %Get autocorrellations- note xcorr is fast but requires demeaning
    [acs, lgs] = xcorr(bHist-mean(bHist),10,"coef");

    figure();
    hold on;
    plot(lgs(lgs>=0),acs(lgs>=0),'+');
    title(strcat('Autocorrelations Sigma=', num2str(sigma)));
    xlabel('lags');
    ylabel('Autocorr');
    %legend('Low','Med','High','45 deg');
    %legend('location','southeast');
    hold off;


    %check: invariant dist method

    pmat = zeros(n*m,n*m);
    for i = 1:n
        for j = 1:m
            pmat((j-1)*n+i,[I(i,j);n+I(i,j);(2*n+I(i,j))])=...
                [piy(j,1) piy(j,2) piy(j,3)];
        end
    end

    err=1;
    uProb = ones(n*m,1)*1/(n*m);
    uProbLag = uProb;
    while err>tol && iter<10^5
        uProbLag = uProb;
        uProb = pmat'*uProbLag;
        err = max(max(abs(uProbLag-uProb)));
        iter=iter+1;
    end

    bMean = [(uProb(1:n)' * b(I(1:n)))/sum(uProb(1:n))...
        (uProb(n+1:2*n)' * b(I(n+1:2*n)))/sum(uProb(n+1:2*n))...
        (uProb(2*n+1:3*n)' * b(I(2*n+1:3*n)))/sum(uProb(2*n+1:3*n))];

    disp("Average Bond Holdings by Income Level Calculated");
    disp(["Income" 1:3;"b" bMean]);

    clear();

    stat=1;
end
```