```
clear all
clear all
% Part 1, computing steady state

disp('A16 Steady State Vector Calculated With P^k')

initialdistro=[0.2;0.5;0.3];

initialdistro=initialdistro./norm(initialdistro,1);

% Probabilities of each of the cases. Row 1 is probability that a car goes
% to airport; Row 2 is probability that a car goes to train; Row 3 is
% probability that a car goes to city centre

P=[0.8 0.3 0.3;
   0.1 0.6 0.1;
   0.1 0.1 0.6];

Airport=[];
Train=[];
City=[];

Airport=[Airport initialdistro(1)];
Train=[Train initialdistro(2)];
City=[City initialdistro(3)];
for i=1:500
    finaldistro=(P^i)*initialdistro; % Finding the i-th state vector is computed
    if (finaldistro-P*finaldistro)<1e-15
        break;
    end
Airport=[Airport finaldistro(1)];
Train=[Train finaldistro(2)];
City=[City finaldistro(3)];
end

steadyStateVectorProbability = finaldistro

disp('Diagonalization of A16')

[V D] = eig(P) % Finding E-values and E-vectors to show that algebraic = geometric multiplicity

EigVectors1 = P-eye(3);
EigVectors2 = P-(1/2).*eye(3);
rref(EigVectors1); % find Eigen Vectors in whole numbers
rref(EigVectors2);

diagcheck = V*D*inv(V) % verifying that T = VDV^-1 --> diag check should be equal to original P


% Part 2, calculating steady state with VDV^-1

disp('A16 Steady State Vector Calculated With Diagonalization')

Residence=[];
Library=[];
Athletic=[];
```

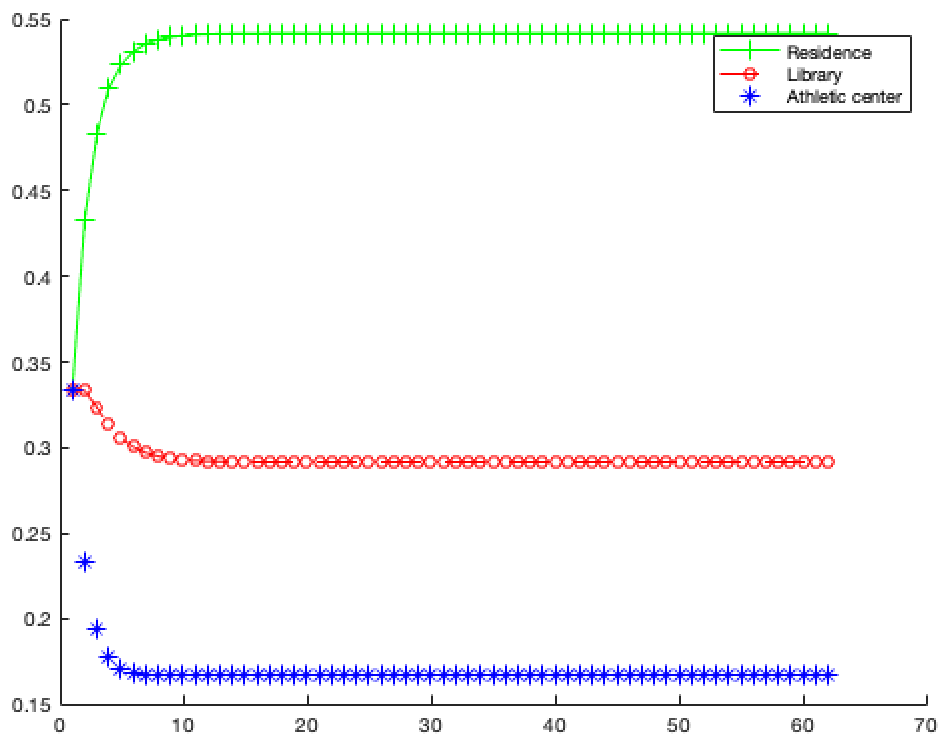
```
Residence=[Residence initialdistro(1)];
Library=[Library initialdistro(2)];
Athletic=[Athletic initialdistro(3)];
for i=1:500
    finaldistro=(P^i)*initialdistro; % computing the ith state vector
    if (finaldistro-P*finaldistro)<1e-15
        break;
    end
Residence=[Residence finaldistro(1)];
Library=[Library finaldistro(2)];
Athletic=[Athletic finaldistro(3)];
end

steadyStateVectorProbability = finaldistro


% Part 3A) when there is a new question (B15)

disp('B15 Steady State Vector Calculated With P^k')

% Random vector showing distribution of cars at each location
initialdistro=[200;200;200];

initialdistro=initialdistro./norm(initialdistro,1);

% Probabilities of each of the cases.
% Row 1: probability that a bike goesto residence;
% Row 2: probability that a bike goes to library
% Row 3: probability that a bike goes to athletic centre

P=[0.8 0.2 0.3;
   0.1 0.7 0.2;
   0.1 0.1 0.5];

Residence=[];
Library=[];
Athletic=[];

Residence=[Residence initialdistro(1)];
Library=[Library initialdistro(2)];
Athletic=[Athletic initialdistro(3)];
for i=1:500
    finaldistro=(P^i)*initialdistro; % computing the ith state vector
    if (finaldistro-P*finaldistro)<1e-15
        break;
    end
Residence=[Residence finaldistro(1)];
Library=[Library finaldistro(2)];
Athletic=[Athletic finaldistro(3)];
end

steadyStateVectorProbability = finaldistro

disp('Diagonalization of B15')

[V D] = eig(P)
```

```
EigVectors1 = P-eye(3);
EigVectors2 = P-(0.6).*eye(3);
Eigenvectors3 = P-(0.4).*eye(3);
rref(EigVectors1) % To find Eigen vectors manually
rref(EigVectors2)
rref(Eigenvectors3)


diagcheck = V*D*inv(V)


% Part 3B), redoing the calculations with diaganalization

disp('B15 Steady State Vector Calculated With Diagonalization')

Residence=[];
Library=[];
Athletic=[];

Residence=[Residence initialdistro(1)];
Library=[Library initialdistro(2)];
Athletic=[Athletic initialdistro(3)];
for i=1:500
    finaldistro=(P^i)*initialdistro; % Finding the i-th state vector is computed
    if (finaldistro-P*finaldistro)<1e-15
        break;
    end
Residence=[Residence finaldistro(1)];
Library=[Library finaldistro(2)];
Athletic=[Athletic finaldistro(3)];
end

steadyStateVectorProbability = finaldistro

hold on
plot(Residence,'g+-')
plot(Library,'ro--')
plot(Athletic,'b*')
legend('Residence','Library','Athletic center')
```

```
A16 Steady State Vector Calculated With P^k

steadyStateVectorProbability =

    0.6000
    0.2000
    0.2000

Diagonalization of A16

V =

    0.9045    0.8165    0.3275
    0.3015   -0.4082   -0.8115
    0.3015   -0.4082    0.4840
```

```
D =

    1.0000         0         0
         0    0.5000         0
         0         0    0.5000


diagcheck =

    0.8000    0.3000    0.3000
    0.1000    0.6000    0.1000
    0.1000    0.1000    0.6000

A16 Steady State Vector Calculated With Diagonalization

steadyStateVectorProbability =

    0.6000
    0.2000
    0.2000

B15 Steady State Vector Calculated With P^k

steadyStateVectorProbability =

    0.5417
    0.2917
    0.1667

Diagonalization of B15

V =

   -0.8498   -0.7071   -0.4082
   -0.4576    0.7071   -0.4082
   -0.2615    0.0000    0.8165


D =

    1.0000         0         0
         0    0.6000         0
         0         0    0.4000


ans =

    1.0000         0   -3.2500
         0    1.0000   -1.7500
         0         0         0


ans =

    1.0000    1.0000         0
         0         0    1.0000
         0         0         0
```

```
ans =

    1.0000         0    0.5000
         0    1.0000    0.5000
         0         0         0


diagcheck =

    0.8000    0.2000    0.3000
    0.1000    0.7000    0.2000
    0.1000    0.1000    0.5000

B15 Steady State Vector Calculated With Diagonalization

steadyStateVectorProbability =

    0.5417
    0.2917
    0.1667
```



Published with MATLAB® R2023b