

MRP with Value function and MDP

July 30, 2020

0.0.1 MRP With Value Function

```
In [1]: import numpy as np
import tensorflow_core as tf
```

```
In [2]: Ptr=np.array([
    [0.3,0.5,0.2],
    [0.1,0.7,0.2],
    [0.4,0.3,0.3]
])
R=np.array([
    [-1],
    [5],
    [-2]
])
```

```
gamma=0.9
```

0.0.2 Analytic solution

```
In [3]: def get_analytic(Ptr,R,gamma):
    idt=np.identity(Ptr.shape[0])
    # Equation:  $V=(I-\text{gamma}*P)^{-1}*R$ 

    Val= np.matmul(np.matrix(idt-gamma*Ptr).I,R)
    print("Value:")
    print(Val)
```

```
In [4]: get_analytic(Ptr,R,gamma)
```

```
Value:
[[18.34360761]
 [25.66068078]
 [15.79737336]]
```

0.0.3 Dynamic Approach Solution

```
In [5]: def get_dynamic_soln(Ptr,R,gamma,epsilon):
        V_curr=np.zeros((Ptr.shape[0],1))
        V_prev=np.ones((Ptr.shape[0],1))

        while np.sqrt(np.square(sum(V_curr-V_prev)))>epsilon:
            V_prev=V_curr
            V_curr= R+gamma*np.matmul(Ptr,V_curr)

        print('Value:')
        print(V_curr)
```

```
In [6]: Ptr=np.array([
        [0.3,0.5,0.2],
        [0.1,0.7,0.2],
        [0.4,0.3,0.3]
        ])
        R=np.array([
        [-1],
        [5],
        [-2]
        ])

        gamma=0.9
        epsilon=0.01

        get_dynamic_soln(Ptr,R,gamma,epsilon)
```

```
Value:
[[18.31485994]
 [25.63193311]
 [15.76862569]]
```

```
In [7]: Ptr=np.array([
        [0.3,0.5,0.2],
        [0.1,0.7,0.2],
        [0.4,0.3,0.3]
        ])
        R=np.array([
        [-1],
        [5],
        [-2]
        ])

        gamma=0.9
```

```
epsilon=0.0001
```

```
get_dynamic_soln(Ptr,R,gamma,epsilon)
```

Value:

```
[[18.34332882]
 [25.660402  ]
 [15.79709457]]
```

0.0.4 Dynamics of MDP

```
In [8]: Ptr=np.array([
        [
            [0,1,0,0,0,0],
            [0,1,0,0,0,0],
            [0,0,1,0,0,0],
            [0,0,0,1,0,0],
            [0,0,0,0,1,0],
            [0,0,0,0,0,1],
        ],
        [
            [0,0,1,0,0,0],
            [0,0,0,1,0,0],
            [0,0,0,0,1,0],
            [0,0,0,0,0,1],
            [0,0,0,0,1,0],
            [0,0,0,0,0,1],
        ],
        [
            [1,0,0,0,0,0],
            [1,0,0,0,0,0],
            [0,0,1,0,0,0],
            [0,0,0,1,0,0],
            [0,0,0,0,1,0],
            [0,0,0,0,0,1],
        ],
        [
            [1,0,0,0,0,0],
            [0,1,0,0,0,0],
            [1,0,0,0,0,0],
            [0,1,0,0,0,0],
            [0,0,1,0,0,0],
            [0,0,0,1,0,0],
        ]
    ])
```

```
In [9]: R=np.array([
        [
```

```

        [-1],
        [-1],
        [-1],
        [-1],
        [-1],
        [10]
    ],
    [
        [-1],
        [-1],
        [-1],
        [10],
        [-1],
        [10]
    ],
    [
        [-1],
        [-1],
        [-1],
        [-1],
        [-1],
        [10]
    ],
    [
        [-1],
        [-1],
        [-1],
        [-1],
        [-1],
        [-1]
    ]
],
])

```

In []: