Exercise 2 - Implement Q-learning Algorithm

```
import gym
import random
import numpy as np
```

Q-table

In the case of FrozenLake game, we have 16 * 4 table of Q-values. We start by initializing the table to be uniform(all zeros).

```
Q = np.zeros([env.observation_space.n,env.action_space.n])
```

```
env = gym.make('FrozenLake-v0') #Load environment
print ("Agent Environment")
env.render()  # Output 4*4 state
rList = [] # Record reward
lr = .85
y = .99     # weight
num_episodes = 2000  #training times
Q = np.zeros([env.observation_space.n,env.action_space.n]) # Initialize Q-table
```

```
Agent Environment

[41mS[0mFFF
FHFH
FFFH
HFFG
```

Use Q-table to make decisions `test`

- set record times(such as 4) to check move times to goals in every episode.

```
def test(i):
    d1 = False
    j1 = 0
    start = 0
    r_sum = 0
    while d1 == False:
        j1 +=1
        a = np.argmax(Q[start,:] + np.random.randn(1,env.action_space.n)*(1./(i+1)))
        s1,r,d1,_ = env.step(a)
        start = s1
        r_sum +=r
    if(r_sum == 1.0):
        print ('To Goal -- Times',j1)
    else:
        print ('Not to Goal')
```

Q-learning Algorithm

```
Initialize Q(s,a) arbitrarily
Repeat (for each episoda):
    Initialize s
    Repeat(for each step of episode):
        Choose a from s using policy derived from Q(e.g., epsilon-greedy)
        Take action a, observe r, s'
        Q(s,a) <— Q(s,a) + alpha[reward + gamma * maxQ(s',a') - Q(s,a)]
        s <— s'
    until s is terminal
```

```
def Q_learning():
    for i in range(num_episodes):
        s = env.reset()  # Reset environment
        rAll = 0
        d = False
        j = 0

        if (i % 500) ==0:  # test Q-table
            test(i)

        while j < 99:  # Q-learning Algorithm
            j+=1
            a = np.argmax(Q[s,:] + np.random.randn(1,env.action_space.n)*(1./(i+1))) # Greedy action
            s1,r,d,_ = env.step(a)  # Obtain new state and new reward
            Q[s,a] = Q[s,a] + lr*(r + y*np.max(Q[s1,:]) - Q[s,a])  # Update Q-table
            rAll += r
            s = s1 # Update state
            if d == True:  # If agent go to the goals, break
                break

        rList.append(rAll)


Q_learning()
print ("Accurately: " +  str(sum(rList)/num_episodes*100) + "%")
print ("Q-Table")
print (Q) # Output Q-Table
```

```
Not to Goal
To Goal -- Times 45
To Goal -- Times 93
Not to Goal
Accurately: 42.35%
Q-Table
[[  6.94393311e-03   9.22273669e-03   7.98197631e-01   1.52587948e-02]
 [  2.54983908e-04   2.40952682e-04   1.70217715e-03   4.79265981e-01]
 [  1.87617220e-03   6.92227004e-03   2.88978635e-03   2.96276911e-01]
 [  6.74502081e-04   1.70874729e-03   4.01216360e-05   1.97994534e-01]
 [  8.81168022e-01   9.52888495e-04   7.63777537e-04   8.19136195e-04]
 [  0.00000000e+00   0.00000000e+00   0.00000000e+00   0.00000000e+00]
 [  4.14501407e-02   3.61209013e-04   6.48114295e-04   6.50031736e-05]
 [  0.00000000e+00   0.00000000e+00   0.00000000e+00   0.00000000e+00]
 [  5.96933584e-04   5.09564402e-06   2.27069141e-04   9.15041882e-01]
 [  0.00000000e+00   3.72380812e-01   1.36572750e-04   5.97140490e-04]
 [  8.81861318e-01   7.06644577e-04   0.00000000e+00   5.61846973e-04]
 [  0.00000000e+00   0.00000000e+00   0.00000000e+00   0.00000000e+00]
 [  0.00000000e+00   0.00000000e+00   0.00000000e+00   0.00000000e+00]
 [  7.04790792e-05   4.28103812e-04   9.73697905e-01   6.24231220e-05]
 [  3.94679883e-03   9.98518477e-01   0.00000000e+00   4.03450101e-03]
 [  0.00000000e+00   0.00000000e+00   0.00000000e+00   0.00000000e+00]]
```

Results

Training for 2000 times: record Intermediate results。

We found that the probability of finding Object is gradually increase.

At the same time, we found that Q-table is hard to extend, because the states of real world or other games maybe too big to describe.

```
Not to Goal
To Goal -- Times 17
Not to Goal
To Goal -- Times 28
Accurately: 55.00000000000001%
Q-Table
[[  6.76287684e-01   5.49837272e-03   8.71674656e-03   1.55124055e-02]
 [  0.00000000e+00   6.35107943e-04   6.47323983e-04   6.20417833e-01]
 [  2.38530988e-03   2.32691945e-03   1.40579745e-03   5.16022897e-01]
 [  0.00000000e+00   6.52400348e-04   0.00000000e+00   3.81566096e-01]
 [  7.44494610e-01   1.13418022e-03   9.10058355e-04   1.04875412e-03]
 [  0.00000000e+00   0.00000000e+00   0.00000000e+00   0.00000000e+00]
 [  6.32231634e-06   4.86780063e-07   4.50010741e-01   1.47883991e-04]
 [  0.00000000e+00   0.00000000e+00   0.00000000e+00   0.00000000e+00]
 [  1.34252195e-03   1.15485742e-03   2.34708130e-03   6.88377116e-01]
 [  7.15184978e-04   8.63895945e-01   3.35047622e-03   0.00000000e+00]
 [  4.98786917e-01   4.65501400e-05   4.89861909e-04   0.00000000e+00]
 [  0.00000000e+00   0.00000000e+00   0.00000000e+00   0.00000000e+00]
 [  0.00000000e+00   0.00000000e+00   0.00000000e+00   0.00000000e+00]
 [  0.00000000e+00   0.00000000e+00   9.16239260e-01   7.06581453e-03]
 [  0.00000000e+00   0.00000000e+00   9.87941446e-01   0.00000000e+00]
 [  0.00000000e+00   0.00000000e+00   0.00000000e+00   0.00000000e+00]]
```

```
Not to Goal
To Goal -- Times 37
To Goal -- Times 61
To Goal -- Times 13
Accurately: 62.050000000000004%
Q-Table
[[  7.93333767e-01   6.77555104e-03   5.23899008e-03   1.09000423e-02]
 [  6.93640163e-04   1.77556889e-04   2.52210767e-03   6.85509029e-01]
 [  1.33294015e-03   6.92357683e-01   9.85167513e-04   0.00000000e+00]
 [  2.64443124e-03   1.15394393e-03   0.00000000e+00   4.34393785e-01]
 [  7.81427835e-01   0.00000000e+00   5.49335092e-04   1.99824010e-04]
 [  0.00000000e+00   0.00000000e+00   0.00000000e+00   0.00000000e+00]
 [  4.66602932e-04   1.05489788e-04   4.71074905e-01   3.25528133e-05]
 [  0.00000000e+00   0.00000000e+00   0.00000000e+00   0.00000000e+00]
 [  0.00000000e+00   1.97708279e-03   1.60363319e-04   6.02597882e-01]
 [  2.02024194e-04   4.44162154e-01   2.93973685e-03   7.82402121e-04]
 [  7.85192420e-01   8.34939676e-04   3.41749560e-04   2.93422913e-04]
 [  0.00000000e+00   0.00000000e+00   0.00000000e+00   0.00000000e+00]
 [  0.00000000e+00   0.00000000e+00   0.00000000e+00   0.00000000e+00]
 [  8.98784529e-03   4.40549688e-04   7.78307858e-01   3.05065096e-03]
 [  0.00000000e+00   9.67771223e-01   0.00000000e+00   0.00000000e+00]
 [  0.00000000e+00   0.00000000e+00   0.00000000e+00   0.00000000e+00]]
```

```
Not to Goal
To Goal -- Times 45
To Goal -- Times 93
Not to Goal
Accurately: 42.35%
Q-Table
[[  6.94393311e-03   9.22273669e-03   7.98197631e-01   1.52587948e-02]
 [  2.54983908e-04   2.40952682e-04   1.70217715e-03   4.79265981e-01]
 [  1.87617220e-03   6.92227004e-03   2.88978635e-03   2.96276911e-01]
 [  6.74502081e-04   1.70874729e-03   4.01216360e-05   1.97994534e-01]
 [  8.81168022e-01   9.52888495e-04   7.63777537e-04   8.19136195e-04]
 [  0.00000000e+00   0.00000000e+00   0.00000000e+00   0.00000000e+00]
 [  4.14501407e-02   3.61209013e-04   6.48114295e-04   6.50031736e-05]
 [  0.00000000e+00   0.00000000e+00   0.00000000e+00   0.00000000e+00]
 [  5.96933584e-04   5.09564402e-06   2.27069141e-04   9.15041882e-01]
 [  0.00000000e+00   3.72380812e-01   1.36572750e-04   5.97140490e-04]
 [  8.81861318e-01   7.06644577e-04   0.00000000e+00   5.61846973e-04]
 [  0.00000000e+00   0.00000000e+00   0.00000000e+00   0.00000000e+00]
 [  0.00000000e+00   0.00000000e+00   0.00000000e+00   0.00000000e+00]
 [  7.04790792e-05   4.28103812e-04   9.73697905e-01   6.24231220e-05]
 [  3.94679883e-03   9.98518477e-01   0.00000000e+00   4.03450101e-03]
 [  0.00000000e+00   0.00000000e+00   0.00000000e+00   0.00000000e+00]]
```