

```

"""
Created on Thu Nov 10 23:26:51 2016
Viterbi
@author: gopal
"""
import numpy as np
import matplotlib.pyplot as plt
O = np.loadtxt('observations.txt', dtype=int)
A = np.loadtxt('transitionMatrix.txt')
B = np.loadtxt('emissionMatrix.txt')
Pi = np.loadtxt('initialStateDistribution.txt')
logA = np.log(A)
logB = np.log(B)
T = len(O) #Timeseries
n = len(Pi) #number of values for each State
m = np.shape(B)[1] #number of values for each Observation

#Lit = after t observations, chances that (State at t) = i after taking the most likely states from 1 to t-1.
L = np.zeros(shape=(n,T))
L[:,0] = np.log(Pi) + logB[:,O[0]]
for t in range(1, T):
    L[:,t] = np.amax((L[:,t-1] + logA), axis=1) + logB[:,O[t]]

#Now backtrack from where we came max.
S_star = np.zeros(T, dtype=int)
S_star[T-1] = np.argmax(L[:,T-1])
for t in range(T-2, 1, -1):
    S_star[t] = np.argmax(L[:,t] + logA[:, S_star[t+1]])

xRange = np.arange(T, dtype=int)
plt.plot(xRange, S_star)
#np.savetxt('a.csv', np.column_stack((xRange, S_star)), delimiter=',', fmt='%d')
plt.ylim(0,n+1)
plt.ylabel("$S_t$")
plt.xlabel("$t$")
plt.title("Plot of most likely $S_t$ verses time")
Char = "ABCDEFGHIJKLMNOPQRSTUVWXYZ"
for i in range(T):
    if S_star[i] != S_star[i-1]:
        print(Char[S_star[i]], i)

```

