Gabriel Buginga

Machine Learning
CPS 863

Lista 7

December 5, 2019

Date Performed:	December 5, 2019
Institution:	PESC/COPPE/UFRJ
Instructors:	Edmundo de Souza e Silva
	Rosa M. M. Leão
	Daniel Sadoc Menasché

Models and theory used is based on [1]. The paper cited for question 4 is [2].

Code in Python

Questao 3

```
#Questao 3
from matplotlib import pyplot as plt
%matplotlib inline
from IPython.core.pylabtools import figsize
from sklearn.calibration import calibration_curve
from sklearn.linear_model import LinearRegression, LogisticRegression
from sklearn.datasets import load_boston
from sklearn import datasets
from sklearn.preprocessing import PolynomialFeatures, StandardScaler
import numpy as np
import pandas as pd
import os

p0,p1,p2= 0.25,0.6,0.15
vo = np.zeros(3)
v = np.zeros(3)

np.amax([25.0,75.0]) *g*

np.max([2,3])
k = 50
g = 0.9
for i in np.arange(500):
    v[1]=np.amax([(p1+p2)*(45+g*vo[1])+p0*(-55+g*vo[2]),(p1+p2)*(100+g*vo[0])+p0*(0
    v[2]=p0*(g*vo[2])+p1*(100+g*vo[1])+p2*(200+g*vo[0])
    v[0]=np.amax([(-55+g*vo[1]),(-100+g*vo[2])])

    vo=v


print (v)
[(-55+g*vo[1]),(-100+g*vo[2])]

[(p1+p2)*(45+g*vo[1])+p0*(-55+g*vo[2]),(p1+p2)*(100+g*vo[0])+p0*(0+g*vo[1])]

p0*(g*vo[2])+p1*(100+g*vo[1])+p2*(200+g*vo[0])

(0.5**(4))*(1-0.5)*((1-0.5)**(5))

((1-0.5)*(0.5))**(5)

x=np.linspace(1000,0,1)
x

x=np.linspace(0,1,1000)
y=((x)**4)*((1-x)**(6))
plt.plot(x,y)
print (x[np.argmax(y)])
```

```
x=np.linspace(0,1,1000)
y=((1-x)*(x))**(5)
plt.plot(x,y)
print (x[np.argmax(y)])

x=np.linspace(0,1,1000)
y=((x)**2)*((1-x)**(5))
plt.plot(x,y)
print (x[np.argmax(y)])

P = [[0.6,0.08,0.02,0.3],[0.4,0.2,0.3,0.1],[0.2,0.3,0,0.5],[1,0,0,0]]
pi = [0,0,1,0]
print (np.dot(np.dot(pi,P),P))
print (np.matmul(pi,P))

np.sum([0.74 , 0.076 ,0.094 ,0.09 ])
```

Questao 7

```
#Questao 7

TRANS = [0.3 , 0.7 ; 0.7 ,0.3];
EMIS = [0.9 , 0.1 ; 0.1 , 0.9];

emission_1 = [1 1 1 1 1 2 2 2 2 2 2]; %HHHHHTTTTTT
emission_2 = [1 2 1 2 1 2 1 2 1 2 1]; %HTHTHTHTHTH

% [seq,states] = hmmgenerate(1000,TRANS,EMIS);
% likelystates = hmmviterbi(seq, TRANS, EMIS);
% sum(states==likelystates)/1000


p = [0.5 , 0.5] %initial state probabilities
TRANS_HAT = [0 p; zeros(size(TRANS,1),1) TRANS];
EMIS_HAT = [zeros(1,size(EMIS,2)); EMIS];

[PSTATES_1,logpseq_1] = hmmdecode(emission_1,TRANS_HAT,EMIS_HAT)
likelystates_1 = hmmviterbi(emission_1, TRANS_HAT,EMIS_HAT)

[PSTATES_2,logpseq_2] = hmmdecode(emission_2,TRANS_HAT,EMIS_HAT)
```

Questao 8

```
#Questao 8

TRANS = [0.3 , 0.7 ; 0.7 ,0.3];
EMIS = [0.9 , 0.1 ; 0.1 , 0.9];

emission_1 = [1 2 1 1 2]; %HTHHT
emission_2 = [1 2 1 2 1 2 1 2 1 2 1]; %HTHTHTHTHTH

% [seq,states] = hmmgenerate(1000,TRANS,EMIS);
% likelystates = hmmviterbi(seq, TRANS, EMIS);
```

```
% sum(states==likelystates)/1000

p = [0.5 , 0.5] %initial state probabilities
TRANS_HAT = [0 p; zeros(size(TRANS,1),1) TRANS];
EMIS_HAT = [zeros(1,size(EMIS,2)); EMIS];


% likelystates_1 = hmmviterbi(emission_1, TRANS_HAT,EMIS_HAT)
% [PSTATES_1,logpseq_1] = hmmdecode(emission_1,TRANS_HAT,EMIS_HAT)

likelystates_2 = hmmviterbi(emission_2, TRANS_HAT,EMIS_HAT)
[PSTATES_2,logpseq_2] = hmmdecode(emission_2,TRANS_HAT,EMIS_HAT);

p = [0.2 , 0.8] %initial state probabilities
TRANS_HAT = [0 p; zeros(size(TRANS,1),1) TRANS];
EMIS_HAT = [zeros(1,size(EMIS,2)); EMIS];

likelystates_2_ = hmmviterbi(emission_2, TRANS_HAT,EMIS_HAT)
[PSTATES_2_,logpseq_2_] = hmmdecode(emission_2,TRANS_HAT,EMIS_HAT);
```

Questao 9

```
#Questao 9

TRANS = [0.75 , 0.33 ; 0.4 ,0.6];

EMIS = [0.05,0.4,0.55;0.8,0.10,0.10];

emission_1 = [1 2 1 3]; %SMSL

% [seq,states] = hmmgenerate(1000,TRANS,EMIS);
% likelystates = hmmviterbi(seq, TRANS, EMIS);
% sum(states==likelystates)/1000

p = [0.5 , 0.5] %initial state probabilities
TRANS_HAT = [0 p; zeros(size(TRANS,1),1) TRANS];
EMIS_HAT = [zeros(1,size(EMIS,2)); EMIS];

likelystates_1 = hmmviterbi(emission_1, TRANS_HAT,EMIS_HAT)
[PSTATES_1,logpseq_1] = hmmdecode(emission_1,TRANS_HAT,EMIS_HAT)
```

References

[1] K. P. Murphy, *Machine learning: a probabilistic perspective*. MIT press, 2012.

[2] A. Srivastava, A. Kundu, S. Sural, and A. Majumdar, "Credit card fraud detection using hidden markov model," *IEEE Transactions on dependable and secure computing*, vol. 5, no. 1, pp. 37–48, 2008.