Dashboard ▶ My courses ▶ Computer Science 315 / Applied Maths 796 ▶ Assignments 1-4 ▶ Assignment 3 Test - Part B/ Werkopdrag 3 Toets - Afdeling B

Started on	Tuesday, 23 April 2019, 12:18
State	Finished
Completed on	Tuesday, 23 April 2019, 12:52
Time taken	34 mins 18 secs
Grade	**0.00** out of 1.00 (**0**%)

Question **1**

Incorrect

Mark 0.00 out of 1.00

We calculate responsibilities in a Gaussian mixutre model with:

Verantwoordelikhede word as volg in 'n Gaussiese mengselmodel bereken:

$$\gamma(z_{nj}) = \frac{\pi_j \mathcal{N}(\mathbf{x}_n | \mathbf{u}_j, \Sigma_j)}{\sum_{i=1}^{k} \pi_i \mathcal{N}(\mathbf{x}_n | \mathbf{u}_i, \Sigma_i)}$$

Write a Python 3 function that implements the above equation. Assume that you have one dimensional data and that you are fitting a two component mixture model. The skeleton of this function is pre-loaded in the answer box below.

Skryf 'n Python 3 funksie wat die bostaande vergelyking implementeer. Aanvaar dat jy met een dimensionele data werk en dat jy 'n twee-komponent mengsel model wil pas. Die bloudruk van die funksie is te vinde in die onderstaande kode antwoordblok.

Recall that the formula for a one-dimensional Gaussian density has the form:

Onthou dat die formule vir 'n een-dimensionele Gaussiese verdeling die volgende vorm het:

$$p(x|\mu, \sigma^2) = \frac{1}{\sqrt{2\pi\sigma^2}} e^{-\frac{(x-\mu)^2}{2\sigma^2}}$$

NB: You may NOT use numpy in formulating your answer. Doing so will lead to syntax errors and in turn will make you loose marks.

NB: Jy mag NIE van numpy gebruik maak nie. As jy van numpy gebruik maak sal dit tot uitvoeringsfoute lei en jy sal gevolglik punte verloor.

For example:

Test
`print(["%.4f"%item for item in estimat`

Answer: (penalty regime: 10, 20, ... %)

Reset answer

Ace editor not ready. Perhaps reload page? Falling back to raw text area.

```
'''
Computes responsibilities. Assumes
one-dimensional data and a two
component mixture model.

@param p: mixture coeffecients.
```

	Test
✗	`print(estimate_gamma([0.4,0.6],[0`
✗	`print(estimate_gamma([0.4,0.6],[0`
✗	`print(estimate_gamma([0.3,0.7],[0`
✗	`print(estimate_gamma([0.6,0.4],[2`

Your code must pass all tests to earn any marks. Try again.

Show differences

Question author's solution:

```
'''
Computes responsibilities. Assumes
one-dimensional data and a two comp
```

```
onent mixture model.

@param p: mixture coefficients.
@type p: 1-dimensional list of floats of length 2.
@param u: class means.
@type u: 1-dimensional list of floats length 2.
@param s: class standard deviations.
@type s: 1-dimensional list of floats of length 2.
@param x: vector of scalar observations
@type x: 1-dimensional list of floats of length n.
@param c: class label
@type c: 1 or 0 [integer]

@return: the calculated responsibility of each observation associated with class c
@rtype: 1-dimensional list of floats of length n
'''
def estimate_gamma(p,u,s,x,c):
    import math
    from math import exp
    from math import sqrt
    #p[0] mixture coefficient for class 0
    #p[1] mixture coefficient for class 1

    #u[0] mean of class 0
    #u[1] mean of class 1

    #s[0] std of class 0
    #s[1] std of class 1

    #c class label (can be either a 0 or 1)

    #x vector of scalar observations
    #You may assume that x will not be empty and that the user will provide valid inputs.

    g = [None]*len(x) #responsibilities

    for k in range(len(x)):
        t0 = (1.0/(sqrt(2*math.pi*s[0]**2)))*exp(-1*((x[k] - u[0])**2)/(2*s[0]**2))
        t1 = (1.0/(sqrt(2*math.pi*s[1]**2)))*exp(-1*((x[k] - u[1])**2)/(2*s[1]**2))
        if c == 0:
            g[k] = (p[0]*t0)/(p[0]*t0+p[1]*t1)
        else:
            g[k] = (p[1]*t1)/(p[1]*t1+p[0]*t0)

    return g
```

Incorrect

Marks for this submission: 0.00/1.00.