Національний технічний університет України

«Київський політехнічний інститут»

Факультет інформатики і обчислювальної техніки

Кафедра обчислювальної техніки

**Лабораторна робота №3.2**

**з курсу: «Інтелектуальні вбудовані системи»**

Виконав:

студент групи ІП-84

Сапбиєв М. Н.

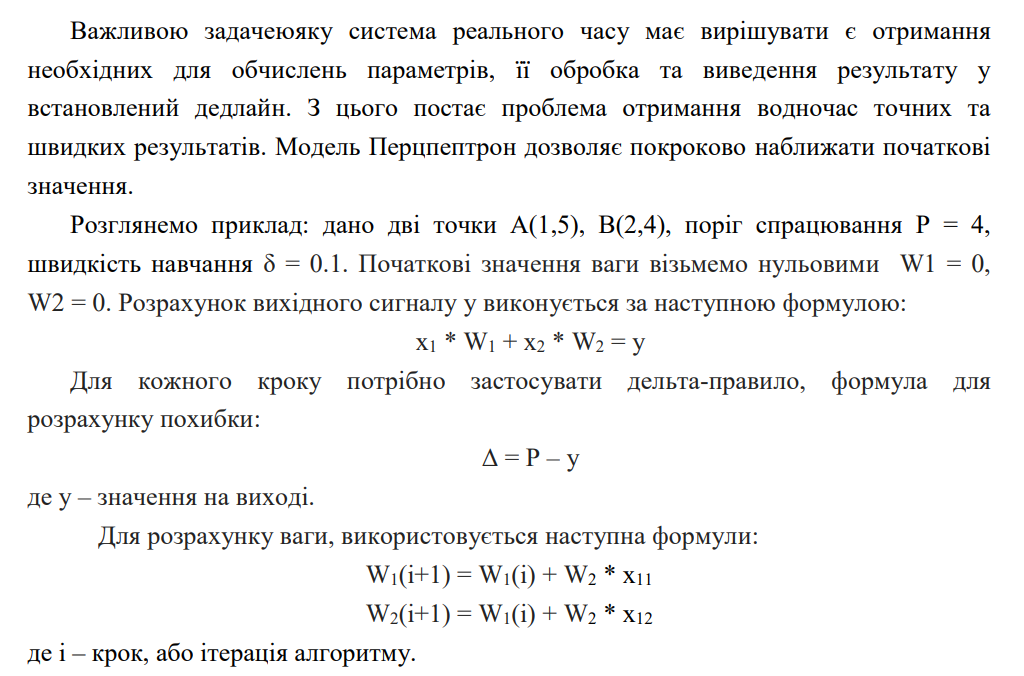
Залікова книжка №8419

Перевірив:

Регіда П. Г.

Київ 2020 р.

**Теоретичні дані:**

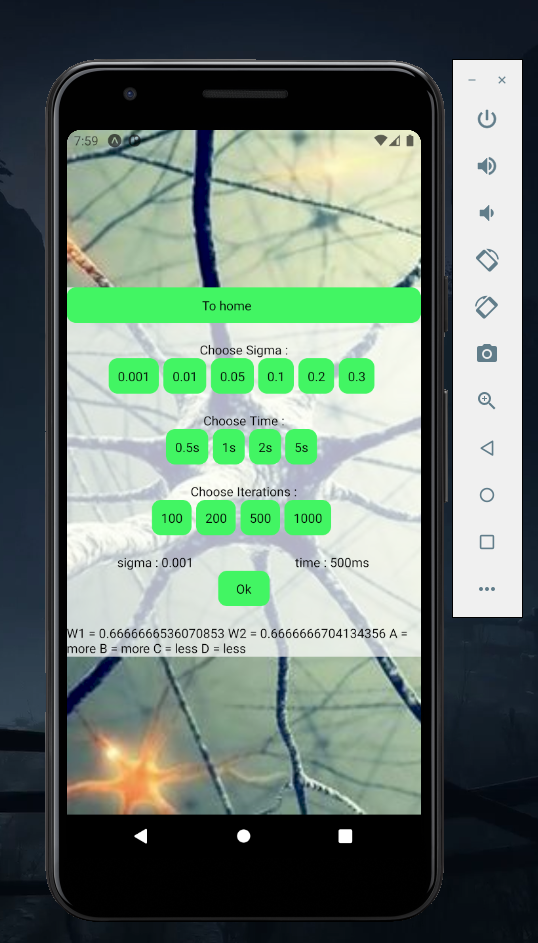


**Код:**

import React, *{*FC, useState*}* from "react";  
import *{*ImageBackground, Text, TouchableOpacity, View*}* from "react-native";  
import *{*NavigationType*}* from "../App";  
import *{****stl****}* from "../lab3.1/Lab31";  
import *{*neuron*}* from "./neuron"  
  
export const Lab32: FC*<{* setNavigation: React.Dispatch*<*React.SetStateAction*<*NavigationType*>> }>* = *({*setNavigation*})* => *{* const *[*sigma, changeSigma*]* = useState*(*'0.001'*)* const *[*time, changeTime*]* = useState*(*'500'*)* const *[*iterations, changeIterations*]* = useState*(*''*)* const *[*result, changeResult*]* = useState*(*''*)* return *<*ImageBackground style=*{****stl***.mainContainer*}* source=*{{*uri: 'https://tadviser.ru/images/1/14/IMG\_20200428\_072748\_881.jpg'*}}>  
 <*View style=*{****stl***.whiteContainer*}>  
 <*TouchableOpacity style=*{****stl***.button*}* onPress=*{()* => setNavigation*(*'Home'*)}>  
 <*Text*>*To home*</*Text*>  
 </*TouchableOpacity*>  
 <*View style=*{****stl***.labContainer*}>  
 <*Text*>*Choose Sigma :*</*Text*>  
 <*View style=*{{*flexDirection: "row"*}}>  
 <*TouchableOpacity style=*{{*...***stl***.button, paddingHorizontal: 10, marginRight: 5*}}* onPress=*{()* => *{* changeSigma*(*'0.001'*)  
 }}>  
 <*Text*>*0.001*</*Text*>  
 </*TouchableOpacity*>  
 <*TouchableOpacity style=*{{*...***stl***.button, paddingHorizontal: 10, marginRight: 5*}}* onPress=*{()* => *{* changeSigma*(*'0.01'*)  
 }}>  
 <*Text*>*0.01*</*Text*>  
 </*TouchableOpacity*>  
 <*TouchableOpacity style=*{{*...***stl***.button, paddingHorizontal: 10, marginRight: 5*}}* onPress=*{()* => *{* changeSigma*(*'0.05'*)  
 }}>  
 <*Text*>*0.05*</*Text*>  
 </*TouchableOpacity*>  
 <*TouchableOpacity style=*{{*...***stl***.button, paddingHorizontal: 10, marginRight: 5*}}* onPress=*{()* => *{* changeSigma*(*'0.1'*)  
 }}>  
 <*Text*>*0.1*</*Text*>  
 </*TouchableOpacity*>  
 <*TouchableOpacity style=*{{*...***stl***.button, paddingHorizontal: 10, marginRight: 5*}}* onPress=*{()* => *{* changeSigma*(*'0.2'*)  
 }}>  
 <*Text*>*0.2*</*Text*>  
 </*TouchableOpacity*>  
 <*TouchableOpacity style=*{{*...***stl***.button, paddingHorizontal: 10, marginRight: 5*}}* onPress=*{()* => *{* changeSigma*(*'0.3'*)  
 }}>  
 <*Text*>*0.3*</*Text*>  
 </*TouchableOpacity*>  
 </*View*>  
  
 <*Text*>*Choose Time :*</*Text*>  
 <*View style=*{{*flexDirection: "row"*}}>  
 <*TouchableOpacity style=*{{*...***stl***.button, paddingHorizontal: 10, marginRight: 5*}}* onPress=*{()* => *{* changeTime*(*'500'*)* changeIterations*(*''*)  
 }}>  
 <*Text*>*0.5s*</*Text*>  
 </*TouchableOpacity*>  
 <*TouchableOpacity style=*{{*...***stl***.button, paddingHorizontal: 10, marginRight: 5*}}* onPress=*{()* => *{* changeTime*(*'1000'*)* changeIterations*(*''*)  
 }}>  
 <*Text*>*1s*</*Text*>  
 </*TouchableOpacity*>  
 <*TouchableOpacity style=*{{*...***stl***.button, paddingHorizontal: 10, marginRight: 5*}}* onPress=*{()* => *{* changeTime*(*'2000'*)* changeIterations*(*''*)  
 }}>  
 <*Text*>*2s*</*Text*>  
 </*TouchableOpacity*>  
 <*TouchableOpacity style=*{{*...***stl***.button, paddingHorizontal: 10, marginRight: 5*}}* onPress=*{()* => *{* changeTime*(*'5000'*)* changeIterations*(*''*)  
 }}>  
 <*Text*>*5s*</*Text*>  
 </*TouchableOpacity*>  
 </*View*>  
  
 <*Text*>*Choose Iterations :*</*Text*>  
 <*View style=*{{*flexDirection: "row"*}}>  
 <*TouchableOpacity style=*{{*...***stl***.button, paddingHorizontal: 10, marginRight: 5*}}* onPress=*{()* => *{* changeTime*(*''*)* changeIterations*(*'100'*)  
 }}>  
 <*Text*>*100*</*Text*>  
 </*TouchableOpacity*>  
 <*TouchableOpacity style=*{{*...***stl***.button, paddingHorizontal: 10, marginRight: 5*}}* onPress=*{()* => *{* changeTime*(*''*)* changeIterations*(*'200'*)  
 }}>  
 <*Text*>*200*</*Text*>  
 </*TouchableOpacity*>  
 <*TouchableOpacity style=*{{*...***stl***.button, paddingHorizontal: 10, marginRight: 5*}}* onPress=*{()* => *{* changeTime*(*''*)* changeIterations*(*'500'*)  
 }}>  
 <*Text*>*500*</*Text*>  
 </*TouchableOpacity*>  
 <*TouchableOpacity style=*{{*...***stl***.button, paddingHorizontal: 10, marginRight: 5*}}* onPress=*{()* => *{* changeTime*(*''*)* changeIterations*(*'1000'*)  
 }}>  
 <*Text*>*1000*</*Text*>  
 </*TouchableOpacity*>  
 </*View*>  
  
 <*View style=*{{*flexDirection: 'row'*}}>  
 <*Text style=*{{*flex: 1, textAlign: "center"*}}>*sigma : *{*sigma*}</*Text*>  
 {*!!time && *<*Text style=*{{*flex: 1, textAlign: "center"*}}>*time : *{*time*}*ms*</*Text*>}  
 {*!!iterations && *<*Text style=*{{*flex: 1, textAlign: "center"*}}>*iterations : *{*iterations*}</*Text*>}  
 </*View*>  
 <*TouchableOpacity style=*{****stl***.ok*}* onPress=*{()* => *{* changeResult*(*neuron*(****Number****(*sigma*)*, ***Number****(*time*)*, ***Number****(*iterations*)))  
 }}>  
 <*Text*>*Ok*</*Text*>  
 </*TouchableOpacity*>  
 <*Text*>{*result*}</*Text*>  
 </*View*>  
 </*View*>  
 </*ImageBackground*>  
  
}*

export const neuron: *(*sigma: number, time: number, iterations: number*)* => string = *(*sigma, time, iterations*)* => *{* const A: Array*<*number*>* = *[*0, 6*]* const B: Array*<*number*>* = *[*1, 5*]* const C: Array*<*number*>* = *[*3, 3*]* const D: Array*<*number*>* = *[*2, 4*]* const ABCD: Array*<*Array*<*number*>>* = *[*A, B, C, D*]* const P: number = 4  
 let W1: number = 0  
 let W2: number = 0  
 let res : Array*<*string*>* = *[*'', '', '', ''*]* const start: number = new ***Date****()*.getTime*()* let iteration: number = 0  
 while *(*new ***Date****()*.getTime*()* - start < time && !!time || iteration < iterations && !!iterations*) {* let delta = 0  
 let index = iteration % ABCD.length  
 delta = P - *(*W1 \* ABCD*[*index*][*0*]* + W2 \* ABCD*[*index*][*1*])* W1 = W1 + delta \* ABCD*[*index*][*0*]* \* sigma  
 W2 = W2 + delta \* ABCD*[*index*][*1*]* \* sigma  
 res*[*index*]* = delta <= 0 ? 'more' : 'less'  
 iteration++  
 *}* return 'W1 = ' + W1 + ' W2 = ' + W2 + ' A = ' + res*[*0*]* + ' B = ' + res*[*1*]* + ' C = ' + res*[*2*]* + ' D = ' + res*[*3*]  
}*

**Результати виконання:**

****

**Висновки:**

В ході виконання лабораторної роботи ми ознайомилися з принципами машинного навчання за допомогою математичної моделі сприйняття інформації Перцептрон(Perceptron). Змоделювали роботу нейронної мережі та дослідили вплив параметрів на час виконання та точність результату