Министерство образования Республики Беларусь

Учреждение образования

БЕЛОРУССКИЙ ГОСУДАРСТВЕННЫЙ УНИВЕРСИТЕТ

ИНФОРМАТИКИ И РАДИОЭЛЕКТРОНИКИ

Факультет компьютерных систем и сетей

Кафедра программного обеспечения информационных технологий

Дисциплина: Методы и алгоритмы принятия решений (МиАПР)

ОТЧЕТ

по лабораторной работе №3

по теме:

«РАЗДЕЛЕНИЕ ОБЪЕКТОВ НА ДВА КЛАССА ПРИ

ВЕРОЯТНОСТНОМ ПОДХОДЕ»

Выполнил

студент: гр. 851006 Верещагин Н.В.

Проверил: Марина И.М.

Минск 2021

Содержание

[1 Постановка задачи 3](#_Toc64459015)

[1.1 Цель работы 3](#_Toc64459016)

[1.2 Исходные данные 3](#_Toc64459017)

[1.3 Результат работы алгоритма 3](#_Toc64459018)

[2 Алгоритм 4](#_Toc64459019)

[3 Решение задачи 6](#_Toc64459020)

# Постановка задачи

## Цель работы

Изучить особенности классификации объектов при вероятностном подходе и научиться находить ошибку классификации.

## Исходные данные

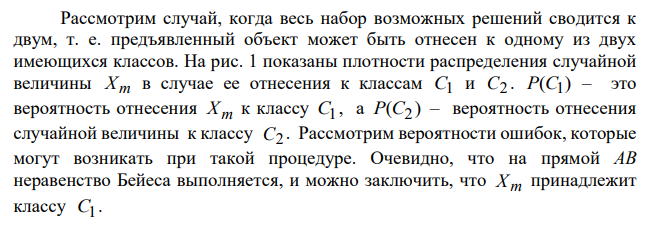
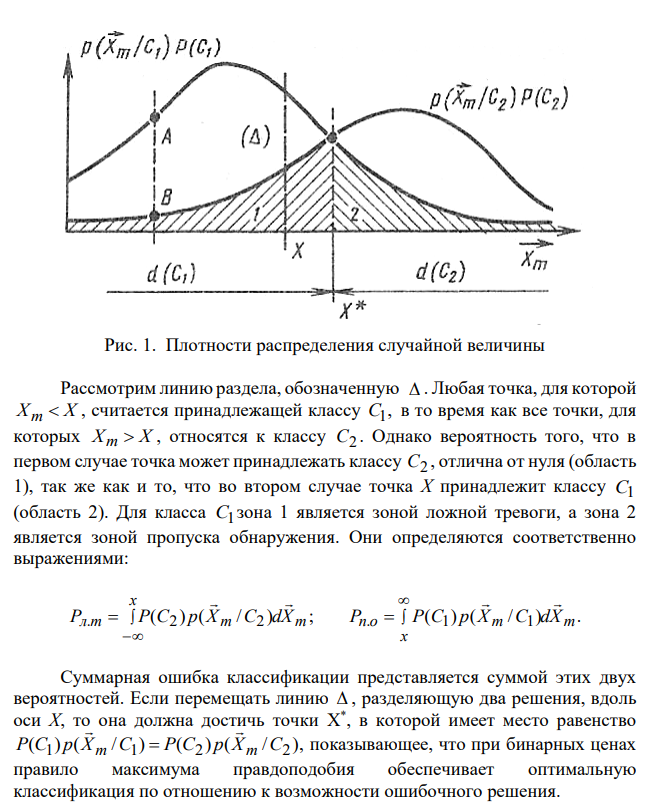
1. Две случайные величины, распределенные по закону Гаусса.

2. Априорные вероятности отнесения каждой из случайных величин к первому из двух классов, в зависимости от того, для какого из них определяется ошибка классификации.

## Результат работы алгоритма

Вероятность ложной тревоги, вероятность пропуска обнаружения ошибки, вероятность суммарной ошибки классификации. Результаты работы программы должны представляться в графическом виде.

# Алгоритм



# Решение задачи

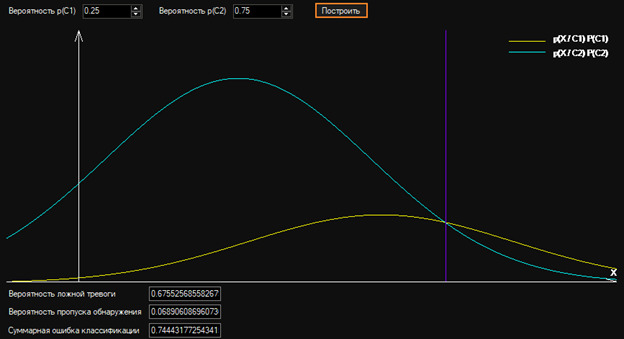


Рисунок 1 – Пример работы программы 1

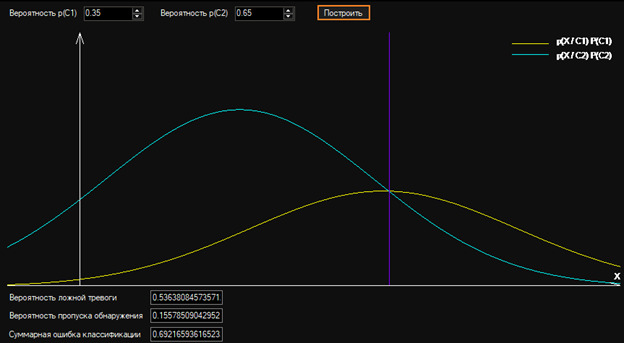


Рисунок 2 – Пример работы программы 2

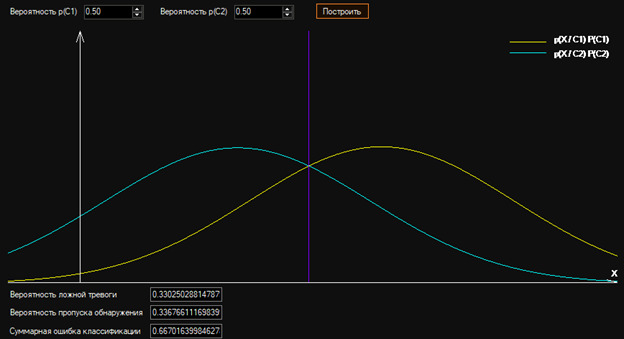


Рисунок 3 – Пример работы программы 3

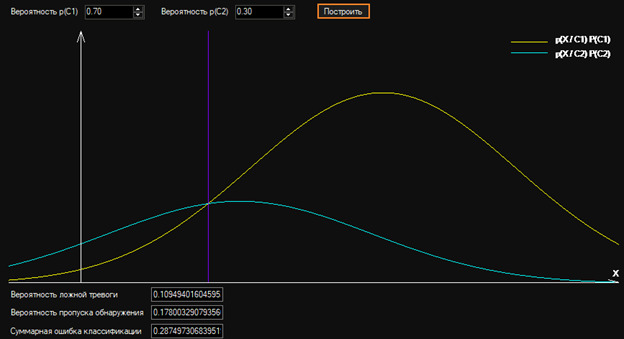


Рисунок 4 – Пример работы программы 4

**Код программы:**

using System;

using System.Collections.Generic;

using System.ComponentModel;

using System.Data;

using System.Drawing;

using System.Linq;

using System.Text;

using System.Threading.Tasks;

using System.Windows.Forms;

namespace Lab3

{

    public partial class Form1 : Form

    {

        private const int pointsCount = 10000;

        private double pc1;

        private double pc2;

        private Random random;

        public Form1()

        {

            InitializeComponent();

        }

        private void buttonRun\_Click(object sender, EventArgs e)

        {

            pc1 = (double)pc1NumericUpDown.Value;

            pc2 = (double)pc2NumericUpDown.Value;

            random = new Random();

            var bitmap = new Bitmap(pictureBox.Width, pictureBox.Height);

            using (Graphics graphics = Graphics.FromImage(bitmap) )

            {

                Do(graphics);

                pictureBox.Image = bitmap;

            }

        }

        private void Do( Graphics graphics)

        {

            var arrOfPoints1 = new int[pointsCount];

            var arrOfPoints2 = new int[pointsCount];

            double max1 = 0;

            double max2 = 0;

            for (int i = 0; i < pointsCount; i++)

            {

                arrOfPoints1[i] = random.Next(100,740);

                arrOfPoints2[i] = random.Next(-100,540);

                max1 += arrOfPoints1[i];

                max2 += arrOfPoints2[i];

            }

            max1 /= pointsCount;

            max2 /= pointsCount;

            double sigma1 = 0;

            double sigma2 = 0;

            for (int i = 0; i < pointsCount; i++)

            {

                sigma1 += Math.Pow(arrOfPoints1[i] - max1,2);

                sigma2 += Math.Pow(arrOfPoints2[i] - max2,2);

            }

            sigma1 = Math.Sqrt(sigma1/pointsCount);

            sigma2 = Math.Sqrt(sigma2/pointsCount);

            var result1 = new double[pictureBox.Width];

            var result2 = new double[pictureBox.Width];

            result1[0] = (Math.Exp(-0.5 \* Math.Pow((-100 - max1) / sigma1, 2)) /

                    (sigma1 \* Math.Sqrt(2 \* Math.PI)) \* pc1);

            result2[0] =

                    (Math.Exp(-0.5 \* Math.Pow((-100 - max2) / sigma2, 2)) /

                    (sigma2 \* Math.Sqrt(2 \* Math.PI)) \* pc2);

            int D = 0;

            for (int x = 1; x < pictureBox.Width; x++)

            {

                result1[x] =

                    (Math.Exp(-0.5\*Math.Pow((x-100 - max1)/sigma1, 2))/

                    (sigma1\*Math.Sqrt(2\*Math.PI))\*pc1);

                result2[x] =

                    (Math.Exp(-0.5\*Math.Pow((x-100 - max2)/sigma2, 2))/

                    (sigma2\*Math.Sqrt(2\*Math.PI))\*pc2);

                if (Math.Abs(result1[x]\*500 - result2[x]\*500) < 0.002) D = x;

                graphics.DrawLine(Pens.Blue,

                     new Point(x - 1, (pictureBox.Height - (int)(result1[x-1]\*pictureBox.Height\*500))),

                    new Point(x, (pictureBox.Height - (int)(result1[x] \* pictureBox.Height \* 500))));

                graphics.DrawLine(Pens.Red,

                     new Point(x - 1, (pictureBox.Height - (int)(result2[x - 1] \* pictureBox.Height \* 500))),

                    new Point(x, (pictureBox.Height - (int)(result2[x] \* pictureBox.Height \* 500))));

            }

            double error1 = result2.Take((int)D).Sum();

            double error2;

            if (pc1 > pc2)

            {

                error2 = result2.Skip((int) D).Sum();

            }

            else

            {

                error2 = result1.Skip((int) D).Sum();

            }

            using (var textBrush = new SolidBrush(Color.Black))

            {

                graphics.DrawLine(Pens.Chartreuse, D, 0, D, pictureBox.Height);

                graphics.DrawLine(Pens.Black, 0, pictureBox.Height - 1,

                    pictureBox.Width, pictureBox.Height - 1);

                graphics.DrawLine(Pens.Black, pictureBox.Width,

                    pictureBox.Height - 1, pictureBox.Width - 15,

                    pictureBox.Height - 5);

                graphics.DrawLine(Pens.Black, 100, pictureBox.Height - 1, 100, 0);

                graphics.DrawLine(Pens.Black, 100, 0, 95, 15);

                graphics.DrawLine(Pens.Black, 100, 0, 105, 15);

                graphics.DrawString("X", this.Font, Brushes.Black,

                    pictureBox.Width - 10, pictureBox.Height - 20);

                graphics.DrawLine(Pens.Blue, pictureBox.Width - 150, 15,

                    pictureBox.Width - 100, 15);

                graphics.DrawString("p(X / C1) P(C1)", this.Font, textBrush,

                    pictureBox.Width - 90, 5);

                graphics.DrawLine(Pens.Red, pictureBox.Width - 150, 30,

                    pictureBox.Width - 100, 30);

                graphics.DrawString("p(X / C2) P(C2)", this.Font, textBrush,

                    pictureBox.Width - 90, 25);

            }

            error1TextBox.Text = error1.ToString();

            error2TextBox.Text = error2.ToString();

            sumErrorTextBox.Text = (error1 + error2).ToString();

        }

    }

}