

Министерство транспорта Российской Федерации
Федеральное агентство железнодорожного транспорта
Федеральное государственное бюджетное образовательное учреждение
высшего образования
«Дальневосточный государственный университет путей сообщения»
Кафедра «Вычислительная техника и компьютерная графика»

МОДЕЛИРОВАНИЕ ЭКСПЕРИМЕНТАЛЬНЫХ ДАННЫХ ДЛЯ РЕШЕНИЯ ЗАДАЧ РАСПОЗНАВАНИЯ ОБРАЗОВ

Лабораторная работа №1

ЛР 09.04.01.МРО.08.01.МО921ИВС

Выполнил

студент гр. МО921ИВС

А.Ю. Панченко

Проверил

доцент, к.ф.-м.н.

Ю.В. Пономарчук

Цель работы: подготовка экспериментального материала для решения задач распознавания образов.

1 УСЛОВИЕ ЗАДАЧИ

Выполнить генерацию трех наборов нормально распределенных двумерных случайных векторов, объема 200 штук каждый, с ненулевыми ковариациями между компонентами.

1.1 Описание процесса решения

Произвольным образом зададим векторы математических ожиданий $M1$, $M2$, $M3$, размерности 2×1 , и ковариационные матрицы $B1$, $B2$, $B3$, размерности 2×2 для каждого класса соответственно.

$$\begin{aligned} M1 &= \begin{pmatrix} 6,1 \\ 4,5 \end{pmatrix}, B1 = \begin{pmatrix} 3,1 & 2,8 \\ 2,8 & 3,4 \end{pmatrix} \\ M2 &= \begin{pmatrix} -4,2 \\ 4,2 \end{pmatrix}, B2 = \begin{pmatrix} 2,2 & -1,1 \\ -1,1 & 2,6 \end{pmatrix} \\ M3 &= \begin{pmatrix} 2,5 \\ -4 \end{pmatrix}, B3 = \begin{pmatrix} 2,6 & 2,1 \\ 2,1 & 3,1 \end{pmatrix} \end{aligned}$$

Описанные исходные данные нормального распределения задаются в программе следующим образом:

Листинг 1 – Исходные данные нормального распределения

```
var xM1 = 6.1;
var yM1 = 4.5;
var B1varX = 3.1;
var B1varY = 3.4;
var B1covXY = 2.8;
var B1covYX = 2.8;

var xM2 = -4.2;
var yM2 = 4.2;
var B2varX = 2.2;
var B2varY = 2.6;
var B2covXY = -1.1;
var B2covYX = -1.1;

var xM3 = 2.5;
var yM3 = -4;
var B3varX = 2.6;
var B3varY = 3.1;
```

```

var B3covXY = 2.1;
var B3covYX = 2.1;

var distributions = new (Vector<double> M, Matrix<double> B)[]
{
    (
        Vector<double>.Build.DenseOfArray(new[] { xM1, yM1 }),
        Matrix<double>.Build.DenseOfArray(new[,] { { B1varX ,
B1covXY }, { B1covYX, B1varY } })
    ),
    (
        Vector<double>.Build.DenseOfArray(new[] { xM2 , yM2 }),
        Matrix<double>.Build.DenseOfArray(new[,] { { B2varX ,
B2covXY }, { B2covYX, B2varY } })
    ),
    (
        Vector<double>.Build.DenseOfArray(new[] { xM3, yM3 }),
        Matrix<double>.Build.DenseOfArray(new[,] { { B3varX,
B3covXY }, { B3covYX, B3varY } })
    )
};

```

Проверка корректности подобранной матрицы В осуществляется в методе `ValidateMatrix`. Это нужно, чтобы проверить, что матрица является положительно определенной и соответственно дальнейшее разложение будет успешным.

Листинг 2 – Проверка матрицы В

```

private static void ValidateMatrix(Matrix<double> B)
{
    var eigen = B.Evd();
    var eigenValues = eigen.EigenValues;

    // Проверка собственных значений матрицы В на
    // положительность
    if (eigenValues.Real().Any(v => v <= 0))
    {
        throw new Exception("Error: Matrix B is not positive
definite.");
    }
}

```

Далее определяются параметры линейного преобразования и матрица линейного преобразования.

Листинг 3 – Матрица линейного преобразования

```

private static Matrix<double> CalculateA(Matrix<double> B){
    double R00 = B[0, 0];
    double R01 = B[0, 1];
}

```

```

double R11 = B[1, 1];

double A00 = Math.Sqrt(R00);
double A01 = 0;
double A10 = R01 / A00;
double A11 = Math.Sqrt(R11 - A10 * A10);

if (double.IsNaN(A00) || double.IsNaN(A10) ||
double.IsNaN(A11))
{
    throw new Exception("Error: Invalid values in matrix
A.");
}

var A = Matrix<double>.Build.DenseOfArray(new[, ]
{
    { A00, A01 },
    { A10, A11 }
});

return A;
}

```

Далее производится генерация $N = 200$ реализаций случайного вектора, компоненты которого – суть независимые и нормально распределенные $N(1,0)$ случайные величины.

Листинг 4 – Генерация N реализаций случайного вектора

```

private static Matrix<double> CalculateY(int N, Random random)
{
    return Matrix<double>.Build.Dense(2, N, (i, j) =>
Normal.Sample(random, 0, 1));
}

```

Далее выполняется генерация $N = 200$ реализаций случайного вектора с требуемым нормальным законом распределения $N(\bar{M}, B)$.

Листинг 5 – Генерация N реализаций случайного вектора с требуемым нормальным законом распределения

```

private static Matrix<double> CalculateX(Matrix<double> A,
Matrix<double> Y, int N, Vector<double> M)
{
    var X = A * Y;

    // Из-за разницы типов данных, +M выполняется так
    for (var i = 0; i < N; i++)
    {
        X.SetColumn(i, X.Column(i) + M);
    }
}

```

```
    return X;  
}
```

Далее выводятся графически полученные наборы данных объема $N=200$ (рисунки 1-3).

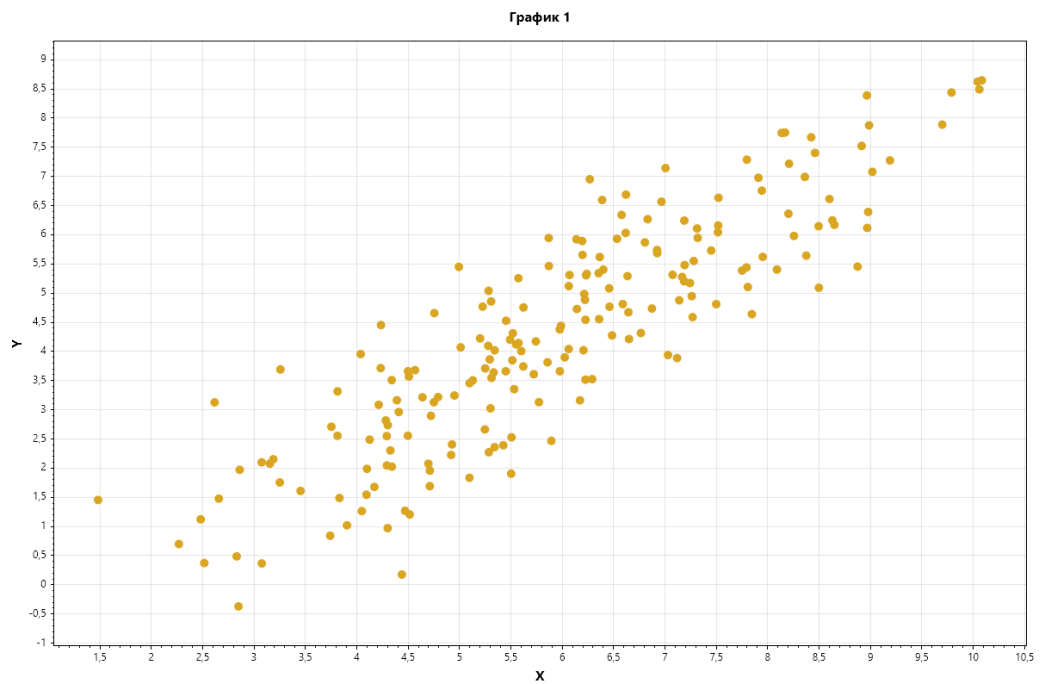


Рисунок 1 – Первая выборка

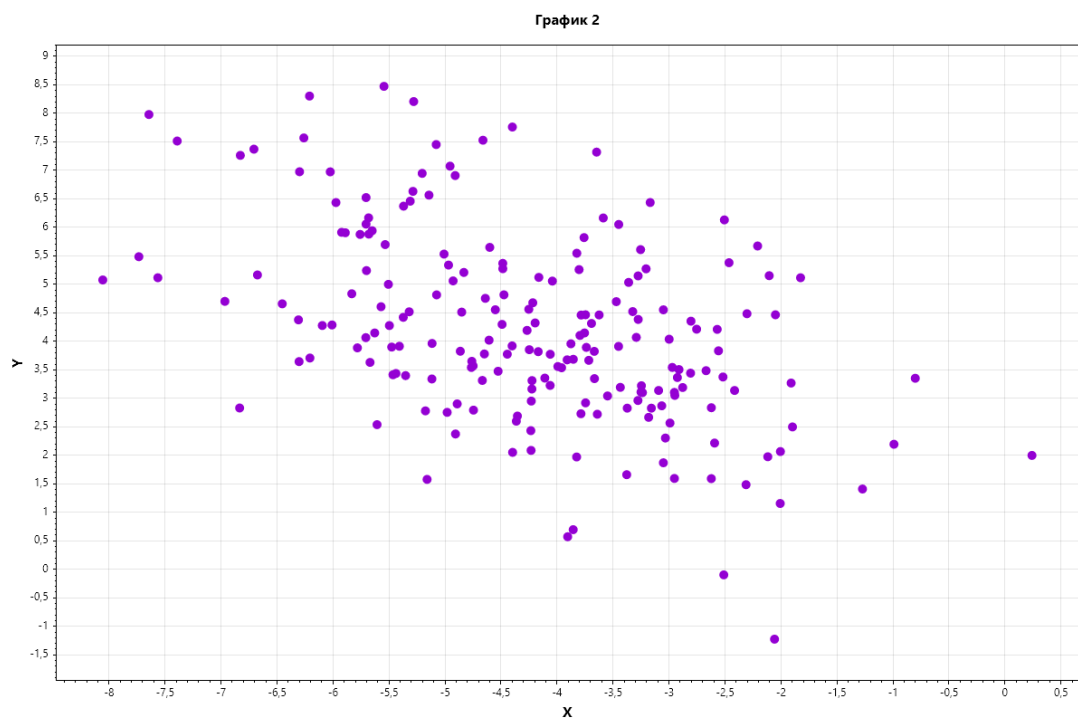


Рисунок 2 – Вторая выборка

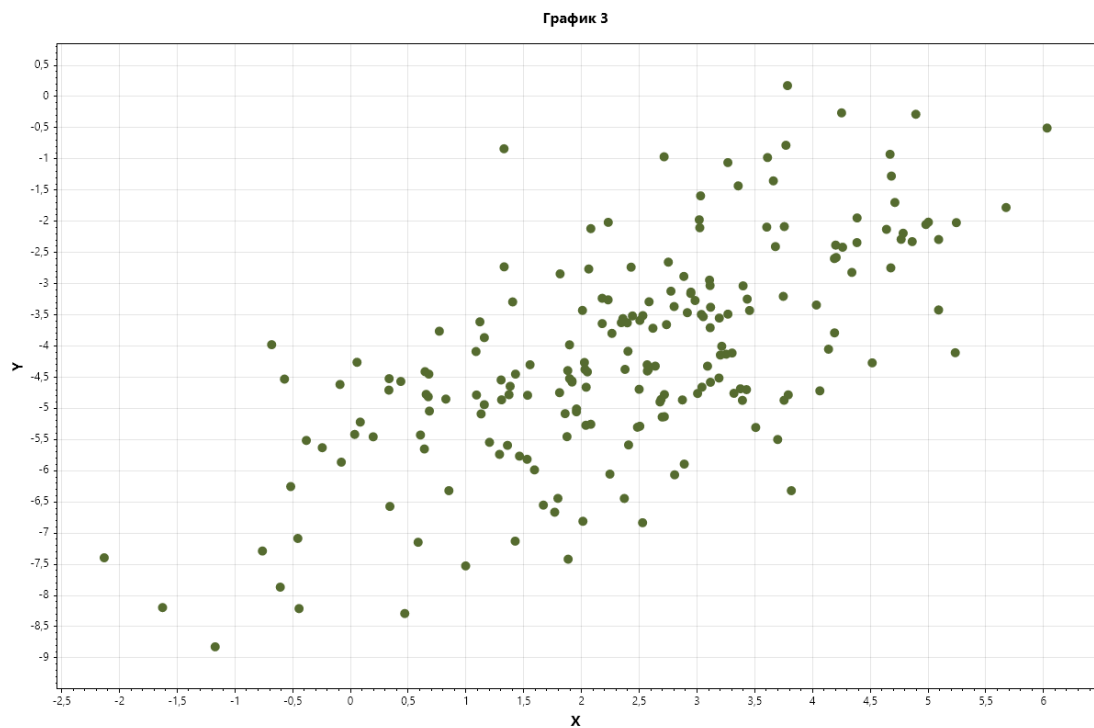


Рисунок 3 – Третья выборка

Все выборки на одном графике, представлены на рисунке 4.

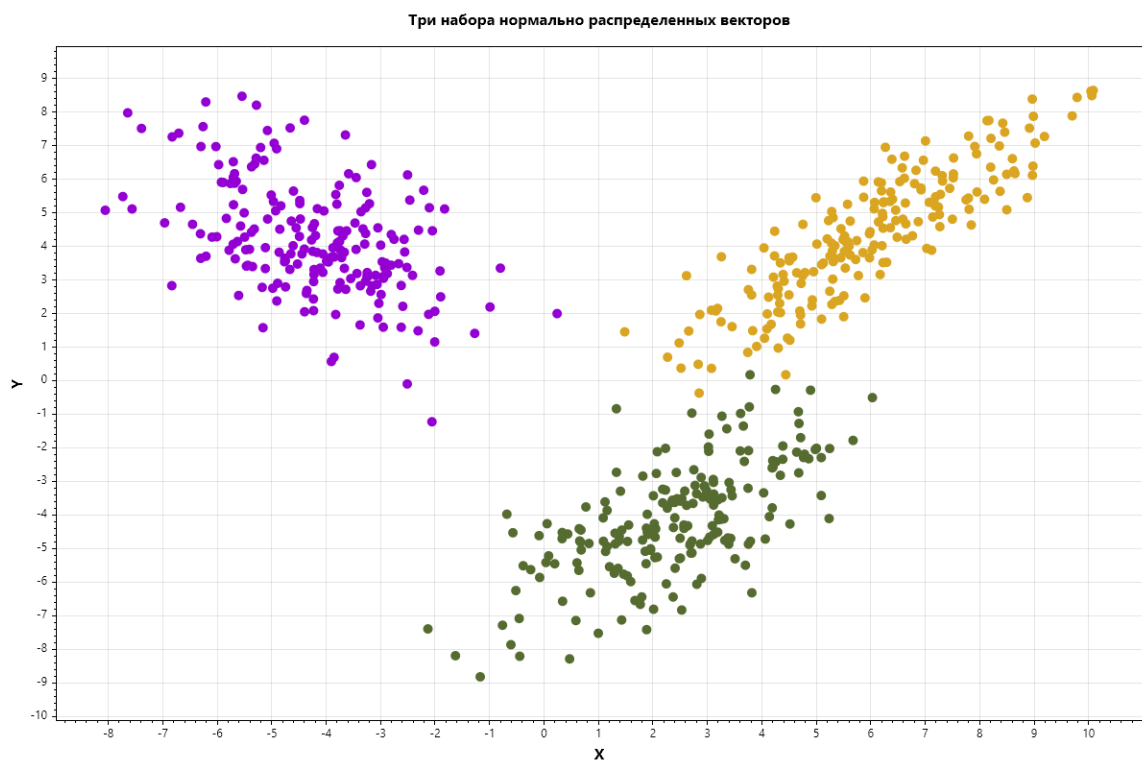


Рисунок 4 – Общий график

Вывод: в ходе выполнения лабораторной работы был подготовлен экспериментальный материал для решения задач распознавания образов.