云南大学数学与与统计学院
上机实践报告

课程名称：信息论基础实验	年级：	上机实践成绩：
指导教师：陆正福	姓名：	庞海藏
上机实践名称：信息论中常用函数的图形绘制	学号：	上机实践日期：
上机实践编号：No. 1	组号：	上机实践时间：

一、实验目的

熟悉信息论中常用函数的图形，为后继学习奠定直观实验基础

二、实验内容

图形框架：Jframe.java

```
package com.hai.jframe;

import java.awt.BorderLayout;

public class Jframe extends JFrame {
    private JPanel jp;

    public static void main(String[] args) {
        EventQueue.invokeLater(new Runnable() {
            public void run() {
                try {
                    Jframe frame = new Jframe();
                } catch (Exception e) {
                    e.printStackTrace();
                }
            }
        });
    }

    public Jframe() {
        jp = new JPanel();
        jp.setBackground(Color.BLACK);
```

```
		setDefaultCloseOperation(JFrame.EXIT_ON_CLOSE);
		getContentPane().setLayout(new BorderLayout(0, 0));
		this.setBackground(Color.BLACK);
		this.setContentPane(jp);
		this.setSize(800, 600);
		this.setLocationRelativeTo(null);
		this.setVisible(true);
	}
}
```

框架面板：

```
package com.hai.panel;

import java.awt.Color;

public class JPanel extends JComponent {

	int width, height;
	int zx, zy;
	int unit = 100;

	public void paintComponent(Graphics g) {
		width = this.getWidth();
		height = this.getHeight();
		zx = width / 2; // 新坐标系原点坐标位置，可以改变
		zy = height / 2;
		this.paintAxis(g);

		this.paintMethod(g);

	}

	private double alterX(double x) { // 新坐标对应的原坐标
		return x + zx;
	}
```

```
    private double alterY(double y) { // 新坐标对应的原坐标
        return -1 * (y - zy);
    }

    private void paintMethod(Graphics g1) {
        int   flag=0;
        double s = 0,r=0;
        System.out.println("==========所有的函数如下，请选择需
要绘画的函数===========");
        System.out.println("1、y=lnx");
        System.out.println("2、y=ln(x)-x+1");
        System.out.println("3、y=xlnx");
        System.out.println("4、y=ln(x)/x");
        System.out.println("5、y=H(x)=-xlnx-(1-x)ln(1-x)");
        System.out.println("6、D(p||q)(given q)");
        System.out.println("7、D(p||q)(given p)");
        System.out.println("8、I(X;Y)(given p(y|x))");
        System.out.println("9、I(X;Y)(given p(x))");
        System.out.println("\n请输入选项：");
        Scanner input=new Scanner(System.in);
        flag=input.nextInt();
        if(flag==6||flag==8||flag==9){
            System.out.println("\n请输入给定的s=");
            s=input.nextDouble();
        }
        if(flag==7){
            System.out.println("\n请输入给定的r=");
            r=input.nextDouble();
        }
        Point2D temp1, temp2;
        double x, y;
        Graphics2D g = (Graphics2D) g1;
        g.setColor(Color.GREEN);
```

```
		x = -1.0 * zx / unit;
		y = Math.pow(x, 3);
		temp1 = new Point2D.Double(this.alterX(x * unit), this.alterY(y
* unit));
		for (int i = 0; i < width; i++) {
			x += 1.0 / unit;
			switch (flag) {
			case 0:
				System.out.println("对不起！没有相应的函数图
像。");
				break;
			case 1:
				y =new FunctionService().function1(x);
				break;
			case 2:
				y =new FunctionService().function2(x);
				break;
			case 3:
				y =new FunctionService().function3(x);
				break;
			case 4:
				y =new FunctionService().function4(x);
				break;
			case 5:
				y =new FunctionService().function5(x);
				break;
			case 6:
				y =new FunctionService().function6(x,s);
				break;
			case 7:
				y =new FunctionService().function7(x,r);
				break;
			case 8:
				y =new FunctionService().function8(x,s);
				break;
```

```
            case 9:
                y =new FunctionService().function9(x,s);
                break;
            }
            if (Math.abs(y) < zy) {
                temp2 = new Point2D.Double(this.alterX(x * unit),
this.alterY(y* unit));
                g.draw(new Line2D.Double(temp1, temp2));
                temp1 = temp2;
            }
        }
    }
    // 画坐标轴,可以通过zx,zy的处置来改变原点位置;
    private void paintAxis(Graphics g) {
        g.setColor(Color.yellow);
        // 画X轴
        g.drawLine(10, zy, width-30, zy);
        //X箭头
        g.drawLine(width-40, zy+10, width-30, zy);
        g.drawLine(width-40, zy-10, width-30, zy);
        // 画Y轴
        g.drawLine(zx, 20, zx, height);
        //Y箭头
        g.drawLine(zx-10, 30, zx, 20);
        g.drawLine(zx+10, 30, zx, 20);
        g.setColor(Color.RED);
        g.setFont(new Font("ScanSerif", Font.BOLD, 12));
        // "Y轴"
        g.drawString("Y轴", zx-40, 40);
        // "X轴"
        g.drawString("X轴", width-50, zy+40);
        g.drawString("0", zx + 2, zy + 12); // 画原点数字
        for (int i = 1; i * unit < width; i++) {
            g.setColor(Color.white);
        g.drawLine(zx + i * unit, height, zx + i * unit, -height);
```

```
                g.drawLine(zx - i * unit, height, zx - i * unit,-height);
            g.drawLine(-width, zy + i * unit, width, zy + i * unit);
                g.drawLine(-width, zy - i * unit, width, zy - i * unit);

                g.setColor(Color.green);
                g.drawLine(zx + i * unit, zy - 10, zx + i * unit, zy);// x 正向
                g.drawLine(zx - i * unit, zy - 10, zx - i * unit, zy);// x 负向
                g.drawString(String.valueOf(i), zx + i * unit - 10, zy + 12);
// x轴数字
               g.drawString(String.valueOf(i * -1), zx - i * unit-10, zy + 12);
// x轴数字

                g.drawLine(zx, zy + i * unit, zx + 10, zy + i * unit);// y 负向
                g.drawLine(zx, zy - i * unit, zx + 10, zy - i * unit);// y 正向
                g.drawString(String.valueOf(i), zx - 12, zy - i * unit + 10); //
y轴数字
                g.drawString(String.valueOf(i * -1), zx - 12, zy + i * unit +
10); // y轴数字
            }
        }
}
```

函数接口：FunctionInterface.java

```
package com.hai.functionInterface;
/**
 * 函数接口
 * @author hai
 *
 */
public interface FunctionInterface {
    public double function1(double x);
    public double function2(double x);
    public double function3(double x);
    public double function4(double x);
    public double function5(double x);
    public double function6(double x,double s);
    public double function7(double x,double r);
    public double function8(double x,double s);
    public double function9(double x,double s);
}
```

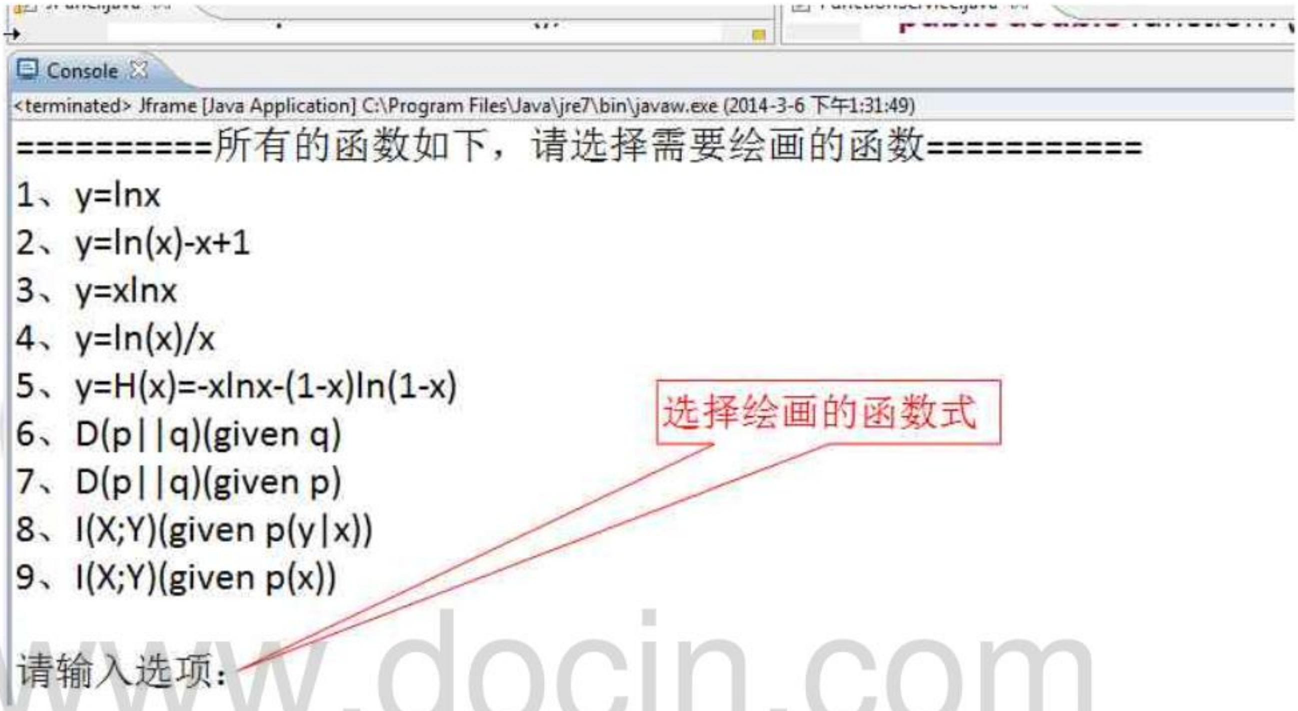
函数服务：

```
package com.hai.functionService;

import com.hai.functionInterface.FunctionInterface;

/**
 *  函数实现类
 * @author hai
 *
 */
public class FunctionService implements FunctionInterface{

    public double function1(double x) {
        // TODO Auto-generated method stub
        return (Math.log((double)x)/Math.log((double)2));
    }

    public double function2(double x) {
        // TODO Auto-generated method stub
        return (Math.log((double)x)/Math.log((double)2))-x+1;
    }

    public double function3(double x) {
        // TODO Auto-generated method stub
        return x*(Math.log((double)x)/Math.log((double)2));
    }

    public double function4(double x) {
```

```
		// TODO Auto-generated method stub
		return (Math.log((double)x)/Math.log((double)2))/x;
	}

	public double function5(double x) {
		// TODO Auto-generated method stub
		return -x*(Math.log((double)x)/Math.log((double)2))-(1-x)*(Math.log((double)(1-x))/Math.log((double)2));
	}

	public double function6(double x ,double s) {
		// TODO Auto-generated method stub
		double r=x;
		return (1-r)*(Math.log((double)((1-r)/(1-s)))/Math.log((double)2))+r*(Math.log((double)(r/s))/Math.log((double)2));
	}

	public double function7(double x,double r) {
		// TODO Auto-generated method stub
		double s=x;
		return (1-s)*(Math.log((double)((1-s)/(1-r)))/Math.log((double)2))+s*(Math.log((double)(s/r))/Math.log((double)2));
	}

	public double function8(double x,double s) {
		// TODO Auto-generated method stub
		return 0;
	}

	public double function9(double x,double s) {
		// TODO Auto-generated method stub
		return 0;
	}
```

```
}
```

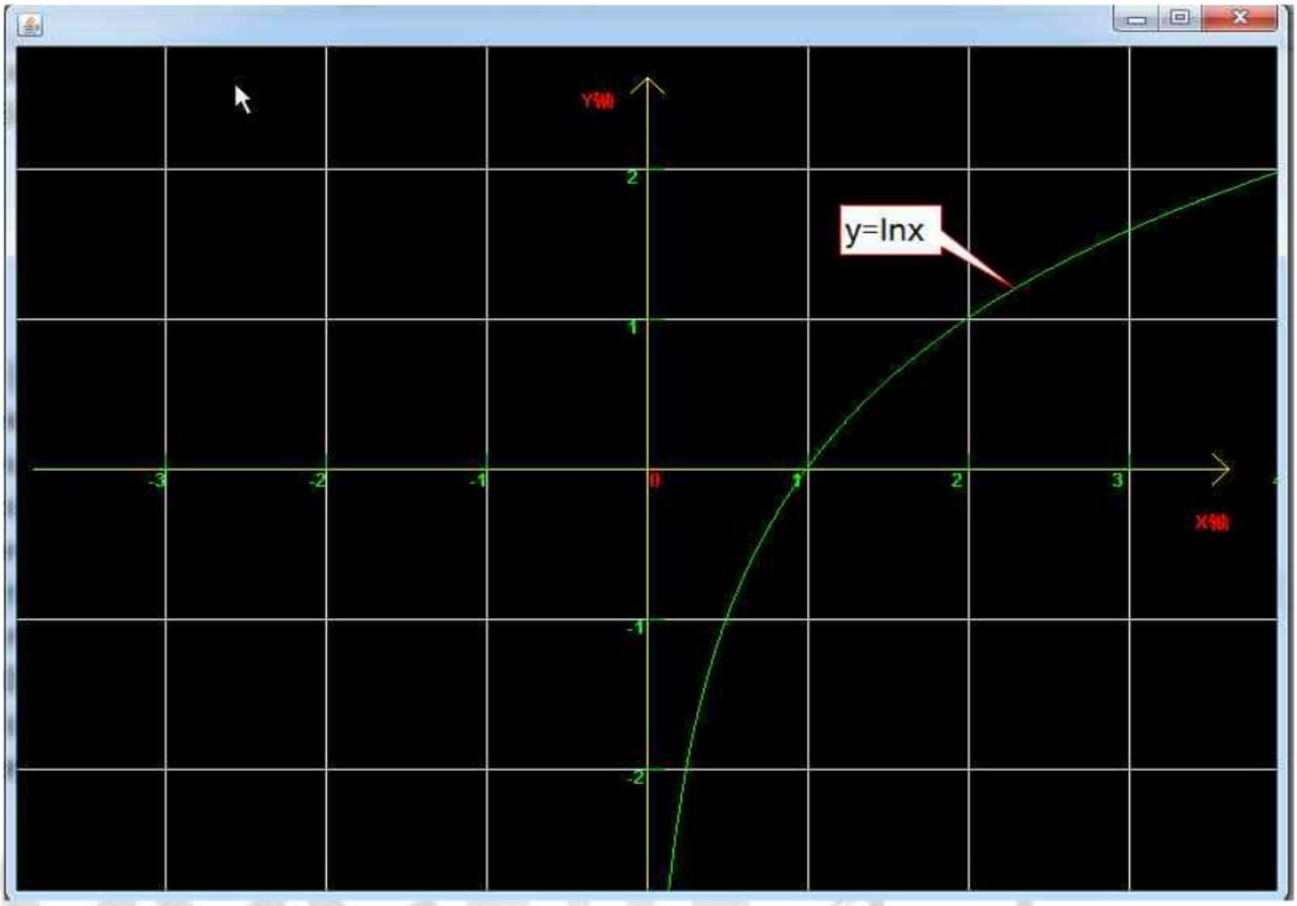
绘制信息论中常用函数的图形



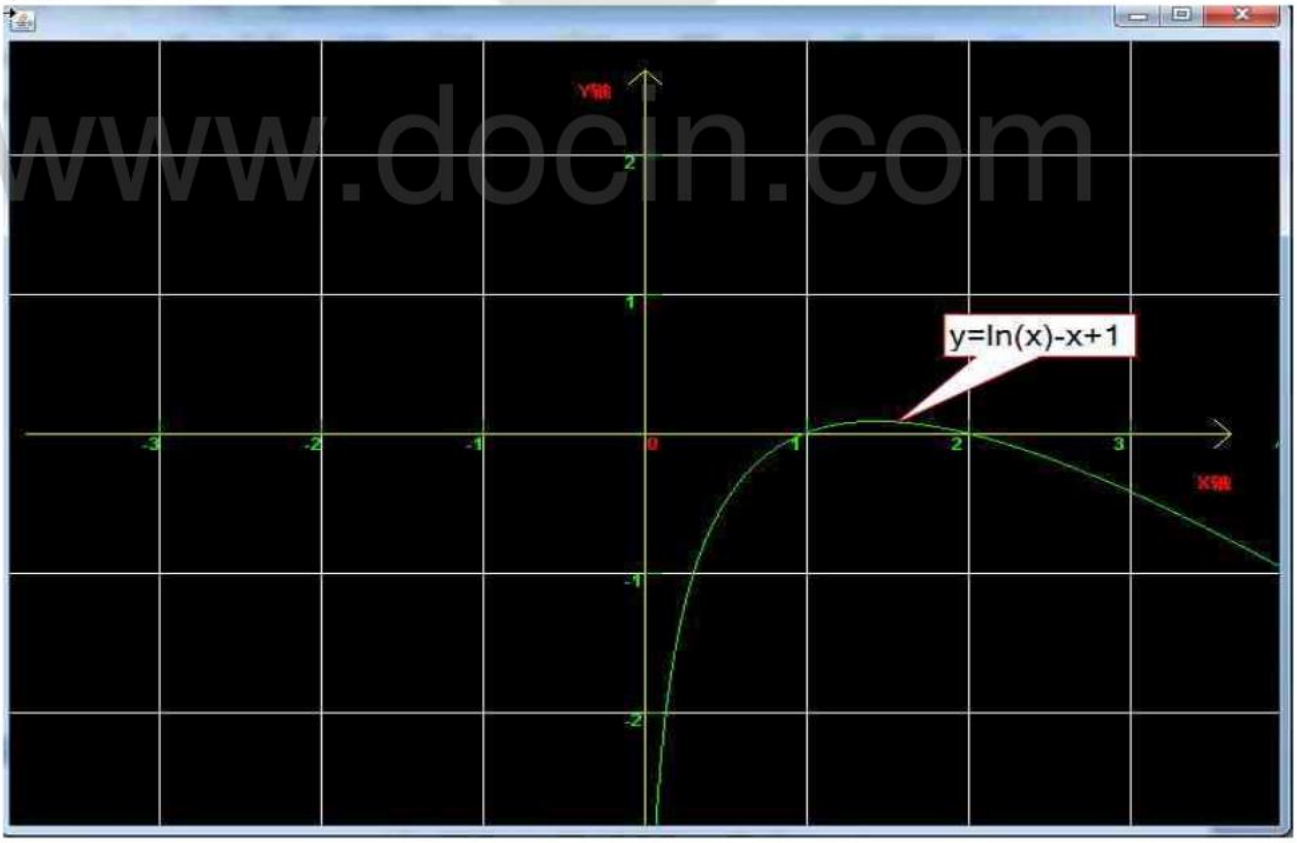
```
<terminated> Jframe [Java Application] C:\Program Files\Java\jre7\bin\javaw.exe (2014-3-6 下午1:31:49)
==========所有的函数如下，请选择需要绘画的函数===========
1、 y=lnx
2、 y=ln(x)-x+1
3、 y=xlnx
4、 y=ln(x)/x
5、 y=H(x)=-xlnx-(1-x)ln(1-x)
6、 D(p||q)(given q)
7、 D(p||q)(given p)
8、 I(X;Y)(given p(y|x))
9、 I(X;Y)(given p(x))

请输入选项：
```

选择绘画的函数式

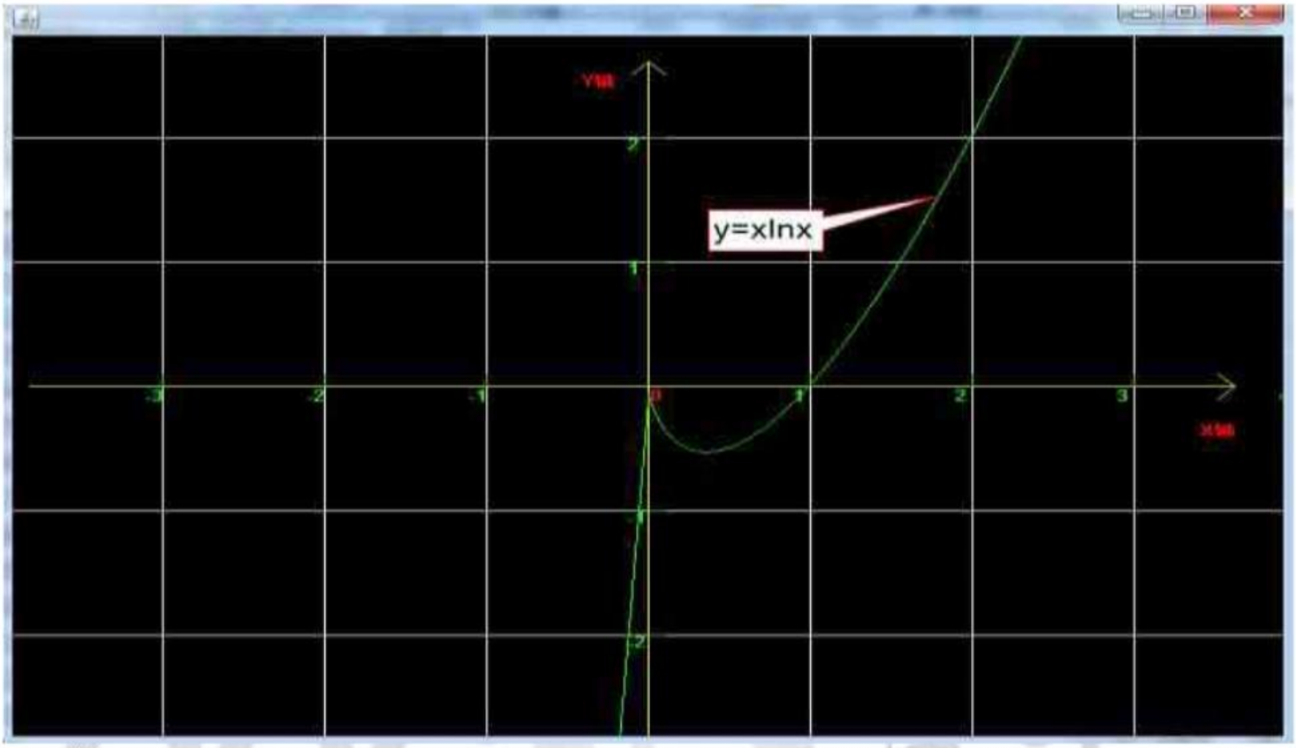
(1) $y=\ln x(H(x)=-\sum_{x\in\chi}p(x)\log p(x))$



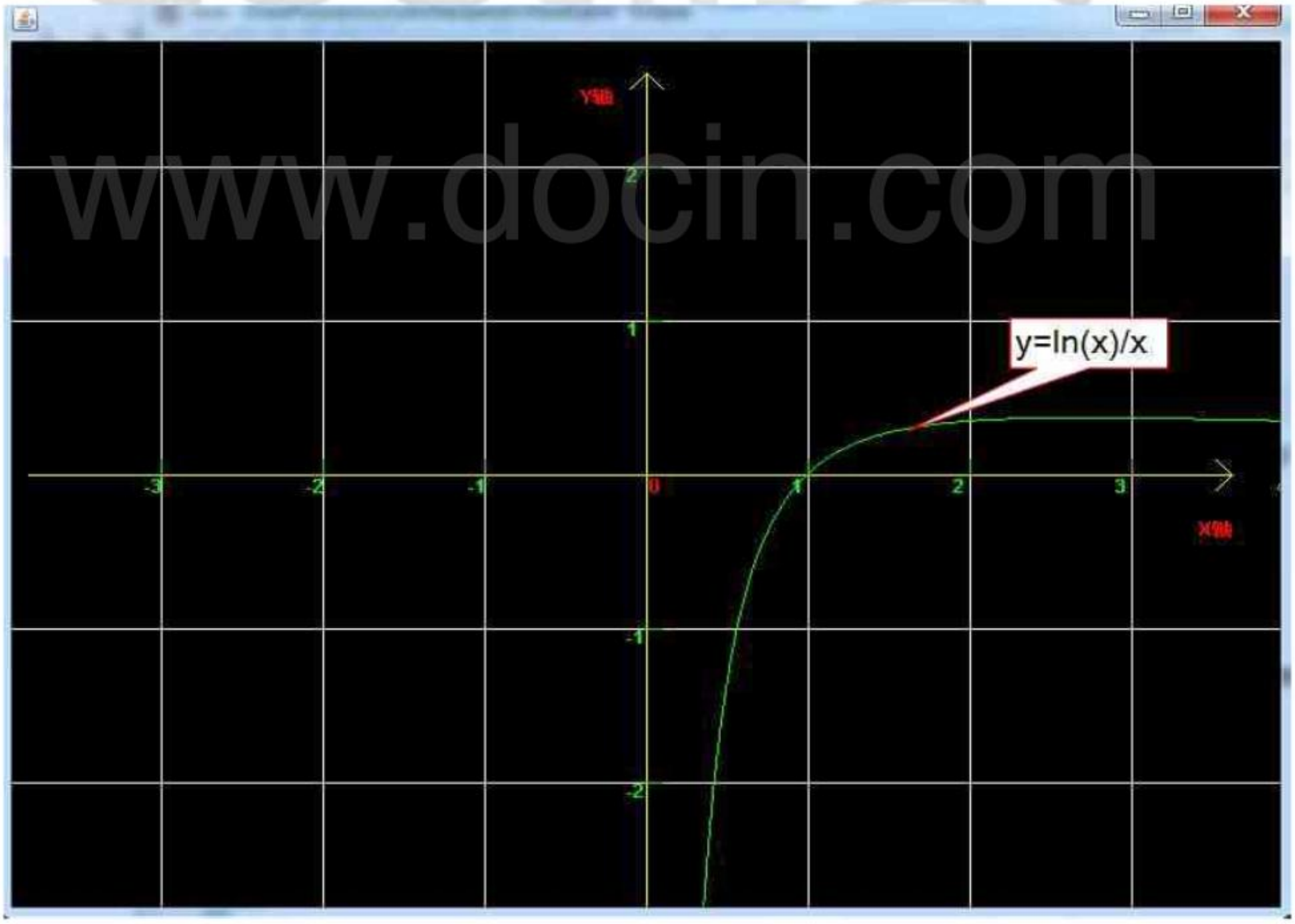
(2) $y = \ln x - x + 1$



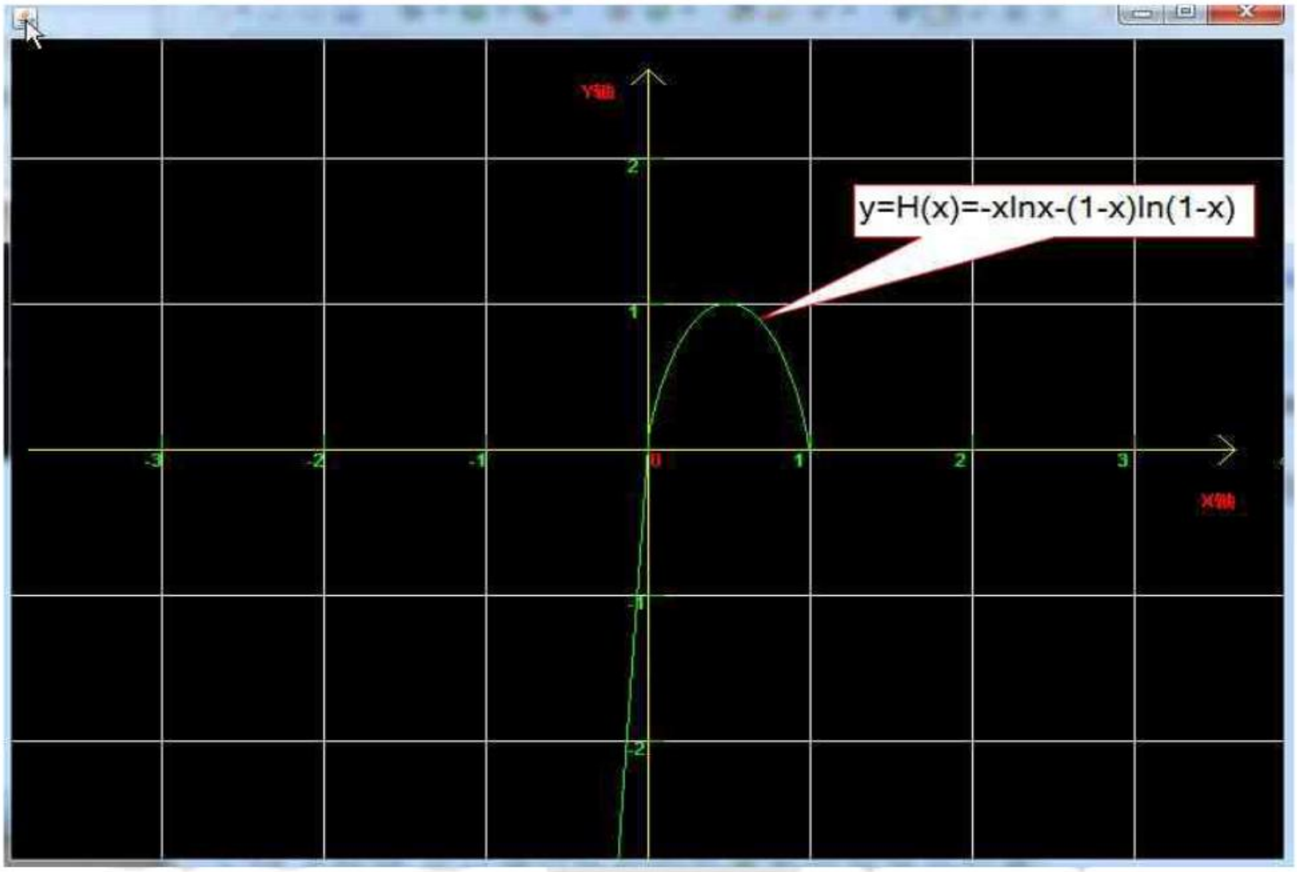
(3) $y = x\ln x$



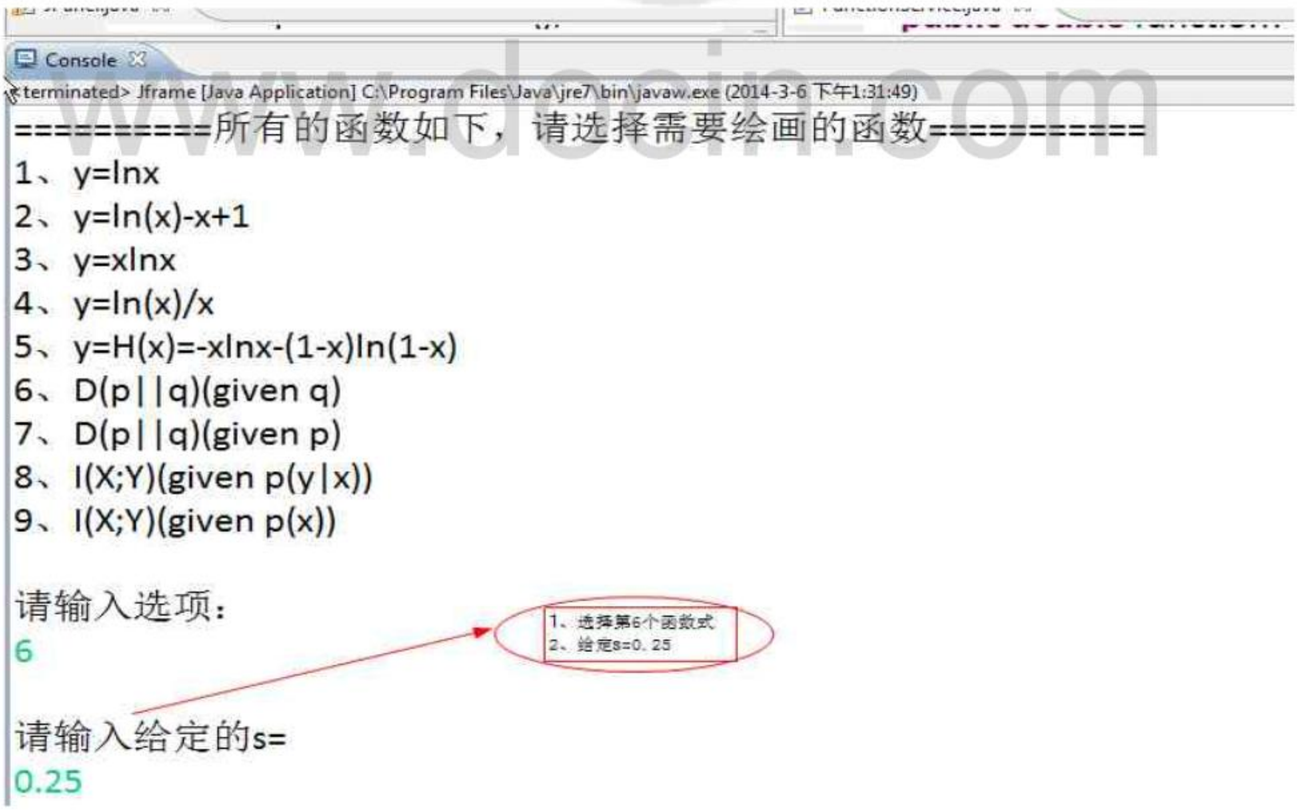
(4) $y = \ln(x)/x$



(5) $y = H(x) = -x\ln x-(1-x)\ln(1-x)$

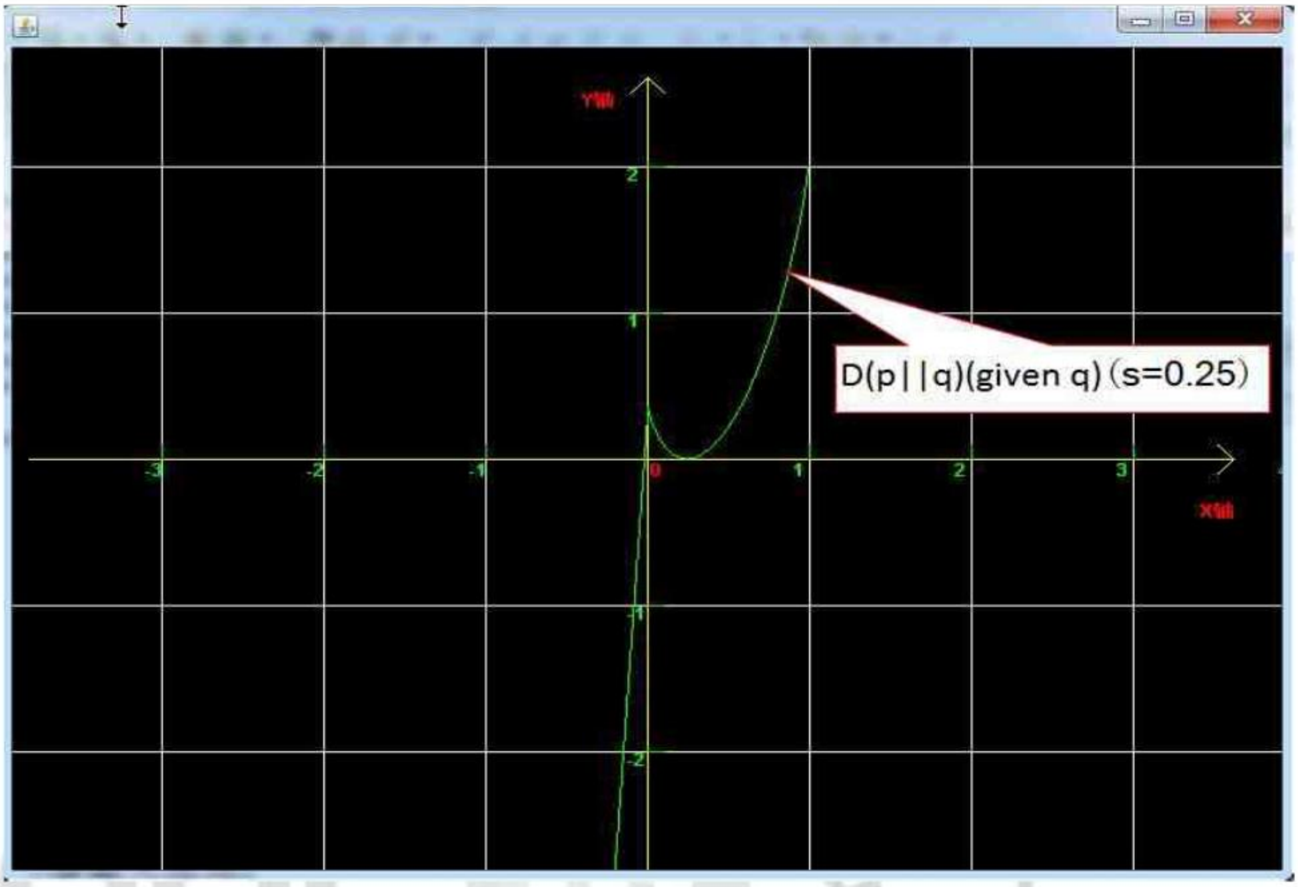


(6) $D(p\,||\,q)(given\quad q)$

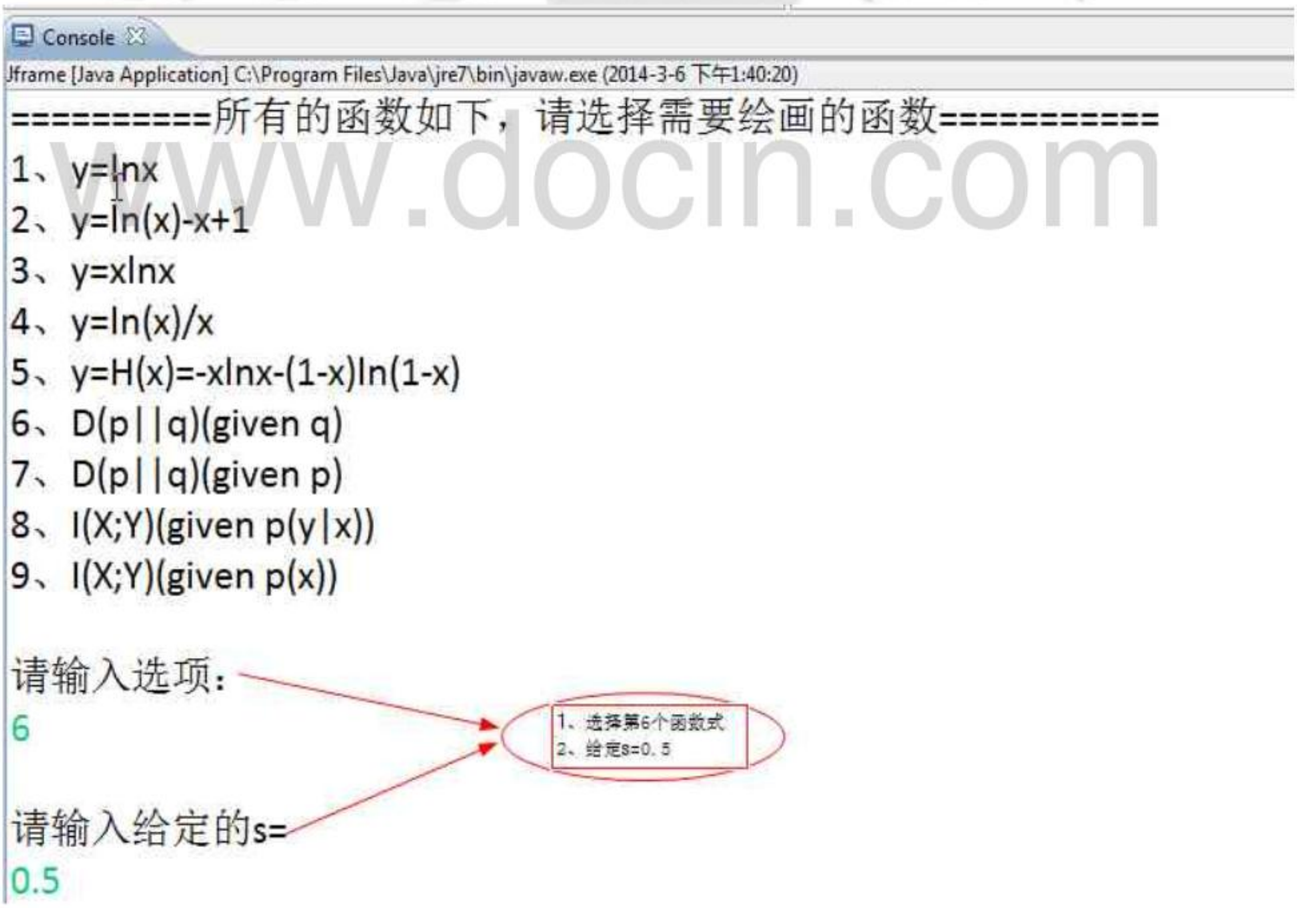


```
<terminated> Jframe [Java Application] C:\Program Files\Java\jre7\bin\javaw.exe (2014-3-6 下午1:31:49)
==========所有的函数如下，请选择需要绘画的函数===========
1、 y=lnx
2、 y=ln(x)-x+1
3、 y=xlnx
4、 y=ln(x)/x
5、 y=H(x)=-xlnx-(1-x)ln(1-x)
6、 D(p||q)(given q)
7、 D(p||q)(given p)
8、 I(X;Y)(given p(y|x))
9、 I(X;Y)(given p(x))

请输入选项:
6

请输入给定的s=
0.25
```

1、选择第6个函数式
2、给定s=0.25

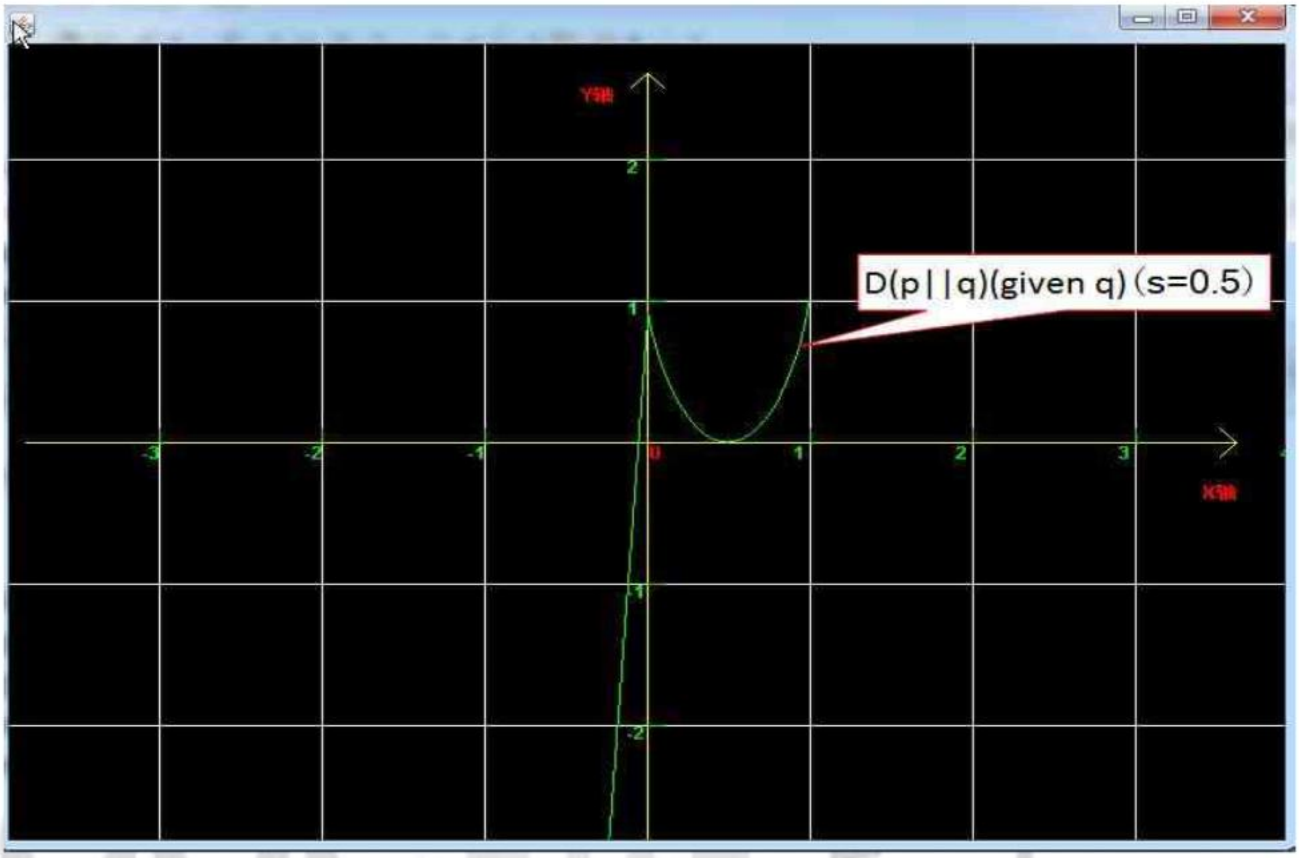


D(p||q)(given q)（s=0.25）

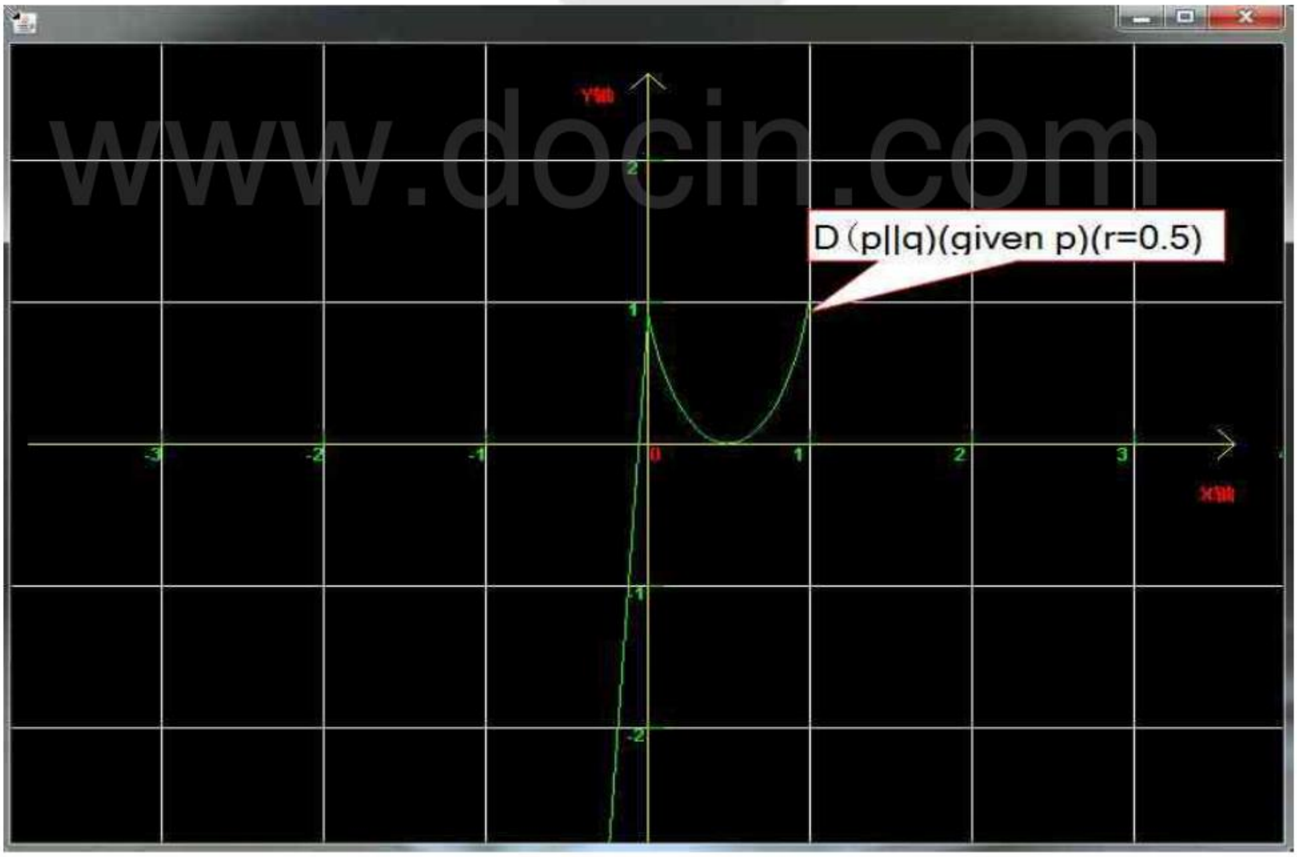


```
Jframe [Java Application] C:\Program Files\Java\jre7\bin\javaw.exe (2014-3-6 下午1:40:20)
==========所有的函数如下，请选择需要绘画的函数==========
1、y=lnx
2、y=ln(x)-x+1
3、y=xlnx
4、y=ln(x)/x
5、y=H(x)=-xlnx-(1-x)ln(1-x)
6、D(p||q)(given q)
7、D(p||q)(given p)
8、I(X;Y)(given p(y|x))
9、I(X;Y)(given p(x))

请输入选项：
6

请输入给定的s=
0.5
```

1、选择第6个函数式
2、给定s=0.5



(7) $D(p\|q)(given\quad p)$



(8) $I(X;Y)(given \quad p(y\mid x))$

(9) $I(X;Y)(given \quad p(x))$

三、使用环境

个人计算机

三、实验环境

个人计算机，MATLAB 平台
对于信息与计算科学专业的学生，可以尝试选择 Java 平台
对于非信息与计算科学专业的学生，可以选择任意编程平台

四、实验记录与实验结果分析

（注意记录实验中遇到的问题。实验报告的评分依据之一是实验记录的细致程度、实验过程的真实性、实验结果的解释和分析。如果涉及实验结果截屏，应选择白底黑字。）

注意的问题：

1、熟记课本的熵计算公式
2、熟悉 swing 图形界面编程
3、界面布局（画线）
4、图形的规范性，包括 x、y 轴坐标、网格绘制
5、对数底的选取（因为 java.math 包里面没有直接以 2 为底的对数必须通过对数的性质转变成以 2 为底的对数函数）

存在的问题：

1、互信息图像目前没有想到更好的解决办法
2、文件数据的读取及相关的运算（分布矩阵）

五、实验体会

（请认真填写自己的真实体会）

六、参考文献

1. （主讲课英文教材）

2. *（如有其它参考文献，请列出）*