

INFO6205 - Assignment01

001565258 Yongyan Deng

Referred from the website:

<https://mathworld.wolfram.com/RandomWalk2-Dimensional.html#:~:text=Amazingly%2C%20it%20has%20been%20proven,number%20of%20steps%20approaches%20infinity>,

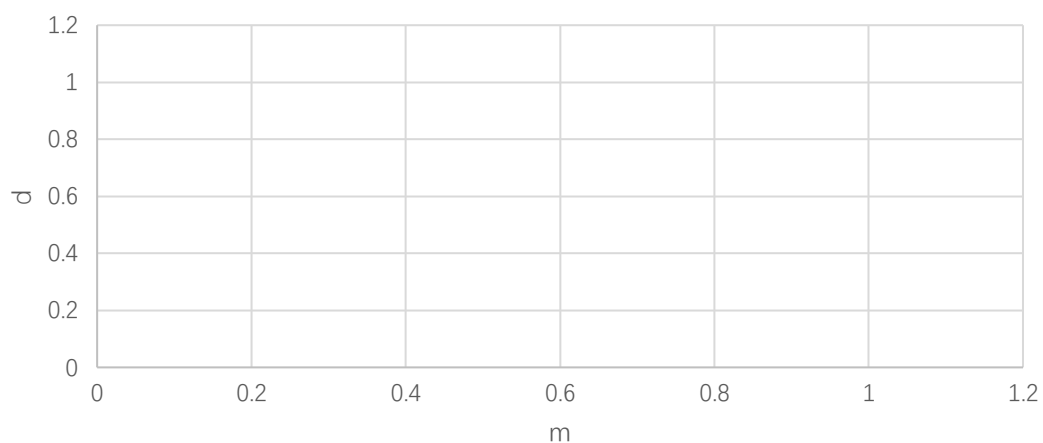
we can know that the relationship between the number of steps m and the distance d is:

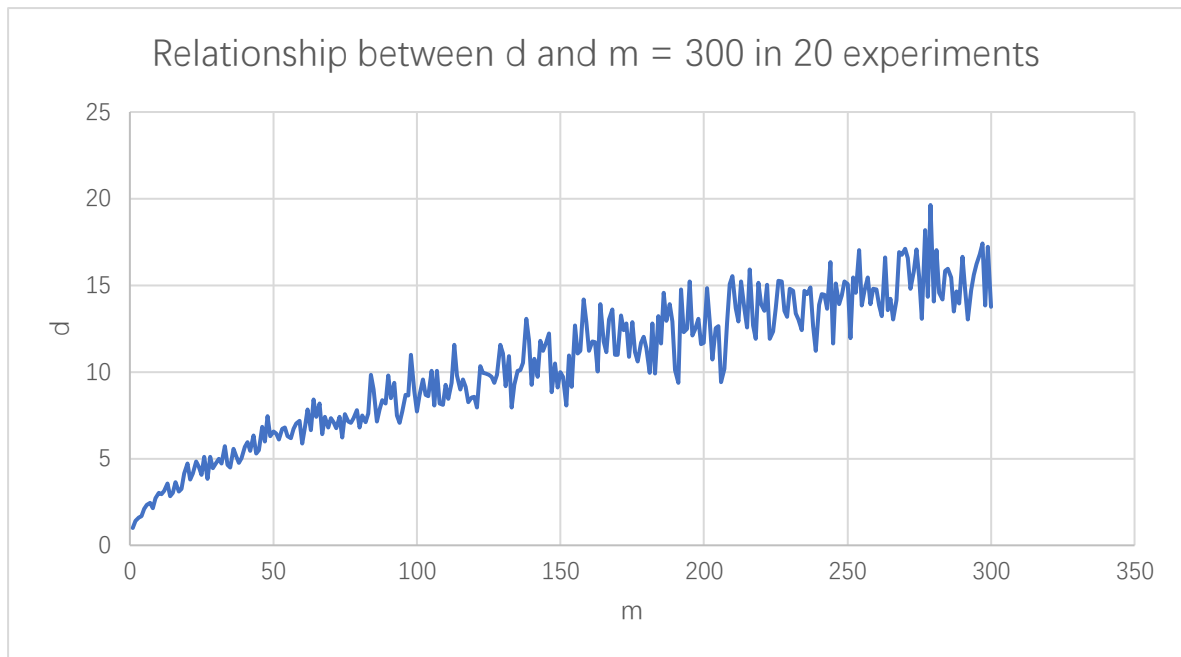
$$d = \sqrt{m}$$

I implement 20 times experiments with 200 steps and 300 steps, and then output the distance corresponded to each number of steps into a line chart.

```
public static void main(String[] args) {  
    if (args.length == 0)  
        throw new RuntimeException("Syntax: RandomWalk steps [experiments]");  
    int m = Integer.parseInt(args[0]);  
    int n = 20, m = -1;  
    for(int i = 1; i <= 200; i++) {  
        m = i;  
        if (args.length > 1) n = Integer.parseInt(args[1]);  
        double meanDistance = randomWalkMulti(m, n);  
        System.out.println(m + " steps: " + meanDistance + " over " + n + " experiments");  
    }  
}
```

Relationship between d and $m = 200$ in 20 experiments





It can be found from the chart that the relationship between d and m is relatively close to the relationship shown from the above formula.

Here is the screen shot for passing all unit tests:

```
1  /*
2   * Copyright (c) 2017, Phasmid Software
3   */
4
5  package edu.neu.coe.info6205.randomwalk;
6
7  import edu.neu.coe.info6205.util.PrivateMethodTester;
8  import org.junit.Test;
9
10 import static org.junit.Assert.assertEquals;
11 import static org.junit.Assert.assertNotSame;
12
13 public class RandomWalkTest {
14
15     @Test
16     public void testMove0() {
17         RandomWalk rw = new RandomWalk();
18         PrivateMethodTester pmt = new PrivateMethodTester(rw);
19         pmt.invokePrivate("move", 1, 0);
20         assertEquals("expected: 1.0, rw.distance()", delta: 1.0E-7);
21     }
22
23     /**
24      *
25      */
26 }
```

Run: RandomWalkTest

Tests passed: 6 of 6 tests - 210 ms

Test	Time
RandomWalkTest (edu.neu.coe.info6205)	4 ms
testRandomWalk2	20 ms
testMove0	1 ms
testMove2	1 ms
testMove3	1 ms
testRandomWalk	183 ms

Process finished with exit code 0