

计算物理B第一次作业

PB20511896 王金鑫

使用说明

文件说明

- `main.py`: 主程序, 包括对2D和3D随机数点的产生和图像绘制。
- `Func.py`: 函数定义文件, 用于定义基于线性同余法的随机数产生器, 3D、2D散点图绘图函数和三维均匀性检验函数。

库说明

- `matplotlib`、`matplotlib.pyplot`、`numpy`: 用于绘制2D和3D图像

主要变量说明

重要的变量在源代码中均有注释, 在这里挑选部分进行展示

- `x_0`: 随机数产生种子 (迭代初值)
- `a`、`c`、`m`: 随机数产生器参数, 对应 $x_{n+1} = (ax_n + c) \bmod m$
- `N1`、`N2`: 产生的随机数点的数目 (3D对应3N个随机数)
- `ax3.scatter3D(s=)`: 绘制的点的大小
- `ax3.scatter3D(alpha=)`: 绘制的点的透明度
- `ax3.scatter3D(c=)`: 绘制的点的颜色

2D同理

运行说明

打开`main.py`, 运行, 得到3D和2D图像。

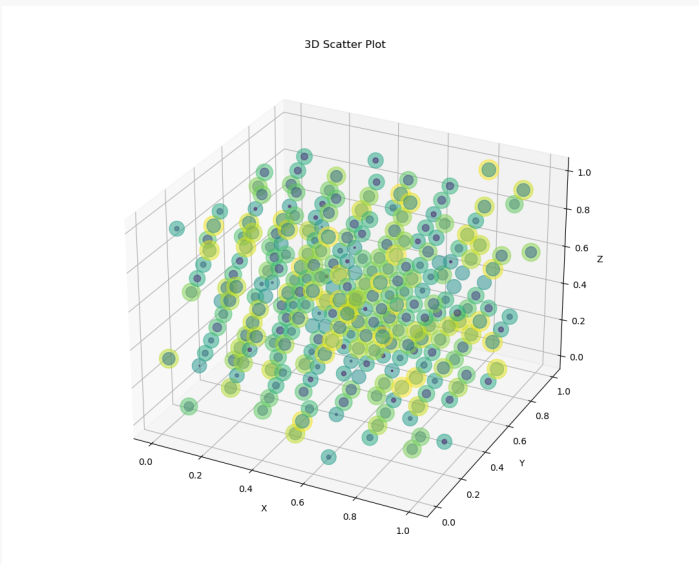
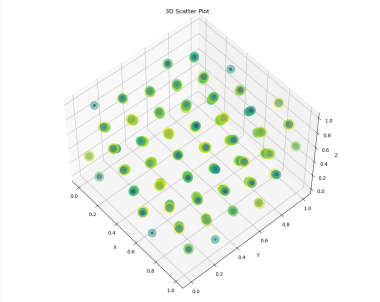
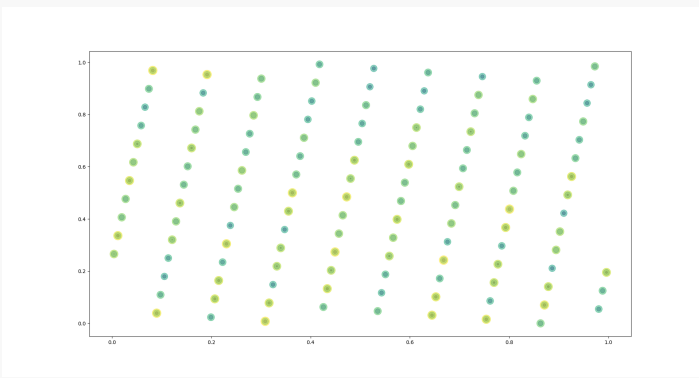
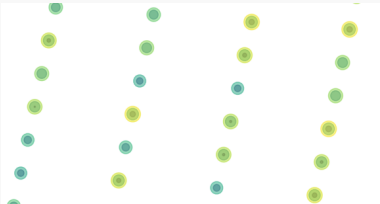
在`Func.py`中修改两个`RandomMakerLC`函数里参数的值来改变RNG的参数

在`main.py`中修改`x_0`的值来改变RNG的种子值; 修改`N1`、`N2`来改变产生的点的数目; 修改`k`的值来修改对空间划分的个数($k \times k \times k$)。

注意事项

- 随机数产生器返回的是列表, 需要用`numpy.array()`转化为数组
- 在绘制散点图时我将不同的点按照绘制的次序赋予不同的颜色和大小, 同时将点的透明度 (0~1.0) 调整为0.5, 使得可以在图像当中较为明显地观察到点的重合现象。
- 均匀性检验函数会返回`N`, `k`和 χ^2 的值。其中 χ^2 服从 $\chi^2((k-1)^3)$ 分布。

运行结果

	
<p>图1. 3D散点图</p>	<p>图2. 在某个特定方向下的3D散点图</p>
	
<p>图3. 2D散点图</p>	<p>图4. 局部放大后的的2D散点图</p>
<pre>for 3D N = 500 k = 5 chi-square = 74.0 Obey the distribution of chi-square(64)</pre>	
<p>图5. 三维均匀性检验结果</p>	

由图2可看出，该伪随机数产生的结果独立性较差。

查表可得自由度为64时，显著性水平 $\alpha = 0.25$ 对应的值为71.225， $\alpha = 0.1$ 对应的值为78.86。故在0.01的显著性水平下，我们不能拒绝该伪随机数满足均匀性的要求的假设。