遗传算法的 Python 实现
Genetic Algorithm in Python

杨元辉 学号：5090209074*
Yuanhui Yang Student ID: 5090209074

上海交通大学
Shanghai Jiaotong University

I. 遗传算法简介

遗传算法是一门边缘交叉学科，是生物、数学、计算机理论等多个领域的完美结合，是生物进化淘汰在二进制计算机领域的出色实现。

1.1 生物学基础

生物世界充满了淘汰、竞争、突变等进化活动。以自然选择学说为核心的现代生物进化理论，其基本观点是：

- 种群是生物进化的基本单位，
- 生物进化的实质是种群基因频率的改变，
- 具体实现是：
  - 基因突变：产生生物进化的原材料 I；
  - 基因重组：产生生物进化的原材料 II；
  - 自然选择：使种群的基因频率定向改变并决定生物进化的方向；
  - 生殖隔离：新物种形成的必要条件。

相应地，在计算机领域，上述过程的计算机实现，就是遗传算法的具体实现。

*杨元辉，学号 5090209074，上海交通大学，机械与动力工程学院，机械电子工程

II. 计算机实现

2.1 编码

遗传算法的编码规则有二进制编码与浮点数编码两种，在下面的算例中使用的是二进制编码。这样操作的优势是：二进制编码及符合计算机处理信息的原理，也方便对染色体进行遗传、编译和突变等操作，因为二进制数组相对应十进制数组更容易实现简并。

$$\delta = \frac{U-L}{2^k-1} \tag{1}$$

2.2 解码

将二进制数组转变为十进制数组的过程就称为解码过程。

$$x = L + \left(\sum_{i=1}^{k} b_k 2^{i-1}\right) \cdot \frac{U-L}{2^k-1} \tag{2}$$

2.3 交配

1. 产生一个 0,1 之间的随机数，若该数小于 0.1(交配概率)，则进行以下操作；
2. 用随机数 (整数) 产生一个交配位点；
3. 将两个二进制数组在以上交配位点之后的所有值互换。

2.4 突变

1. 产生一个 0,1 之间的随机数，若该数小于 0.4(突变概率)，则进行以下操作；
2. 用随机数 (整数) 产生一个突变位点；
3. 将二进制数组上的突变位点 0,1 互换，具体实现是：

$$value = abs\left(value - 1\right) \tag{3}$$

2.5 自然选择

模拟自然选择法则的过程，我们使用轮盘选择法，即以高概率选中结果更加出色的解，作为下一轮计算的样本。

1. 解码；
2. 计算对应的目标函数值；

3. 将以上函数值归一化到 $(0,1)$ 区间，同时加和为 1，即频率；
4. 计算以上归一化结果的积累频率；
5. 产生一个 $(0,1)$ 之间的随机数，将该结果对应的二进制数组提出，这就是我们选出来的相对最优解；
6. 将该结果代入下一轮循环，如此下去，结果相对佳的解便在解集中扩散开去，结果不佳的解会被慢慢地被淘汰。

以上过程是随机的，充满概率的，按照解出色的程度进行概率抽取，也给其他并不是最出色的解机会，这样做是为了防止陷入局部最优解。

III. 算例

$$\begin{aligned} &\min\left\{\exp\times(x)\left(4x^2+2y^2+4xy+2y+1\right)\right\} \\ &\text{subject to}:\begin{cases}1.5+xy-x-y\leq 0\\ -xy\leq 10\end{cases}\end{aligned} \tag{4}$$

3.1 具体实现

3.1.1 预处理

遗传算法之中有使用概率，概率非负。因此，需要做变换：

$$\begin{cases} z\leftarrow z+\text{Intial Value}\\ z>0\end{cases} \tag{5}$$

遗传算法是求取全局最大值的，而式 (4) 是求取最小值的。因此，需要做变换：

$$z\leftarrow\frac{1}{z} \tag{6}$$

3.1.2 边界处理

非常明显，经过预处理之后，以上问题的对应的函数值一定大于 0，因此，我们只需要将超过边界部分的解对应的函数值强行设定成 0。这样的话，在“自然选择”这步，0 就会被屏蔽掉，对应的解便会被淘汰掉。

$$\text{When}:\begin{cases}1.5+xy-x-y>0\\ -xy>10\end{cases}\qquad \text{then}:z\leftarrow 0 \tag{7}$$

3.1.3 实验结果

经过 500 次训练之后，我们得到该 NP 问题可接受的全局最优解，即最后一个训练结果 `VarMax[-1]`：

$$[x, y, f(x, y)] = [-6.3045, 1.5527, 0.2353] \tag{8}$$

另一方面，我们对上述解作边界检测，看一看是否满足边界条件：

表 1: 边界检测

计算项目	$1.5 + xy - x - y \leq 0$	$-xy \leq 10$
计算结果	-3.5377	9.7895
满足与否	是	是

使用 MATLAB 作图，我们能够看到程序输出的平均值与最大值的变化趋势是越来越接近，这说明训练样本在“进化”。

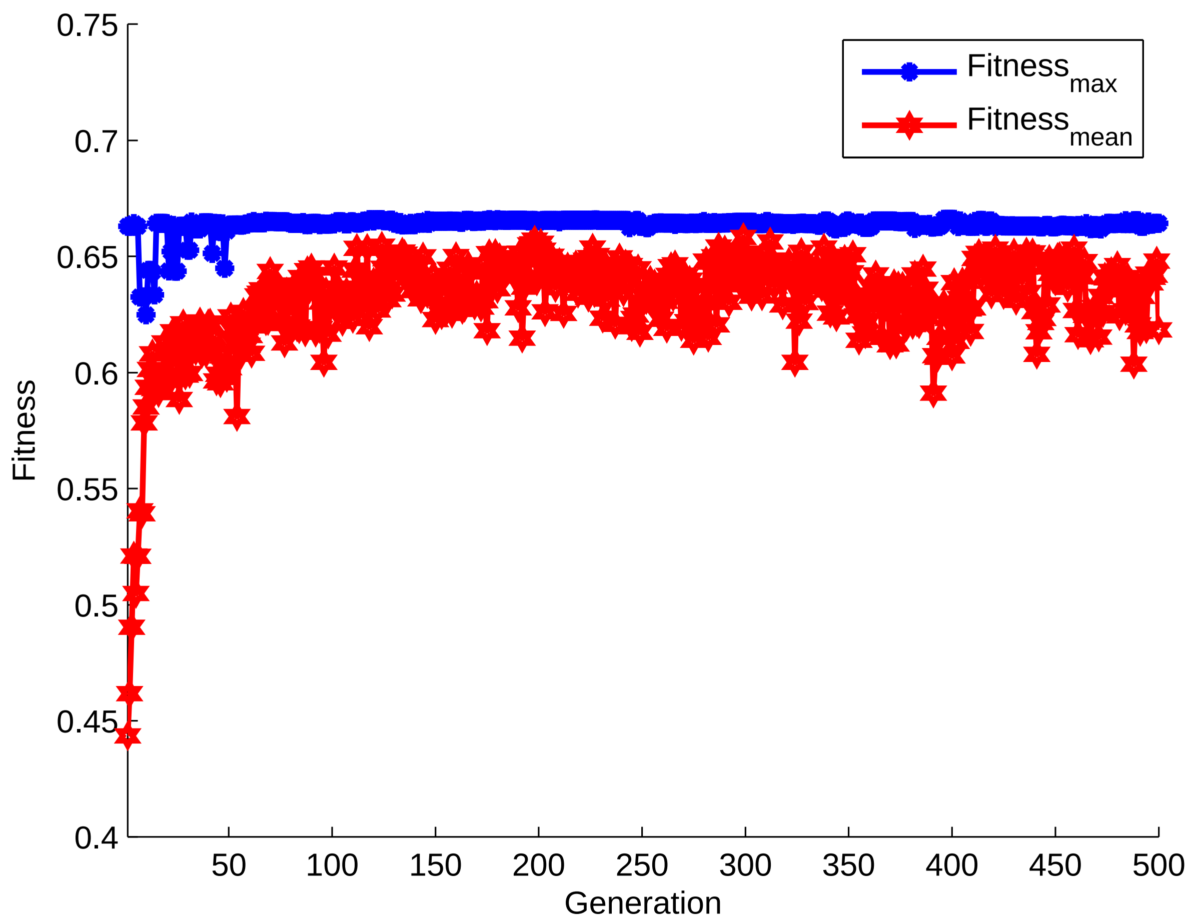

图 1: 平均值与最大值

IV. 总结

经过遗传算法的求解，我们成功地解决了该非线性回归问题，具体的结果满足边界调节，求解的结果也可以接受。

足以见到，遗传算法的结果是一种可以信赖的算法，它基于对之前求解过程的学习。相比穷举算法，遗传算法在解决 NP 问题时有着独特的优势。

V. 鸣谢

最后，感谢陆老师在程序尾声为我提供的帮助，老师面向对象以及解决问题的视角对产生了积极的影响。在老师的 ftp 中我看到了数据库等资料，这是我进一步学习的重要参考。

之后我即将从机械行业转到计算机科学，我选择北京微软亚洲研究院作为我职业生涯的起点，希望有朝一日能够成为一位出色的计算机科学学家。由衷地感谢老师。

全文使用 LaTeX 编译完成，一方面说明我对 cs 的浓厚兴趣，一方面也说明我的职业或学术选择。