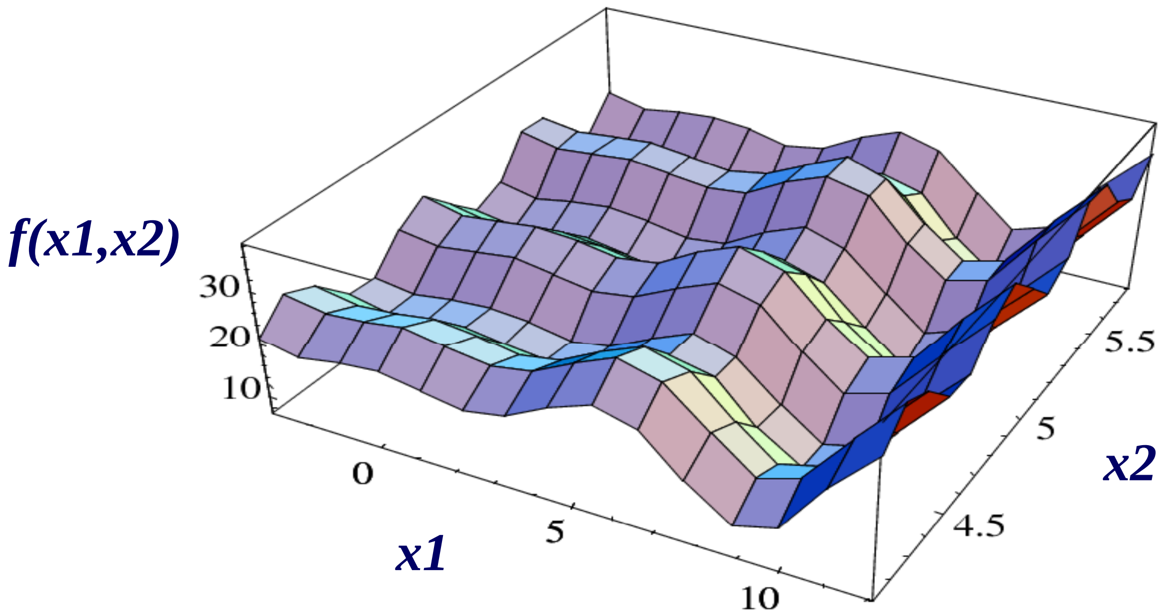
遗传算法——简单实例

- 对如下无约束优化问题：

$$\begin{aligned} \max f(x_1, x_2) &= 21.5 + x_1 \sin(4\pi x_1) + x_2 \sin(20\pi x_2), \\ & -3.0 \le x_1 \le 12.1, \\ & 4.1 \le x_2 \le 5.8. \end{aligned}$$

- 三维图形如图所示：



遗传算法—简单实例

- 基因表达：
  - 应用GA的首要问题，就是进行染色体编码，基本GA的染色体编码为二进制编码，
  - 所以，我们首先需要将实数表示的问题的解，编码为二进制串。
    - 问题的解： $[x_1,\ x_2]$
    - 染色体： $[\overbrace{01\ldots\ldots0110}^{x_1}\overbrace{0010\ldots\ldots10}^{x_2}]$

      m_1位　　m_2位

遗传算法——简单实例

- 二进制串的长度确定：
  - 串的长度取决于需要的精度，例如 x_j 的值域是 $[a_j,b_j]$，要求精度是小数点后6位，这就要求 x_j 的值域至少要分成 $(b_j-a_j)*10^6+1$份，设x_j 所需要的子串长为 m_j，可由如下公式计算：

$$2^{m_j-1} < (b_j - a_j) \times 10^6 \leq 2^{m_j} - 1,$$

- 二进制串到实数值的转换
  - 将二进制串转换为实数值，可由如下公式计算：

$$x_j = a_j + decimal(substring_j) \times \frac{b_j - a_j}{2^{m_j} - 1}$$

$decimal(substring_j)$表示变量 x_j 的子串 $substring_j$ 的十进制值。

- 编码
  - 假设 x_1 ，x_2 需要的精度都是小数点后4位，需要的总串长度按下式计算：

$$(12.1-(-3.0))\times 10000 = 151000$$
$$2^{17} < 151000 \le 2^{18}-1, \qquad m_1 = 18$$
$$(5.8-4.1)\times 10000 = 17000$$
$$2^{14} < 17000 \le 2^{15}-1, \qquad m_2 = 15$$
$$m = 18+15 = 33$$

  - 染色体的总长度为33位，可表示为：

$$\overbrace{v_j \quad \underbrace{000001010100101001}_{18\text{ bits}}\ \underbrace{101111011111111}_{15\text{ bits}}}^{33\text{ bits}}$$

遗传算法—简单实例

$$\overset{\longleftarrow\ 33\ \text{bits}\ \longrightarrow}{v_j\quad \underbrace{000001010100101001}_{18\ \text{bits}}\ \underbrace{101111011111111}_{15\ \text{bits}}}$$

- 对应的 x_1，x_2 变量的值为：

	binary number	decimal number
x_1	000001010100101001	5417
x_2	101111011111110	24318

- 实数值为：

$$x_1 = -3.0 + 5417 \times (12.1 - (-3.0))/(2^{18} - 1) = -2.687969$$

$$x_2 = 4.1 + 24318 \times (5.8 - 4.1)/(2^{15} - 1) = 5.361653$$

- 初始种群：
  - 随机产生初始种群（假设种群大小为10）， 对应十进制值

$$
\begin{aligned}
\boldsymbol{v}_1 &= (000001010100101001101111011111110) & \boldsymbol{v}_1 &= (x_1, x_2) = (-2.687969, 5.361653) \\
\boldsymbol{v}_2 &= (001110101110011000000010101001000) & \boldsymbol{v}_2 &= (x_1, x_2) = (\ \ 0.474101, 4.170144) \\
\boldsymbol{v}_3 &= (111000111000001000010101001000110) & \boldsymbol{v}_3 &= (x_1, x_2) = (\ 10.419457, 4.661461) \\
\boldsymbol{v}_4 &= (100110110100101101000000010111001) & \boldsymbol{v}_4 &= (x_1, x_2) = (\ \ 6.159951, 4.109598) \\
\boldsymbol{v}_5 &= (000010111101100010001110001101000) & \boldsymbol{v}_5 &= (x_1, x_2) = (-2.301286, 4.477282) \\
\boldsymbol{v}_6 &= (111110101011011000000010110011001) & \boldsymbol{v}_6 &= (x_1, x_2) = (\ 11.788084, 4.174346) \\
\boldsymbol{v}_7 &= (110100010011111000100110011101101) & \boldsymbol{v}_7 &= (x_1, x_2) = (\ \ 9.342067, 5.121702) \\
\boldsymbol{v}_8 &= (001011010100001100010110011001100) & \boldsymbol{v}_8 &= (x_1, x_2) = (-0.330256, 4.694977) \\
\boldsymbol{v}_9 &= (111110001011101100011101000111101) & \boldsymbol{v}_9 &= (x_1, x_2) = (\ 11.671267, 4.873501) \\
\boldsymbol{v}_{10} &= (111101001110101010000010101101010) & \boldsymbol{v}_{10} &= (x_1, x_2) = (\ 11.446273, 4.171908)
\end{aligned}
$$

遗传算法——简单实例

- 交叉：
  - 设交叉率 p_c = **0.25**，即平均有**25%**的染色体进行交叉，对每个染色体产生一个 **[0,1]** 间的随机数，小于交叉率的则选作交叉对象。
  - 设随机数序列如下：

0.625721	0.266823	0.288644	0.295114	$\boxed{0.163274}$
0.567461	$\boxed{0.085940}$	0.392865	0.770714	$\boxed{0.148656}$

v_5 and v_7 被选入参与交叉

  - 在 **[1,32]** 之间产生一个随机位置（因染色体长为**33**），假设为**7**，则在**7**处，将染色体切断，并交换断点的右端。

$$v_5 = (0000101\,|\,11101100010001110001101000)$$

$$v_7 = (1101000\,|\,10011111000100110011101101)$$

$$v_5' = (0000101\,|\,10011111000100110011101101)$$

$$v_7' = (1101000\,|\,11101100010001110001101000)$$

遗传算法——简单实例

- 变异：
  - 设变异率 p_m = **0.01**，即种群中平均有**1%**的基因发生变异。每个染色体基因数：**33**，种群中染色体个数：**10**，总的基因：**330**，平均有**3.3**个基因发生变异。对每个基因产生一个 **[0,1]** 间的随机数，小于变异率的，选作变异基因 。
  - 假设选择结果导致如下基因发生变异：

位置	染色体号	基因序号	随机数
105	4	6	0.009857
164	5	32	0.003113
199	7	1	0.000946
329	10	32	0.001282

遗传算法——简单实例

- 变异：
  - 染色体变异如下：

$$
\begin{aligned}
\boldsymbol{v}_4 &= (100110110100101101000000010111001)\\
\boldsymbol{v}_4'' &= (100111110100101101000000010111001)\\
\boldsymbol{v}_5 &= (000010111101100010001110001101000)\\
\boldsymbol{v}_5'' &= (000010111101100010001110001101010)\\
\boldsymbol{v}_7 &= (110100010011111000100110011101101)\\
\boldsymbol{v}_7'' &= (010100010011111000100110011101101)\\
\boldsymbol{v}_{10} &= (111101001110101010000010101101010)\\
\boldsymbol{v}_{10}'' &= (111101001110101010000010101101000)
\end{aligned}
$$

- 评估：计算染色体的适值。
  ① 将二进制的位串（染色体）解码为实数值；
  ② 将目标函数值映射为适值
    - 对于求极大，可简单地取目标函数值为适值。
    - 适值函数起测定染色体对目标的适应性的作用。

遗传算法——简单实例

➢ 染色体解码：

$$v_5' = (000010110011111000100110011101101) \quad v_5' = (x_1, x_2) = (-2.33688, 5.121702)$$

$$v_7' = (110100011101100010001110001101000) \quad v_7' = (x_1, x_2) = (9.377665, 4.477282)$$

$$v_4'' = (100111110100101101000000010111001) \quad v_4'' = (x_1, x_2) = (6.39589, 4.109598)$$

$$v_5'' = (000010111101100010001110001101010) \quad v_5'' = (x_1, x_2) = (-2.30129, 4.477386)$$

$$v_7'' = (010100010011111000100110011101101) \quad v_7'' = (x_1, x_2) = (1.792038, 5.121702)$$

$$v_{10}'' = (111101001110101010000010101101000) \quad v_{10}'' = (x_1, x_2) = (11.44627, 4.171804)$$

遗传算法—简单实例

- 计算染色体的适值。

$$
\begin{aligned}
eval(\boldsymbol{v}_1) &= f(-2.687969, 5.361653) = 19.805119 \\
eval(\boldsymbol{v}_2) &= f(0.474101, 4.170144) = 17.370896 \\
eval(\boldsymbol{v}_3) &= f(10.419457, 4.661461) = 9.590546 \\
eval(\boldsymbol{v}_4) &= f(6.159951, 4.109598) = 29.406122 \\
eval(\boldsymbol{v}_5) &= f(-2.301286, 4.477282) = 15.686091 \\
eval(\boldsymbol{v}_6) &= f(11.788084, 4.174346) = 11.900541 \\
eval(\boldsymbol{v}_7) &= f(9.342067, 5.121702) = 17.958717 \\
eval(\boldsymbol{v}_8) &= f(-0.330256, 4.694977) = 19.763190 \\
eval(\boldsymbol{v}_9) &= f(11.671267, 4.873501) = 26.401669 \\
eval(\boldsymbol{v}_{10}) &= f(11.446273, 4.171908) = 10.252480
\end{aligned}
$$

$$eval(v_5') = f(-2.33688, 5.121702) = 24.43828$$

$$eval(v_7') = f(9.377665, 4.477282) = 7.696252$$

$$eval(v_4'') = f(6.39589, 4.109598) = 17.65403$$

$$eval(v_5'') = f(-2.30129, 4.477386) = 15.69017$$

$$eval(v_7'') = f(1.792038, 5.121702) = 25.60888$$

$$eval(v_{10}'') = f(11.44627, 4.171804) = 10.25756$$

遗传算法——简单实例

- 选择：
  - 采用轮盘赌选择。它是一种正比选择策略，能够根据与适值成正比的概率选出新的种群。

① 对各个染色体计算适值

$$eval(\boldsymbol{v}_k) = f(\boldsymbol{x}), k = 1, 2, \ldots, K$$ （包括交叉变异后的所有染色体）

② 计算种群中所有染色体的适值之和

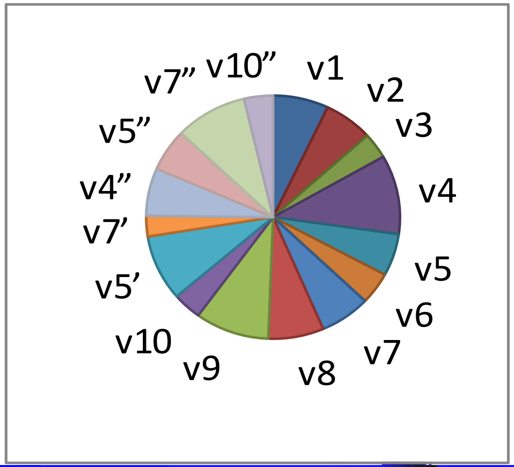
$$F = \sum_{k=1}^{K} eval(\boldsymbol{v}_k)$$

③ 对各染色体计算选择概率（在转盘上占多大的面积）

$$p_k = \frac{eval(\boldsymbol{v}_k)}{F}, \quad k = 1, 2, \ldots, K$$

④ 对各染色体计算累积概率（在转盘上的位置）

$$q_k = \sum_{j=1}^{k} p_j, \quad k = 1, 2, \ldots, K$$



遗传算法——简单实例

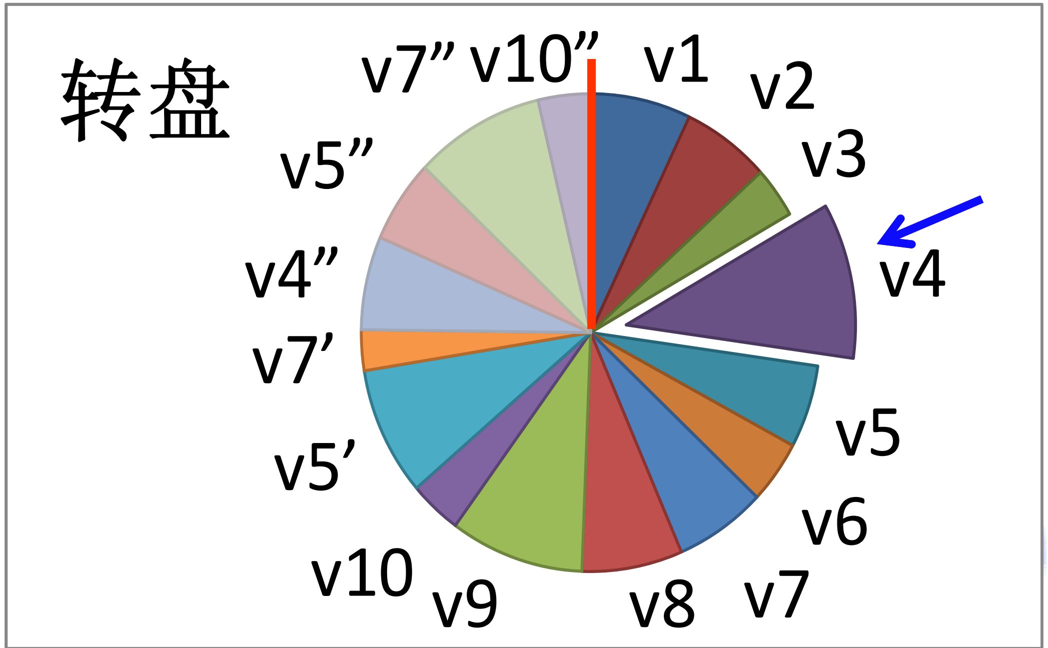
- 选择：
  - 选择过程如下：

  ① 在 **[0,1]** 之间，产生一个均匀分布的伪随机数 ***r***；

  ② 若下述关系成立，则选择第 ***k*** 个染色体。

$$q_{k-1} < r \le q_k,\ q_o = 0,\ (1 \le k \le K)$$

伪随机数表示指针
大小表示位置
所指向的染色体就是
待选择的染色体



遗传算法——简单实例

- 针对例题，首先计算适值之和

$$F = \sum_{k=1}^{16} eval(v_k) = 279.4805$$

- 计算各染色体选择概率、累积概率

No	eval	p	q
v1	19.80512	0.070864	0.070864
v2	17.3709	0.062154	0.133018
v3	9.590546	0.034316	0.167334
v4	29.40612	0.105217	0.272551
v5	15.68609	0.056126	0.328677
v6	11.90054	0.042581	0.371258
v7	17.95872	0.064257	0.435515
v8	19.76319	0.070714	0.506229
v9	26.40167	0.094467	0.600696
v10	10.25248	0.036684	0.63738
v5’	24.43828	0.087442	0.724822
v7’	7.696252	0.027538	0.75236
v4’	17.65403	0.063167	0.815527
v5”	15.69017	0.05614	0.871668
v7”	25.60888	0.09163	0.963298
v10’	10.25756	0.036702	1

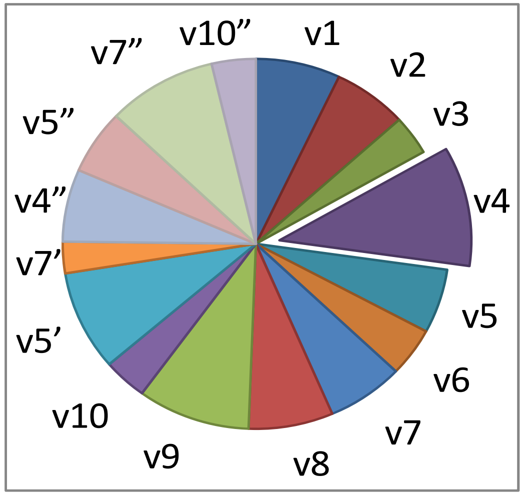
遗传算法——简单实例

- 旋转转轮10次，每次选择一个染色体来构造新种群

0.301431	0.322062	0.766503	0.881893
0.350871	0.583392	0.177618	0.343242
0.032685	0.197577		

- 最后选出的种群如下：

$$v'_1 = (000010111101100010001110001101000)\ (v_5)$$
$$v'_2 = (000010111101100010001110001101000)\ (v_5)$$
$$v'_3 = (100111110100101101000000010111001)\ (v''_4)$$
$$v'_4 = (110100011101100010001110001101000)\ (v'_7)$$
$$v'_5 = (111110101011011000000010110011001)\ (v_6)$$
$$v'_6 = (111110001011101100011101000111101)\ (v_9)$$
$$v'_7 = (100110110100101101000000010111001)\ (v_4)$$
$$v'_8 = (111110101011011000000010110011001)\ (v_6)$$
$$v'_9 = (000001010100101001101111011111110)\ (v_1)$$
$$v'_{10} = (100110110100101101000000010111001)\ (v_4)$$



至此，完成遗传算法的一次迭代。

遗传算法——简单实例

- 如此反复运行，**1000**代后结束，在**419**代得到最佳染色体：

$$\begin{aligned}
\boldsymbol{v}^* &= (111110000000111000111101001010110)\\
eval(\boldsymbol{v}^*) &= f(11.631407, 5.724824) = 38.818208\\
x_1^* &= 11.631407\\
x_2^* &= 5.724824\\
f(x_1^*, x_2^*) &= 38.818208
\end{aligned}$$