



机器学习及其MATLAB实现—从基础到实践 第10课

【声明】 本视频和幻灯片为炼数成金网络课程的教学资料，所有资料只能在课程内使用，不得在课程以外范围散播，违者将可能被追究法律和经济责任。

课程详情访问炼数成金培训网站

<http://edu.dataguru.cn>

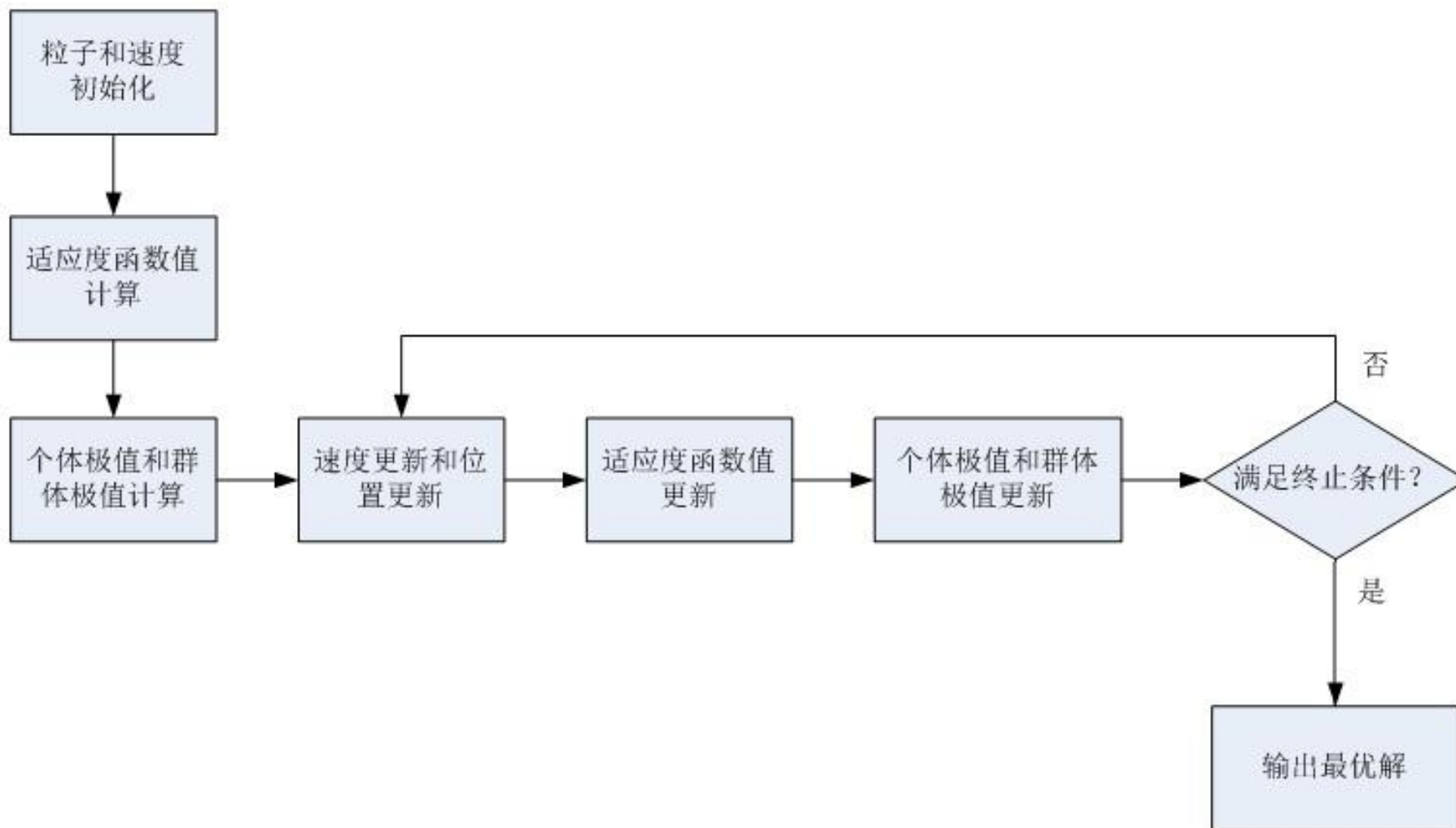
- 第一课 MATLAB入门基础
- 第二课 MATLAB进阶与提高
- 第三课 BP神经网络
- 第四课 RBF、GRNN和PNN神经网络
- 第五课 竞争神经网络与SOM神经网络
- 第六课 支持向量机 (Support Vector Machine, SVM)
- 第七课 极限学习机 (Extreme Learning Machine, ELM)
- 第八课 决策树与随机森林
- 第九课 遗传算法 (Genetic Algorithm, GA)
- **第十课 粒子群优化 (Particle Swarm Optimization, PSO) 算法**
- 第十一课 蚁群算法 (Ant Colony Algorithm, ACA)
- 第十二课 模拟退火算法 (Simulated Annealing, SA)
- 第十三课 降维与特征选择

- 粒子群优化 (PSO, particle swarm optimization) 算法是计算智能领域，除了蚁群算法，鱼群算法之外的一种群体智能的优化算法，该算法最早由Kennedy和Eberhart在1995年提出的，该算法源自对鸟类捕食问题的研究。
- PSO算法首先在可行解空间中初始化一群粒子，每个粒子都代表极值优化问题的一个潜在最优解，**用位置、速度和适应度值三项指标表示该粒子特征。**
- 粒子在解空间中运动，通过跟踪**个体极值**Pbest和**群体极值**Gbest更新个体位置，个体极值Pbest是指个体所经历位置中计算得到的适应度值最优位置，群体极值Gbest是指种群中的所有粒子搜索到的适应度最优位置。
- 粒子每更新一次位置，就计算一次适应度值，并且通过比较新粒子的适应度值和个体极值、群体极值的适应度值更新个体极值Pbest和群体极值Gbest位置。

- 在每一次迭代过程中，粒子通过个体极值和群体极值更新自身的速度和位置，更新公式如下：

$$V_{id}^{k+1} = \omega V_{id}^k + c_1 r_1 (P_{id}^k - X_{id}^k) + c_2 r_2 (P_{gd}^k - X_{id}^k)$$

$$X_{id}^{k+1} = X_{id}^k + V_{id}^{k+1}$$



- 相同点：

- ✓ 种群随机初始化
- ✓ 适应度函数值与目标最优解之间的映射

- 不同点：

- ✓ PSO算法没有选择、交叉、变异等操作算子
- ✓ PSO有记忆的功能
- ✓ 信息共享机制不同，遗传算法是互相共享信息，整个种群的移动是比较均匀地向最优区域移动，而在PSO中，只有gBest或lBest给出信息给其他粒子，属于单向的信息流动，**整个搜索更新过程是跟随当前最优解的过程**。因此，**在一般情况下，PSO的收敛速度更快**。

一元函数优化

二元函数优化

惯性权重 ω 体现的是粒子继承先前的速度的能力，Shi.Y 最先将惯性权重 ω 引入到 PSO 算法中，并分析指出一个较大的惯性权值有利于全局搜索，而一个较小的惯性权值则更利于局部搜索。为了更好的平衡算法的全局搜索与局部搜索能力，其提出了线性递减惯性权重 LDIW(Linear Decreasing Inertia Weight)，如下式所示：

$$\omega(k) = \omega_{start} - (\omega_{start} - \omega_{end})(T_{max} - k) / T_{max} \quad (1)$$

其中， ω_{start} 为初始惯性权重， ω_{end} 为迭代至最大次数时的惯性权重， k 为当前迭代代数， T_{max} 为最大迭代代数。一般来说，惯性权值取值为 $\omega_{start}=0.9$ 、 $\omega_{end}=0.4$ 时算法性能最好。这样，随着迭代的进行，惯性权重由 0.9 线性递减至 0.4，迭代初期较大的惯性权重使算法保持了较强的全局搜索能力，而迭代后期较小的惯性权重有利于算法进行更精确的局部搜索。线性惯性权重只是一种经验做法，常用的惯性权重的选择还包括如下几种。

$$\omega(k) = \omega_{start} - (\omega_{start} - \omega_{end})(k / T_{max})^2 \quad (2)$$

$$\omega(k) = \omega_{start} + (\omega_{start} - \omega_{end})(2k / T_{max} - (k / T_{max})^2) \quad (3)$$

$$\omega(k) = \omega_{end}(\omega_{start} / \omega_{end})^{1+(k/T_{max})} \quad (4)$$

- Dataguru (炼数成金) 是专业数据分析网站，提供教育，媒体，内容，社区，出版，数据分析业务等服务。我们的课程采用新兴的互联网教育形式，独创地发展了逆向收费式网络培训课程模式。既继承传统教育重学习氛围，重竞争压力的特点，同时又发挥互联网的威力打破时空限制，把天南地北志同道合的朋友组织在一起交流学习，使到原先孤立的学习个体组合成有组织的探索力量。并且把原先动辄成千上万的学习成本，直线下降至百元范围，造福大众。我们的目标是：低成本传播高价值知识，构架中国第一的网上知识流转阵地。
- 关于逆向收费式网络的详情，请看我们的培训网站 <http://edu.dataguru.cn>

Thanks

FAQ时间