



武汉大学

Wuhan University

第八章 MATLAB智能算法与机器学习



武汉大学

Wuhan University

§8.1 MATLAB智能算法

■ 智能算法简介

人工智能（artificial intelligence），缩写为AI，它是研究、开发用于模拟、延伸和拓展人的智能的理论、方法、技术及应用系统的一门技术科学。

■ 研究目标

- （1）对智能行为有效解释的理论分析
- （2）解释人类智能
- （3）构造智能的人工制品



武汉大学

Wuhan University

人工智能的典型应用

- 智能机器人
- 智能网络
- 智能医护
- 智能游戏
- 智能交易



武汉大学

Wuhan University

粒子群算法

- 粒子群优化算法属于进化算法的一种，从随机解出发，通过迭代寻找最优解，也是通过适应度来评价解的品质。

- 模拟鸟群捕食行为

一群鸟在随机搜索食物。在这个区域里只有一块食物。所有的鸟都不知道食物在那里。但是他们知道当前的位置离食物还有多远。那么找到食物的最优策略是什么呢。最简单有效的就是搜寻目前离食物最近的鸟的周围区域。



粒子群算法

- 每个寻优的问题解都被想像成一只鸟，称为“**粒子**”。所有粒子都在一个D维空间进行搜索。
- 所有的粒子都由一个fitness-function确定适应值以判断目前的位置好坏。
- 每一个粒子必须赋予**记忆功能**，能记住所搜寻到的**最佳位置**。
- 每一个粒子还有一个速度以决定飞行的**距离和方向**。这个速度根据它本身的飞行经验以及同伴的飞行经验进行动态调整。



基本公式

$$V_{id}^{k+1} = V_{id}^k + c_1 r_1 (p_{id}^k - x_{id}^k) + c_2 r_2 (p_{gd}^k - x_{id}^k)$$

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

- 粒子i的第d维速度更新公式：

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

- 粒子i的第d维位置更新公式：

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

v_{id}^k —第k次迭代粒子i飞行速度矢量的第d维分量
 x_{id}^k —第k次迭代粒子i位置矢量的第d维分量

c_1, c_2 —学习因子，调节学习的最大步长

r_1, r_2 —两个随机函数，取值范围[0, 1]，以增加搜索随机性

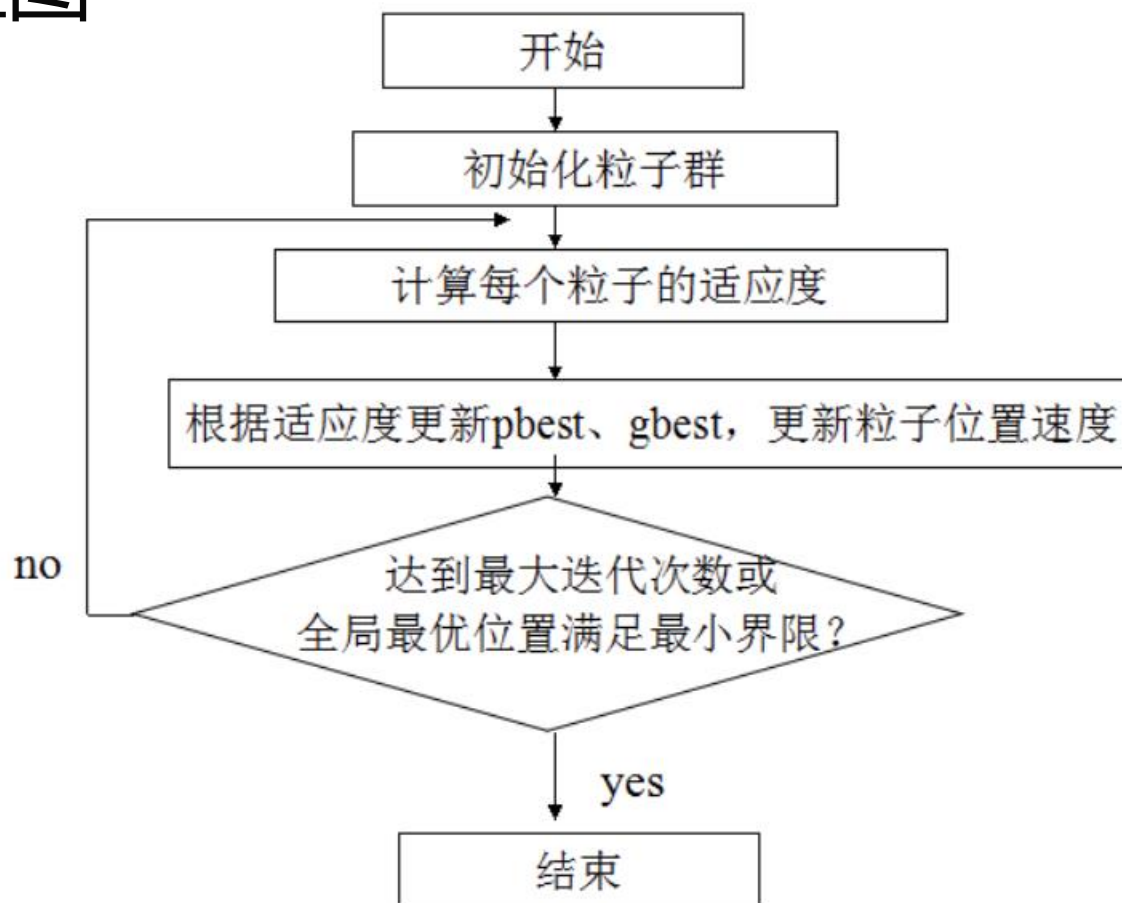
w —惯性权重，调节对解空间的搜索能力



武汉大学

Wuhan University

■ 流程图





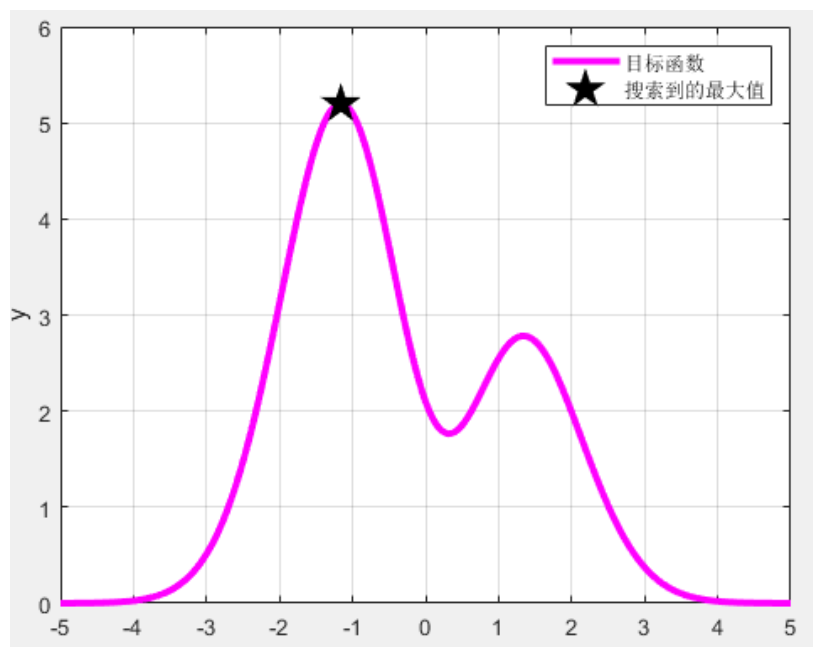
武汉大学

Wuhan University

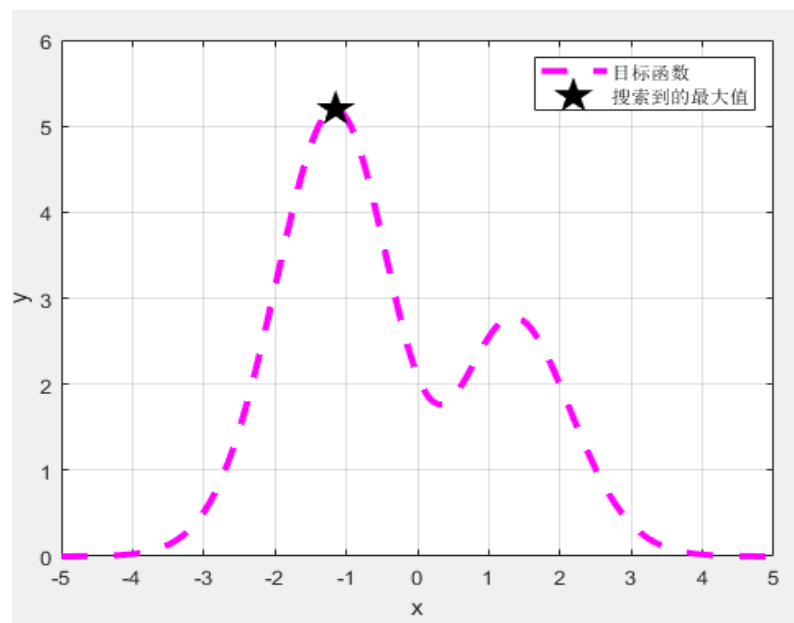
寻找一个给定函数的最大值

两种做法：1.直接计算 m8_1_1.m

2.粒子群算法 m8_1_2.m



直接计算

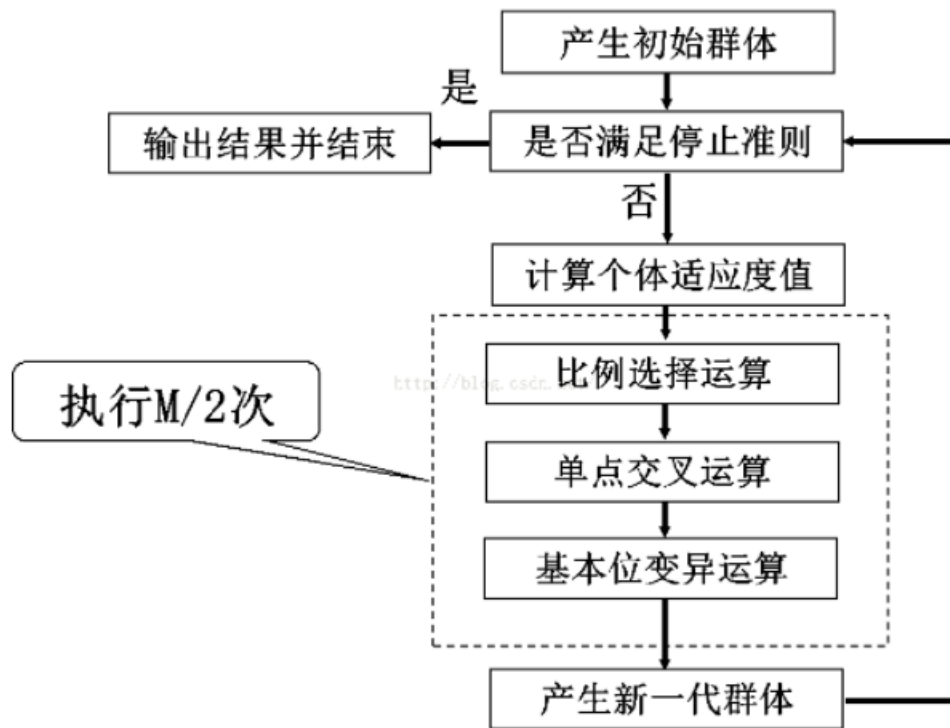


粒子群算法



遗传算法

- 遗传操作是模拟生物基因遗传的做法。在遗传算法中，通过编码组成初始群体后，遗传操作的任务就是对群体的个体按照它们对环境适应度施加一定的操作，从而实现优胜劣汰的进化过程。





武汉大学

Wuhan University

■ 遗传算法程序设计

Begin

$t=0$

初始化 $P(t)$

计算 $P(t)$ 的适应值；

while(不满足停止准则)

do

begin

$t=t+1$

从 $P(t+1)$ 中选择 $P(t)$

重组 $P(t)$

计算 $P(t)$ 的适应值

end

■ initpop.m函数：
实现群体的初始化

■ decodechrom.m函数：
将染色体（或二进制编码转换为十进制数）

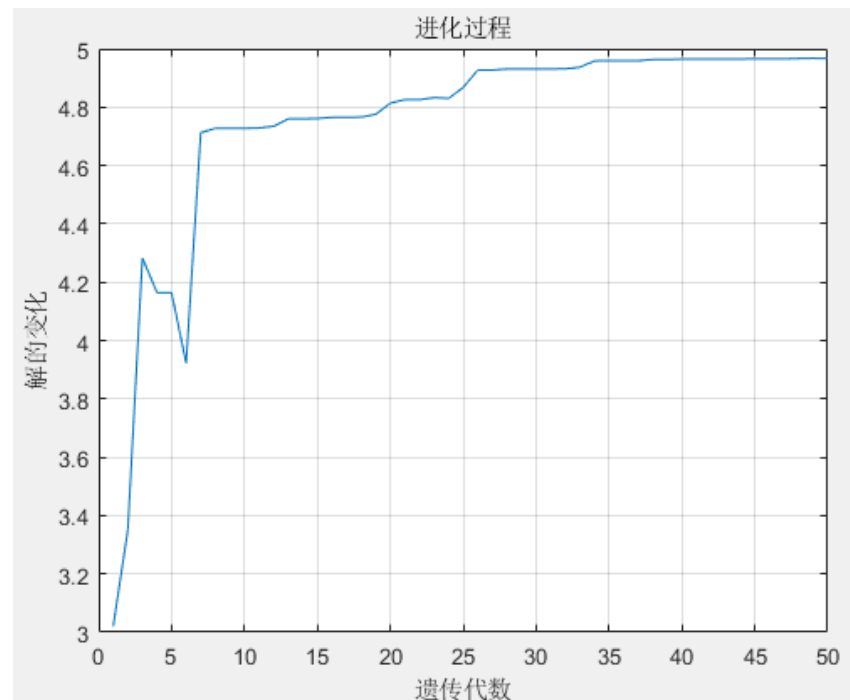
■ calobjvalue.m函数：
实现目标函数计算



武汉大学

Wuhan University

- 利用遗传算法对一个五元非线性函数的解进行最优化搜索
示例：[m8_1_3.m](#) [func.m](#)
- [func.m](#)定义一个任意的五元非线性函数,演示该程序时必须先下载安装gatbx遗传算法工具箱(code中已下，可以直接用)，[m8_1_3.m](#) [func.m](#)这两个文件需要在gatbx工具箱中运行。



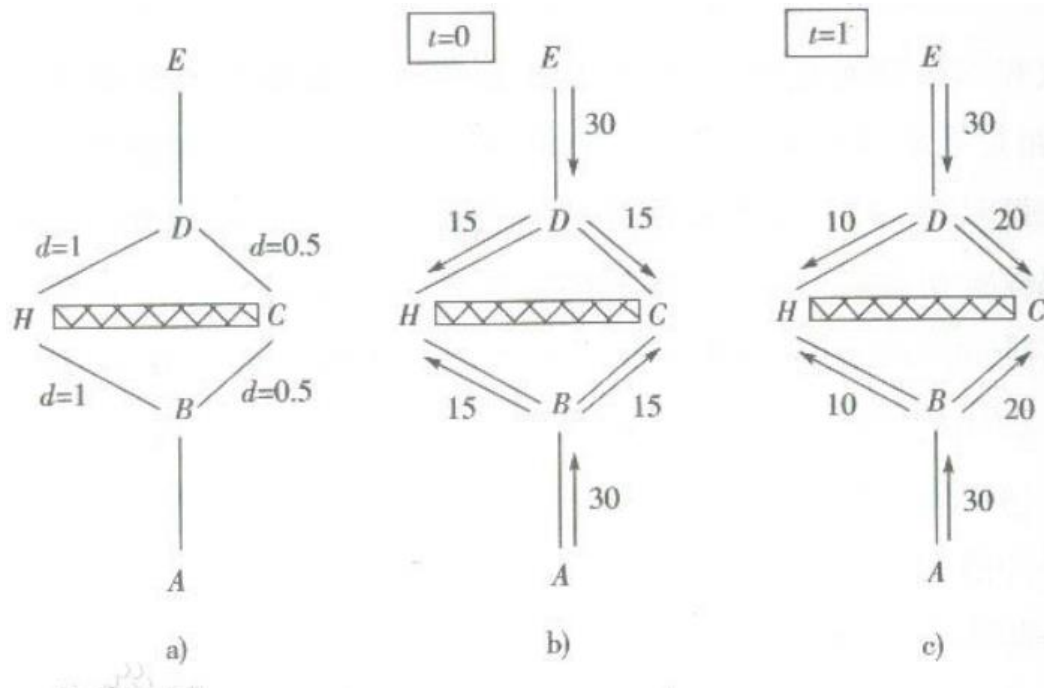


武汉大学

Wuhan University

蚁群算法

- 蚁群优化算法是模拟蚂蚁觅食的原理设计出的一种群集智能算法。蚂蚁在觅食过程中能够在其经过的路径上留下信息素，蚂蚁总是会朝着信息素强度高的方向移动，因此大量蚂蚁组成的集体觅食就表现为一种对信息素的正反馈现象。





武汉大学

Wuhan University

蚁群算法规则

■ 觅食规则

蚂蚁感知范围是一个 3×3 的格子，也就是说，他这个格子中有食物就直接过去。

■ 移动规则

蚂蚁会朝着信息素浓的地方移动，如果没有感知到信息素，就会按着惯性一直走下去，别忘了，蚂蚁会创新呦！

■ 避障规则

蚂蚁在遇到障碍物的时候会选择随机方向移动，同时遵循上面两个规则。

■ 信息素规则

蚂蚁在刚发现食物的时候挥洒的信息素会多，距离越远，信息素越少。



武汉大学

Wuhan University

■ 旅行商问题的蚁群算法优化

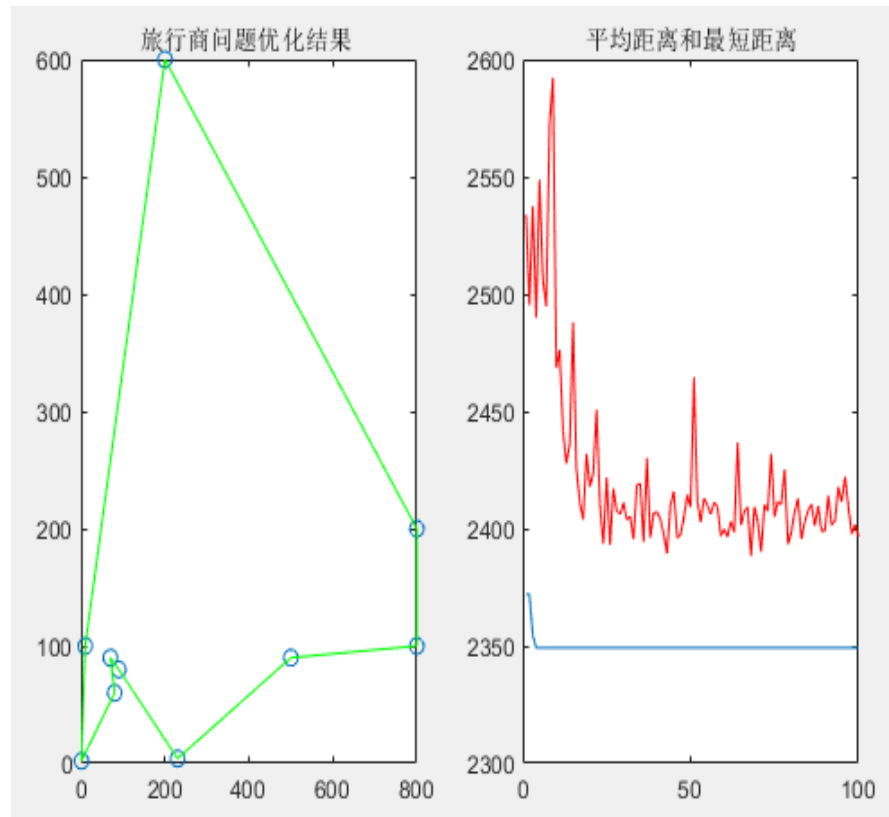
```
>> C=[1,2;70,90;80,60;10,100;800,200;  
800,100;90,80;200,600;230,4;  
500,90];
```

```
nc=100;m=18;a=1;b=5;p=0.5;
```

```
q=1;ACATSP(C,nc,m,a,b,p,q);
```

旅行商问题(Traveling Salesman Problem, TSP)是一个经典的组合优化问题。经典的TSP可以描述为：一个商品推销员要去若干个城市推销商品，该推销员从一个城市出发，需要经过所有城市后，回到出发地。应如何选择行进路线，以使总的行程最短。

ACATSP.m





§8.2 MATLAB机器学习

■ 机器学习简介

机器学习，致力于研究如何通过计算的手段，**利用经验来改善系统自身的性能**。在计算机系统中，“经验”通常以“数据”形式存在，因此机器学习所研究的主要内容，是关于在计算机上从数据中产生“模型”的算法，即“学习算法”。有了学习算法，**我们把经验数据提供给它，它就能基于这些数据产生模型，在面对新情况时，模型就会给我们提供相应的判断。**



武汉大学

Wuhan University

- 常见的机器学习算法

决策树、神经网络、支持向量机、贝叶斯分类器、聚类、深度学习、强化学习等。

- 机器学习领域的研究工作主要围绕以下三个方面进行：

- (1) 面向任务的研究

研究和分析改进一组预定任务的执行性能的学习系统。

- (2) 认知模型

研究人类学习过程并进行计算机模拟。

- (3) 理论分析

从理论上探索各种可能的学习方法和独立于应用领域的算法



BP神经网络

- BP神经网络又被成为误差反向传播（Back Propagation）神经网络，它是一种多层的前向型神经网络。在BP网络中，信号是向前传播的，而误差是反向传播的。BP网络通常具有一个或多个sigmoid隐层和线性输出层，能够对具有有限个不连续点的函数进行逼近。
- 所谓的反向传播是指误差的调整过程是从最后的输出层依次向之前的各层逐渐进行的。

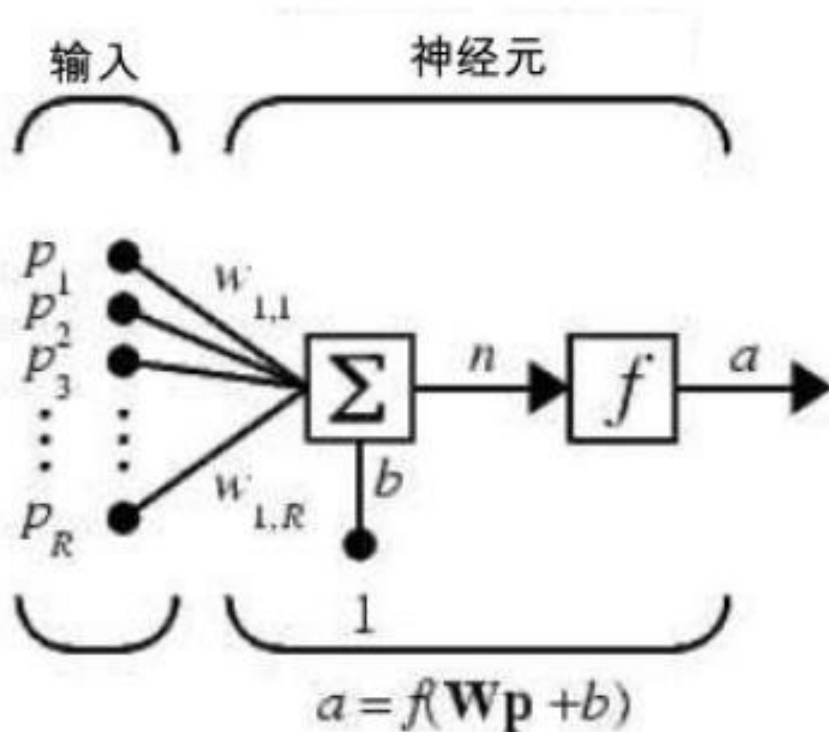


武汉大学

Wuhan University

BP网络神经元模型

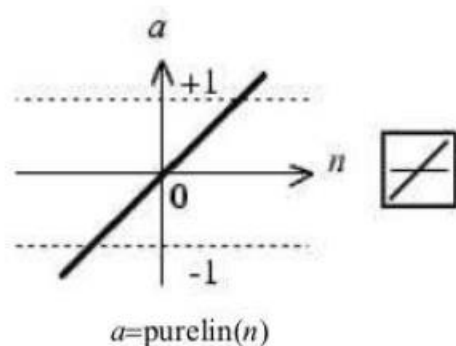
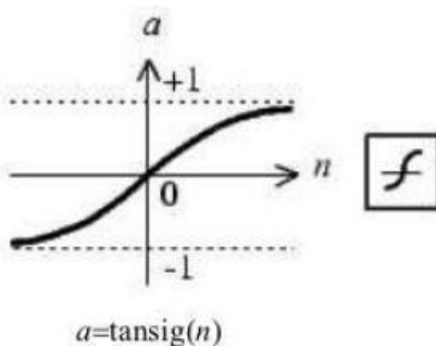
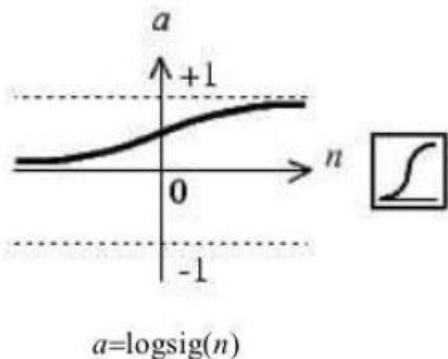
- 右图给出了一个具有R个输入的基本BP神经元模型结构。图中每一个输入被赋予一定的权值，与偏差求和后形成神经元传递函数的输入。





BP网络传递函数

- BP网络属于多层网络，其神经元常用的传递函数包括log-sigmoid型函数logsig、tan-sigmoid函数tansig和线性函数purelin

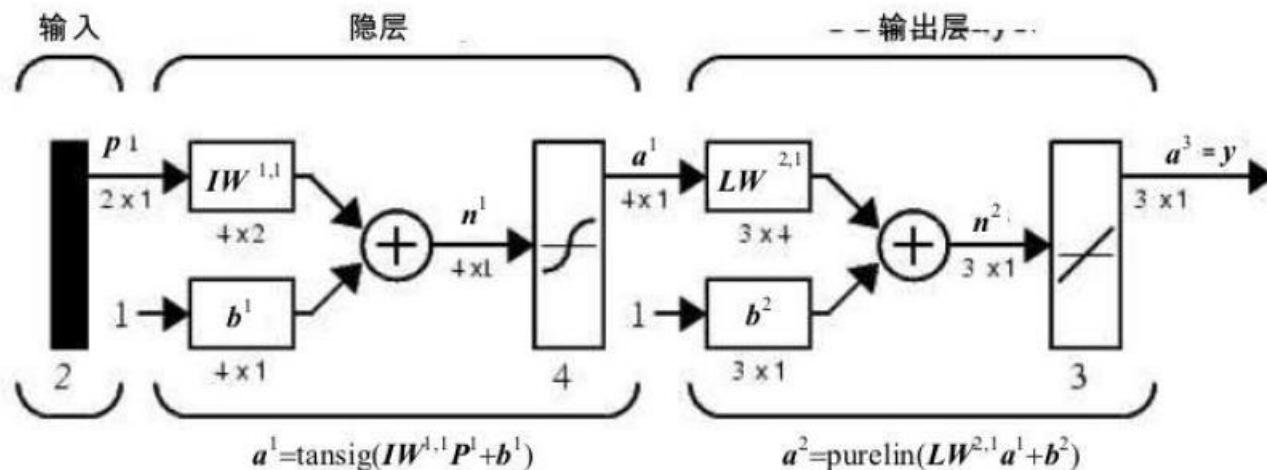


- 如果BP网络的输出层采用sigmoid，那么输出就限制在 $[-1,1]$ 之间，如果采用purelin，那么输出可以取任意值。所以常在隐层中采用sigmoid函数进行中间传递，在最后输出层用线性函数进行传递。



BP网络学习规则

- BP网络是有监督学习，训练过程需要提供输入向量和期望响应，训练过程中，网络的权值和偏差根据网络误差性能进行调整，最终实现期望的功能。
- BP网络的学习算法有很多，对应的训练函数包括traingd, traingdm, traingdx等等。
- 下图是一个两层神经元构成的BP网络结构





BP网络的MATLAB实现

- MATLAB神经网络工具箱提供函数 `feedforwardnet` 来创建一个前向型BP神经网络，常用调用格式为：

`net=feedforwardnet(hiddenSizes,trainFcn)`

- `hiddenSizes`：隐藏层神经元个数（一个行向量），默认值10
- `trainFcn`：用于训练网络性能所采用的函数，默认'`trainlm`'



BP网络创建示例 m8_2_1.m

- feedforward网络可以用作输入和输出的映射，若给予足够的隐藏神经元，两（或更多）层前馈网络可以实现任何有限的输入输出功能，只含有一个隐含层的神经网络可以拟合任意有限的输入输出映射问题。
- 给定一定的输入样本和目标响应，利用feedforwardnet函数创建一个单隐层的BP网络并对其进行仿真

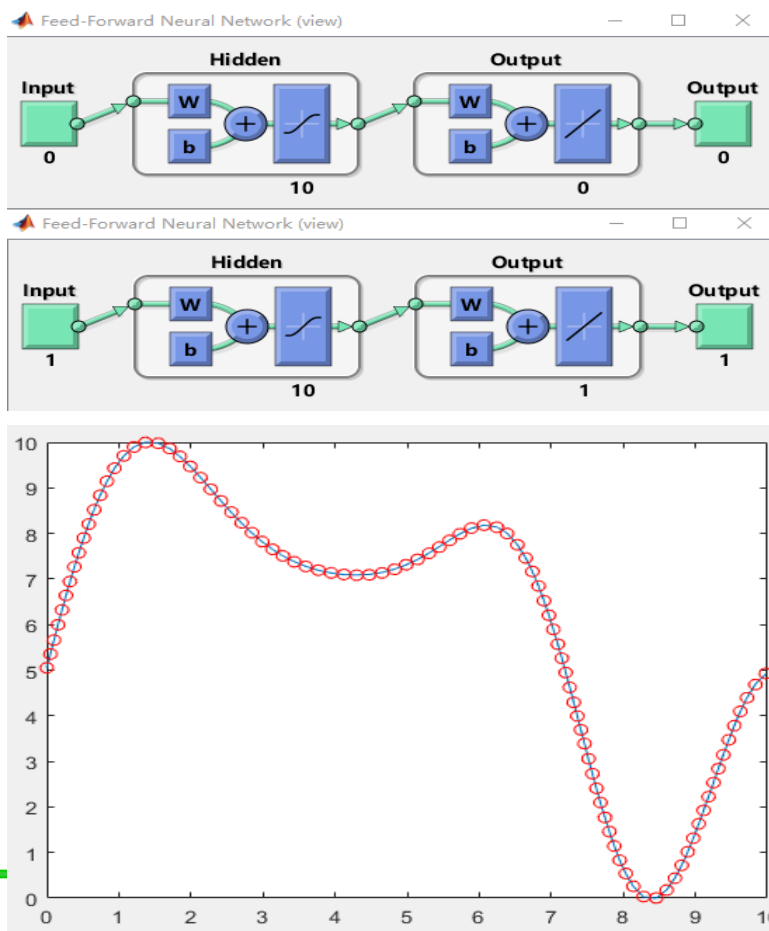
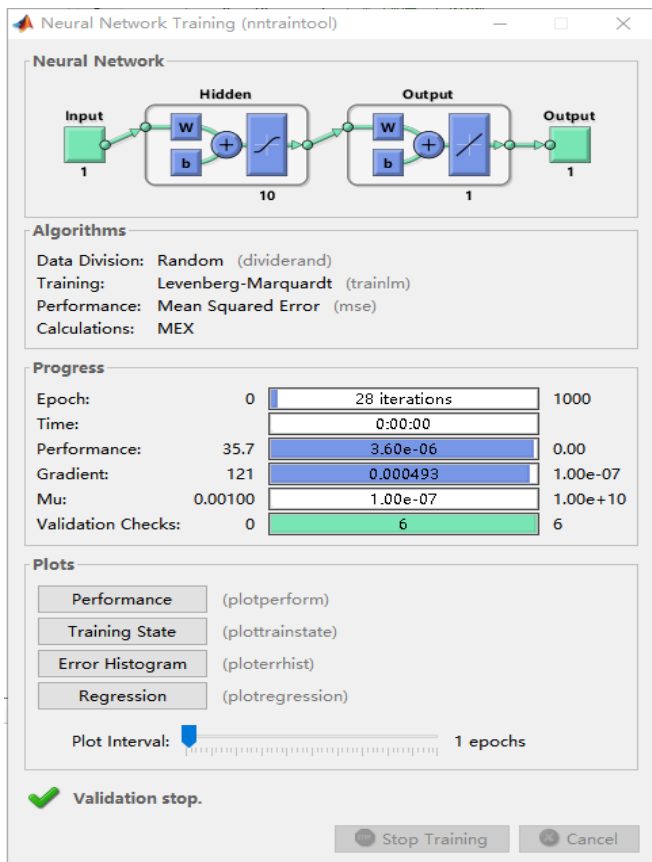
```
>> m8_2_1  
perf =  
3.6404e-05
```



武汉大学

Wuhan University

左图是调用的BP网络工具箱，右图为创建的前向网络的可视化图，以及输出结果图（蓝色线为期望输出，红色点为经过训练之后的拟合点）





武汉大学

Wuhan University

数据聚类

- 对数据进行聚类分析是神经网络一个非常优秀的功能，所谓的聚类，是指通过数据的相似度对数据进行归档。潜在应用方向有：
- 通过消费偏好分析进行市场划分
- 对数据进行数据挖掘和子集划分
- 通过对基因表达形式归类进行生物信息学研究

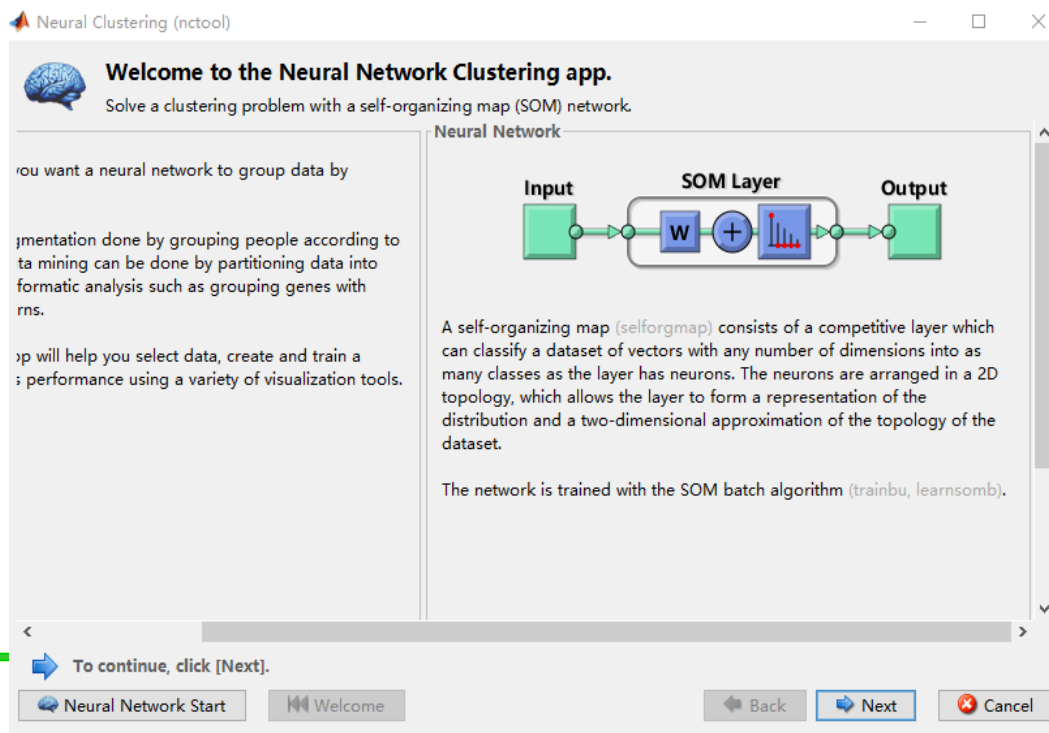


武汉大学

Wuhan University

数据聚类GUI工具nctool

- 对数据进行聚类分析，可以采用命令行、nntool、nctool三种方式，其中后两种均为GUI方式。
- 例：利用GUI工具nctool进行聚类分析
- 1. 输入nctool命令，弹窗如下所示。

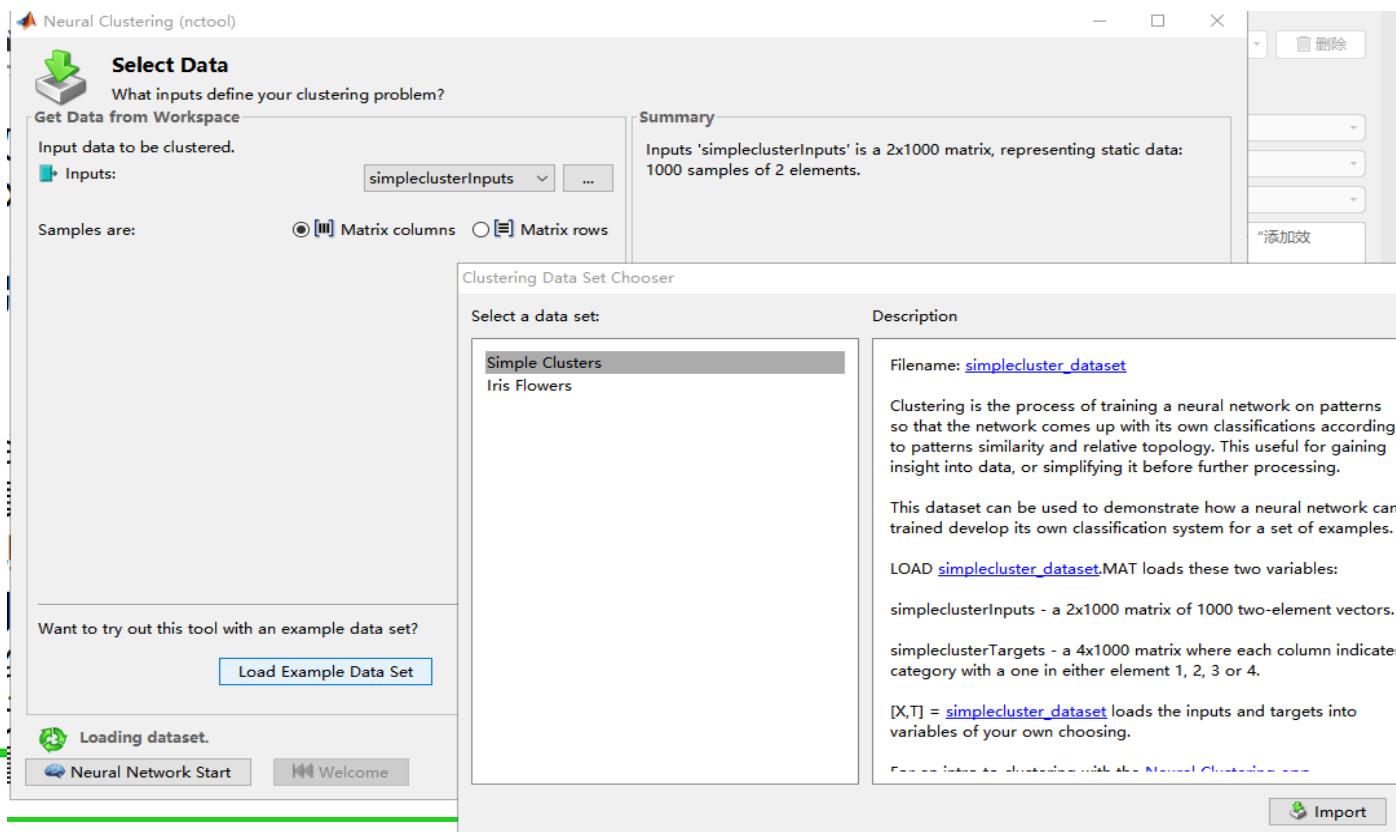




武汉大学

Wuhan University

2. 点击Next，弹出数据选择窗口，点击“Load Example Data Set”，选择MATLAB提供的样本数据集，弹出窗口如下所示：选择“Simple Clusters”，然后单击“Import”，回到数据窗口。此数据集具有1000个样本数据，输入为 2×1000 矩阵，输出划分为4类。

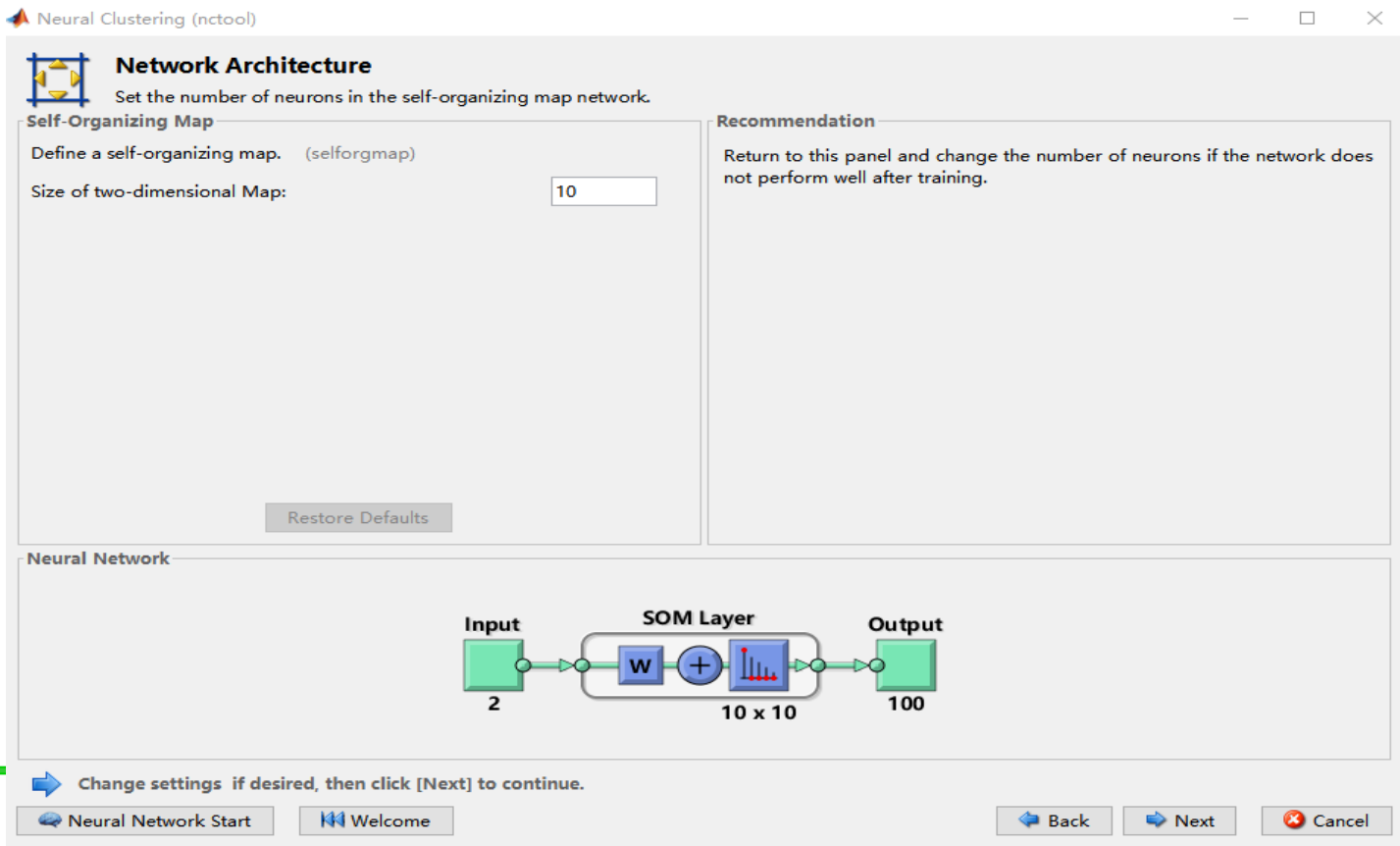




武汉大学

Wuhan University

3. 点击Next，进入到网络尺寸设置窗口，如下图所示，此二维自组织映射网络的尺寸默认设置为10，意味着每一边有10个节点，神经元总数为100个，我们选择此默认值作为待建立网络的设置。

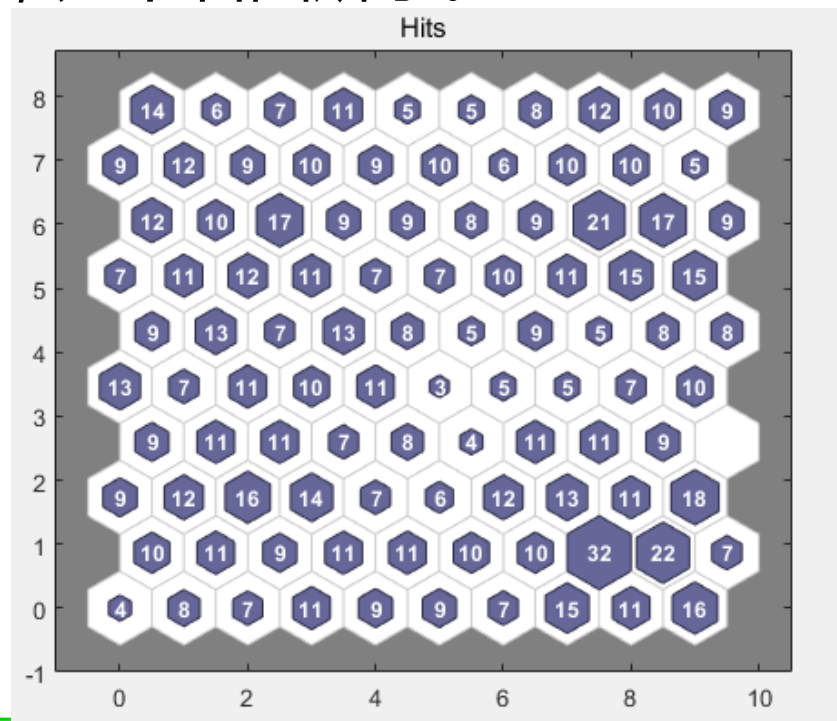
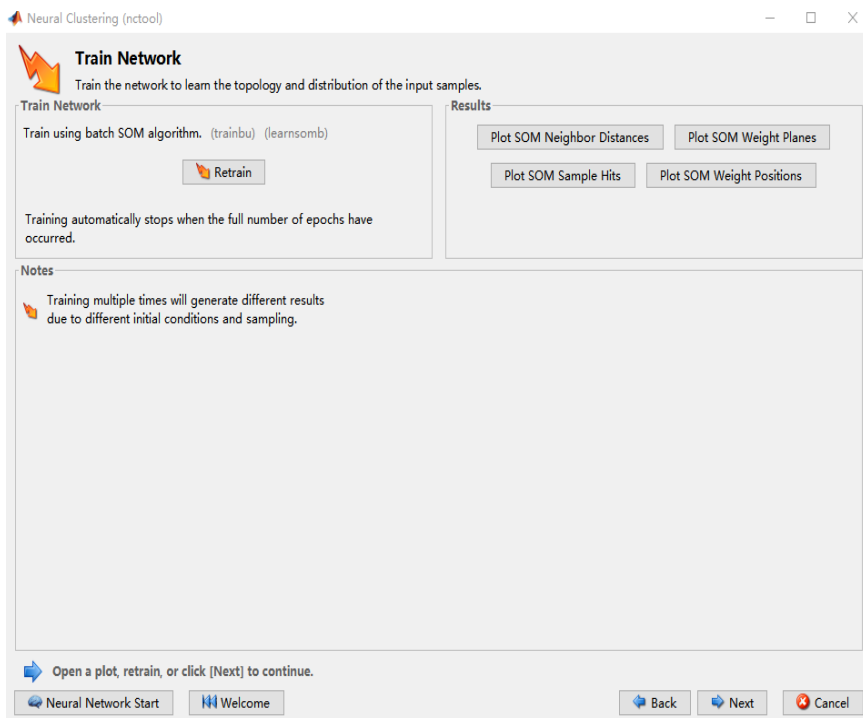




武汉大学

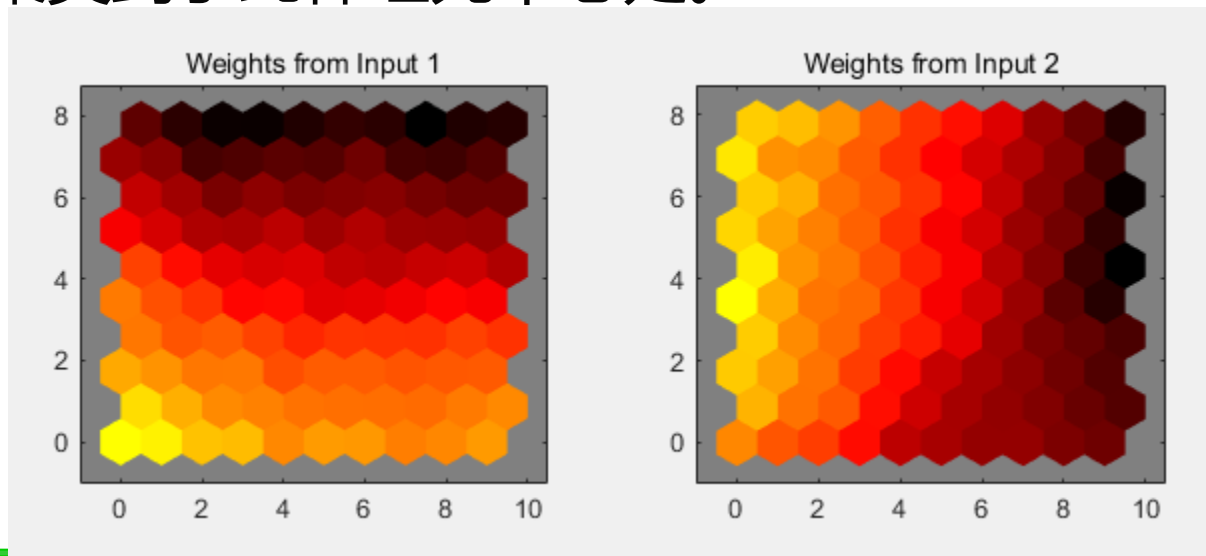
Wuhan University

- 4. 点击Next开始建立网络进入训练窗口，点击Train按钮开始训练，训练最大迭代次数为200，训练完成后如下左图所示。
- 5. 此窗口中点击Plot Sample Hits即可查看此SOM网络对数据进行聚类的命中结果，如下右图所示。





- 6. 如果需要查看此网络的权值位置平面，点击SOM Weight Planes按钮，绘图如下图所示。
- 图中显示了与每一个神经元（也就是聚类中心）相关联的输入训练数据的向量数目。该网络为 10×10 的二维拓扑个点结构，因此一共有100个神经元。从图中可见，具有最大命中数的神经元为32，也就是有32个输入样本被聚类到了此神经元中心处。





支持向量机 (SVM)

- 支持向量机主要是根据有限的样本信息在模型的复杂性和学习能力之间的寻求一个最佳的折中，在形式上类似于多层前向网络，但是能够克服一些多层前向神经网络的固有缺陷。基于统计学习的支持向量机方法能够从理论上实现对不同类别的最优划分，具有很好的泛化能力。
- 支持向量机最初来源于对数据分类的处理，对于线性可分的二值分类，通常是构造一个超平面并且移动它，直到找到一个合适的分界线。

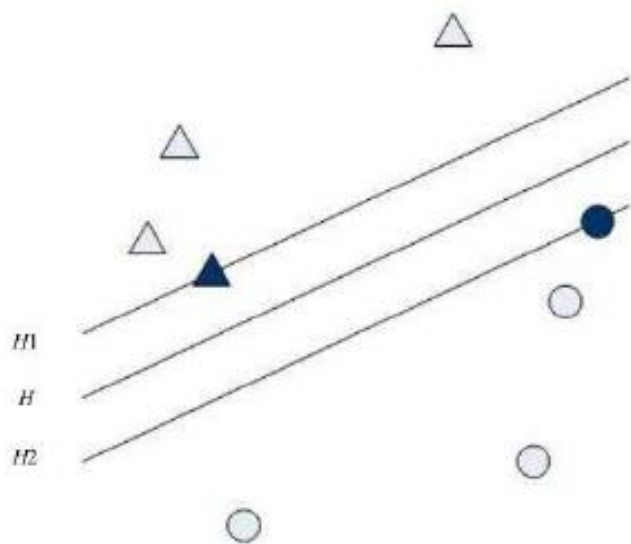


武汉大学

Wuhan University

线性支持向量机

- 对线性可分问题，如下图所示的上下两组样本点，通过平行直线 H_1 ， H ， H_2 都可以成功分开。 H_1 和 H_2 这样的两类直线之间的距离称为分类间隔， H 的分类是更优的。



- 所谓最优分类线，是使两类的分类间隔都最大化的分类线。能够把两类样本点正确无误的分开，是保证经验风险最小化，能够使分类间隔最小化，使得置信范围最小。在高维空间中，最优分类线就成为了最优分类面。

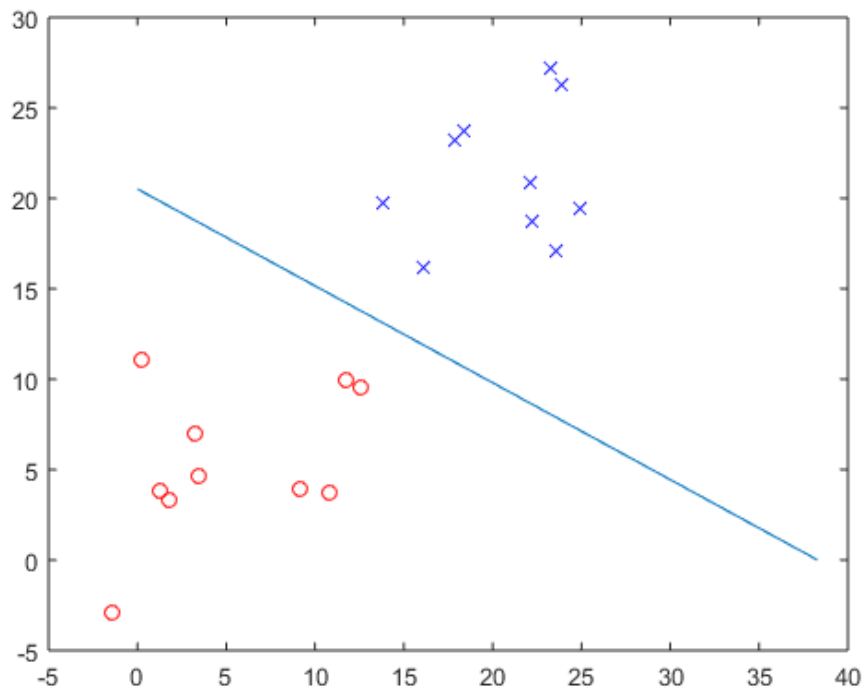


武汉大学

Wuhan University

支持向量机应用实例

- 任意给定两个不同的离散数据集，利用支持向量机得到合理的划分。
- 程序示例 `m8_2_2.m`
分类结果如右所示。





武汉大学

Wuhan University

- 习题：
- 1. 使用BP神经网络工具箱方式，对一个采样率为20Hz、频率为1Hz的正弦函数用BP网络来逼近其波形。
- 2. 文献调研：CNN（卷积神经网络）的MATLAB实现手段、工具等。