模型在控制系统中的作用？

数学模型有两种建模方法：
分析法——根据物理规律和化学规律列出方程式
实验法（系统辨识）——施加信号，记录响应，用适当的数学模型进行逼近
模型是为定性/定量分析、正确描述被控系统从而设计合适的控制算法而存在的，这里面模型指对现实被控系统主要问题的数学描述。
取决于所要描述问题的类型，模型可以表现为多种形式。如果是动态过程，可以用频域传递函数描述也可以用时域状态空间描述，可以在连续域描述，也可以在离散域描述；如果是静态非线性关系，可以用多项式也可以用特殊函数，甚至模糊、神经网络等描述；
建模应该是以控制算法设计/实施为导向的，不是要把整个系统都进行建模，而对和控制性能最相关、能够代表当前被控系统主要问题的部分进行建模。
无模型（Model-free）等在数据驱动控制常用到的称呼，并不是说不需要模型，而是指不需要机理模型。

袁如意老师课件思考题答案：

神经网络控制-1：

1.人工神经元激活函数的作用是什么？
完成数据的非线性变换，解决线性模型的表达、分类能力不足的问题；使它可以学习复杂的事物，复杂的数据，以及表示输入输出之间非线性的复杂的任意函数映射。
激活函数的另一个重要的作用是 执行数据的归一化，将输入数据映射到某个范围内，再往下传递，这样做的好处是可以限制数据的扩张，防止数据过大导致的溢出风险。

2.人工神经网络的学习规则有哪些？各有何特征？
a.Hebb 学习规则：神经元连接强度的变化与两个相互连接的神经元的激活水平成正比。两个神经元同时处于激活或抑制状态时，它们之间的连接强度将得到加强，反之应减弱。
W_ij(k+1)=w_ij(k)+I_iI_j
b.竞争学习规则：网络各输出单元相互竞争，最后达到只有一个最强者激活，最常见的一种情况是神经元之间有侧向抑制性连接。如输出单元中有一较强，则它将获胜并抑制其他单元，最优只有强者处于激活状态
c.Delta 学习规则：沿着J的负梯度方向不断修正W的值，直到J达到最小（梯度下降）

$$e_k(n) = d_k(n) - y_k(n)$$

极小化目标函数 $J = E\left[\frac{1}{2}\sum_k e_k^2(n)\right]$

瞬时值 $\xi(n) = \frac{1}{2}\sum_k e_k^2(n)$

$$y_k = \varphi(WX_k)$$

$$X_k = (x_{k0}, x_{k1}, \cdots, x_{kn})^T, k = 1,2,\cdots,m$$

$$W = (w_0, w_1, \cdots, w_n)$$

3.如何有效地训练一个神经网络？

a.训练一个神经网络，第一步要检查数据（特别是自己的数据集），确保正确

b.下一步就需要建立一个完整的训练/验证框架，并通过一系列的实验来确保正确性。最好先选择一个简单的模型把流程跑通 我们在这个模型上完成训练、可视化 loss 和一些其他的指标、进行模型预测和进行一系列消融实验。一些 tricks: **有效的评估、在初始阶段验证损失函数、一个好的初始化、数据送入网络前先可视化等**

c. **过拟合，**这一阶段，我们应该对数据集有一个很好的了解，并且有完整的训练/验证流程。现在可以开始迭代一个好模型了。一般为两个阶段：首先是使得模型足够强，能够在训练集上过拟合；然后在使用归一化策略，放弃一些训练 loss，从而降低验证 loss，达到一个平衡。

d.正则化，理想情况下，进行到这一步的时候，我们可以得到一个能够拟合训练集的模型了（有可能存在过拟合现象）。现在我们需要加一些正则化操作，是的模型具有更强的泛化能力。一些 tricks:更多的数据、**数据增强、预训练模型** 等

e.调参，tricks:随机网络搜索，超参数优化

f.精益求精，tricks: **模型合并是一种非常有保证非常靠谱的方法，可以在任何事情上提升 2%的精度；**自行训练。把模型放在一边然后让它自己一直训练。

4.BP 学习算法学习率的考虑因素？调整方式？

学习率越大，输出误差对参数的影响就越大，参数更新的就越快，但同时受到异常数据的影响也就越大，很容易发散。最理想的学习率不是固定值，而是一个随着训练次数衰减的变化的值，也就是在训练初期，学习率比较大，随着训练的进行，学习率不断减小，直到模型收敛。常用的衰减机制有：轮数衰减、指数衰减、分数衰减

5.常用传统辨识算法有哪些？神经网络辨识相对于传统算法有何优势？

a.

b.->神经网络本身作为一种辨识模型。其可调参数反映在网络内部的权值上，无需建立实际系统的辨识格式

->借助网络外部的输入/输出数据拟合系统的输入/输出关系可对本质非线性系统进行辨识。（网络内部隐含着系统的特性）辨识的收敛速度不依赖于被辨识系统的维数，只与神经网络本身所采用的学习算法有关。

->神经网络具有大量的连接。连接权值构成神经网络模型的参数，通过调节这些权值使网络输出逼近系统输出

->神经网络作为实际系统的辨识模型、构成系统的个物理实现，可用于在线控制。

6.完成系统辨识需具备哪些条件？辨识采用什么准则？

7.辨识系统的输入信号一般应具备什么特征？如何设计？

合理选择辨识的输入信号是能否获得好的辨识结果的关键之一。

为了使系统可辨识，输入信号必须满足一定的条件。最低要求是在辨识时间内系统的动态必须被输入信号持续激励。也就是说，在试验期间内输入信号必须充分激励系统的所有模态。更进一步，输入信号的选择应能使给定系统的辨识模型精度更高。这就引出了最优输入信号设计的问题。

在具体工程应用中，选择输入信号时还应考虑以下因素：

输入信号的功率和幅值不宜过大，以免使系统工作在非线性区，但也不宜过小，以致信噪比太小，直接影响辨识精度;

输入信号对系统的“静扰动”要小，即应使正负向扰动机会均等;
工程上要便于实现，成本低。
辨识中常用的输入信号有白噪声或伪随机信号。

8.系统可辨识性与可控性、客观性的关系？
不可控不可观的系统是不可辨识的，其次系统模型采用非规范结构（可控可观的规范结构）时，只利用输入输出数据也不能辨识内部模型的全部参数。
辨识是利用外部可观测状态来描述动态系统的数学模型，只反映过程的外部特性

9.只用输入输出信号能否辨识系统内部模型的全部参数？
不能
10.何如合理选择辨识模型，一般应考虑哪些因素？

11.直接逆控制方法优缺点？
优点：简单
缺点：无反馈，用作控制器的神经网络逆模型不准确时，抗干扰能力差，缺乏鲁棒性

神经网络控制-2：

1.lyapunov 函数在控制中的应用？
Lyapunov 方法适用于**线性系统和非线性系统、时变系统和时不变系统、连续时间系统和离散时间系统**。其分为 **Lyapunov 第一方法和第二方法**。Lyapunov 第一方法即 Lyapunov 间接法，属于小范围稳定性分析方法基本思路是，将非线性系统进行线性化，根据线性化**系统特征值**在复平面上的分布判断非线性系统在邻域内的稳定性。Lyapunov 第二方法即 Lyapunov 直接法，属于直接根据系统结构判断内部稳定性的方法，基本思路是，直接面对非线性系统，基于**引入具有广义能量属性的 Lyapunov 函数**和分析 Lyapunov 函数导数的定号性，建立判断系统稳定性的结论。

2.CMAC 网络的思想和优缺点
a.CMAC 的基本思想在于：在输入空间给出一个状态，从储存单元中找到对应于该状态的地址，将这些存储单元的内容求和得到 CMAC 的输出；将此响应值与期望输出值进行比较，并根据学习算法修改这些已激活的存储单元的内容。
b.它是一种基于局部逼近、简单快速的神经网络，能够学习任意多维非线性拟合。相比于 BP 网络等全局逼近方法。
CMAC 具有以下优点：
-局部学习，每次修改权值少，学习速度快，适合在线学习
-具有一定的泛化能力，即相近输入产生相近输出，不同输入给出不同输出
-具有连续（模拟）输入输出能力
-采用寻址编程方式，在利用串行计算机仿真时，它将使响应速度加快
-对学习数据出现的次序不敏感
CMAC 最初主要用来求解机械手的关节运动，后来被进一步应用于机械人控制、模式识别、

信号处理以及自适应控制等领域
CMAC 有以下缺点;
(1)CMC 神经网络权值系数的**存储空间随着小脑模型输入维数增大而急剧增加**,为了解决存储空间问题，人们通常采用杂散编码技术，但采用杂散编码方法在学习时存在冲突问题，即 Ac 中的多个存储单元被映射到 Ap 的同一单元，这就意味着信息的丢失，所以可能会导致学习速度下降或者学习发散。
(2)CMAC 神经网络是一个**确定型网络**，不利于模拟客观事物。
(3)既然 CMAC 是一个神经网络模型，肯定就避免不了神经网络固有的缺陷那就是神经网络的**内部知识表达是不清楚的**。所以在每次学习时候，只能从任意的初始条件开始，不能利用必要的初始经验或知识。
(4)在 CMAC 神经网络应用中，一般来说其**实时性要求都较高**。如非线性动态系统的在线辨识，不仅要求精度高，而且要求快速学习。但是,常规的 CMAC 仍然需要多个周期才能达到一定的收敛精度,也就是说,常规 CMAC 虽然其收敛速度快于 BP 网络,但作为在线学习来说,仍难满足其快速性的要求.

3.机械手控制难点，神经网络控制如何应对？
a.算法难点:如果对机械臂运动实时性要求较高，就需要根据机械臂结构，手写机械臂逆运动学解析解等，对于不同构型机械臂适配不同解析算法，很是麻烦
b.控制的难点在于逆运动学，由于机械臂的各个轴都是耦合的，进行逆运动学解算相当于解非线性方程组。有的人可能会说可以用迭代的方法得到，但是机器人的实时控制等不起，控制周期最多 1ms，一个迭代解算过程可能就需要几十 ms。。

机械手是一种高度非线性、强耦台、时变的系统，对它的控制基本为两种**基于精确数学禛型的传统控制和与模型无关的智能控制**。由于机械手系统的复杂性，其精确的数学模型难以建立，使应用传统的控制手段对其进行的控制效果欠佳。常用的智能控制手段如模糊控制、人工神经网络等又有各自的局限性。一般说来,模糊逻辑方法虽然长亍表达近似与定性的知识，却通常无学习能力；神经网络具有学习能力，但内部知识的表达方式又是不清楚的，这样神经网络在每次学习时只能从任意初始条件开始，不能利用必要的初始经验或知识，收敛速度侵，易入局部极限；而由于缺乏学习能力，模糊逻辑方法只能主观或试凑地选择隶属函数和模糊规则，不能根据积累的经验自动地改系统的性能。

CMCA 神经网络，是一种类似于 Perceptron 的相联记忆方法，与模糊逻辑不但是相互补充的，而且也是相互结台的。首先它用**连接主义**来表达模糊逻辑控制器，引入了学习机制，也带来了两者结合的诸多优点，如**存储容量的减小，泛化能力的增加，以及连接主义结构的容错性等**。其次，**在 CMAC 的分布表达中，一个值由散布于许多计算单元的活性模式表示**，每个计算单元又涉及许多不同值的表达，因此每个计算单元都有一个感受野（ receptive field),即它表达的所有值的集合。

4.神经网络的容错性、联想记忆特性与网络结构的关系？
人工神经网络是有大量的简单处理单元相互连接构成的高度并行的非线性系统,具有大规模并行性处理特征。**结构上的并行性**使得神经网络的信息存储必然分布式方式，即信息不是存储在网络的某个局部，而是**分布在网络所有的连接权中**。神经网络内在的并行性与分布性表现在其信息的存储余处理都是都是在空间上分布、时间上并行的，这连个特点必然使神经网

络在两个方面表现出良好的容错性：**一方面由于信息的分布式存储，当网络中部分神经元损环时不会对系统的整体性能造成影响，**这一点就像人脑中每天都有神经细胞正常死亡而不会影响大脑的功能一样；**另一方面单输入模糊、残缺或变形的信息时，神经网络能够通过联想恢复出完整的记忆，**从而实现对不完整输入信息的正确识别，这一点就像人可以对不规则的手写进行正确识别一样。

人工神经网络是模拟大脑建立起来的模型，因此吸取了生物神经网络的许多优点。它神经网络由许多具有非线性映射能力的神经元组成，神经元之间通过权系数相连接。这种大规模并行结构具有很高的计算速度，完全不同于传统模型。人工神经网络的信息分布存储于连续权系数中，使网络具有很高的容错性，减少了模式识别中往往存在噪声干扰或输入模式部分损失的影响。人工神经网络的自组织、自适应学习功能，大大放松了传统识别方法所需的约束条件，使其对某些识别问题显示出极大的优越性。此外，它还具有高度的非线性全局作用、高度的并行性、联想记忆功能等等。

5.Hopfield 网络如何用于优化计算？

离散 hopfield 网络

若稳态为记忆样本，则收敛过程便是寻找记忆样本过程（联想记忆）

若稳态对应某种优化目标，并作为目标函数的极小点，收敛过程辨识优化计算过程

连续 Hopfield 网络用于组合优化计算

把最优化问题的目标函数转换成网络的能量函数，把问题的变量对应于网络的状态，网络达到稳定状态时，就是它的能量函数达到最小的时候，对应于优化问题的解。网络的计算量不会随维数的增加发生指数性巨增，对于优化问题的快速计算特别有效。

连续 Hopfield 网络用于优化计算一般步骤

1. 变结构控制的原理及特点？

滑模控制(sliding mode control, SMC)也叫变结构控制，本质上是一类特殊的非线性控制变结构控制，根据某种原则，切换控制量。

滑模变结构控制的原理，是根据系统所期望的动态特性来设计系统的切换超平面，通过滑动模态控制器使系统状态从超平面之外向切换超平面收束。系统一旦到达切换超平面，控制作用将保证系统沿切换超平面到达系统原点，这一沿切换超平面向原点滑动的过程称为滑模控制。

由于系统的特性和参数只取决于设计的切换超平面而与外界干扰没有关系，所以滑模变结构控制具有很强的鲁棒性。

滑模控制的优点是能够克服系统的不确定性， 对干扰和未建模动态具有很强的鲁棒性，尤其是对非线性系统的控制具有良好的控制效果。由于变结构控制系统算法简单，响应速度快，对外界噪声干扰和参数摄动具有鲁棒性，在机器人控制领域得到了广泛的应用

滑模控制的缺点：当状态轨迹到达滑动模态面后，难以严格沿着滑动模态面向平

衡点滑动，而是在其两侧来回穿越地趋近平衡点，从而产生抖振——滑模控制实际应用中的主要障碍。

变结构具有如下特点
在满足一定条件下,变结构系统的滑动模对系统的批动和参数摄动具有完全兽棒性或有不变性。这是这个独特的优点使得变结构控制具有强大生命力
在滑动模态阶段,变结构控制系统的特性可以由个降阶的等效运动方程来完全表征。并且这个等效滑动模态方程的品质可以预先通过极点配置、最优控制来保证。

2. 变结构控制的设计步骤与思路？
在系统控制过程中，控制器根据系统当时状态，以跃变方式有目的地不断变换，迫使系统按预定的“ 滑动模态 ” 的状态轨迹运动。变结构是通过切换函数实现的，特别要指出的是，通常要求切换面上存在滑动模态区，故变结构控制又常被称为滑动模态控制。设计变结构控制系统基本可分为两步：、
a.**确定切换函数 S（ x）**
即开关面，使它所确定的滑动模态渐近稳定且有良好的品质，开关面代表了系统的理想动态特性。
b. **设计滑模控制器**
设计滑模控制器，使到达条件得到满足，从而使趋近运动（非滑动模态 ）于有限时间到达开关面，并且在趋近的过程中快速、 抖振小。
既保证所有运动（趋近运动、非滑动模态）于有限时间时间到达切换面，又保证切换面是滑动模态区

3. 变结构控制的品质及实现？
设计可以分解为两个完全独立地阶段：到达阶段和 滑动模态阶段，在到达阶段，系统能够在任意初始 状态出发，在变结构规律的作用下进入并到达滑动 模态，在滑动超平面上产生滑动模态运动，趋向于 状态空间原点
可以在保证稳定性的同时具有快速的响应特性。
变结构控制突破了经典线性控制系统的品质限制，较好解决了动态和静态性能指标之间的矛盾，相对于其他控制方法，变结构控制系统的物理实现较为简单。

4. 抖振产生的原因及消除方法？

抖振问题产生的原因

1. 时间滞后（控制作用对状态准确变化有滞后）　　惯性滞后
2. 空间滞后（状态空间中的状态量变化死区）　　惯性滞后
3. 固定空间滞后（元器件本身静特性具有滞后空间，结构性的）
4. 离散时间系统本身造成的抖振
5. 未建模动力学影响

直观解释：当系统的轨迹到达切换面时，其速度是有限大，惯性使运动点穿越切换面，从而最终形成抖振，叠加在理想的滑动模态上。

变结构控制（VSC)是一种开关型控制，它在工作过程中频繁地切换系统的控制状态，因而产生了抖振。
在实际变结构控制中，由于惯性滞后的因素存在，抖振的产生是必然的。由于现实中物理能量不可能无限大，从而使系统的控制力不能无限大，这就必然使得系统的加速度有限；另外，系统的惯性总是存在的，于是，控制的切换必然伴有后。这种滞后造成的抖振与时间问滞后的结果类似。有的系统本身存在时间和空间的滞后，这种滞后往往造成很大的抖振，
抖振产生的根本原因是在实际的控制系统中不可能实现理想的切换

抖振问题的削弱方法

1. 准滑动模态方法（系统运动轨迹被限制在边界层）

采用饱和函数代替切换函数，即在边界层外采用正常的滑模控制，在边界层内为连续状态的反馈控制，有效地避免或削弱了抖振。

2. 趋近律方法（保证动态品质、减弱控制信号抖振）
3. 滤波方法（通过采用滤波器，对控制信号进行平滑滤波）
4. 观测器方法（补偿不确定项和外界干扰）
5. 动态滑模方法（高阶滑模）
6. 智能控制方法

5. 摄动与干扰及存在不确定情况下的滑模变结构控制设计？

6. 神经网络、模糊控制与滑模变结构方法的结合？

强化学习

强化学习算法思想及与别的类型算法之间的区别？

监督学习一般有标签信息，而且是单步决策问题，比如分类问题。监督学习的样

本一般是独立同分布的。无监督学习没有任何标签信息，一般对应的是聚类问题。强化学习介于监督和无监督学习之间，每一步决策之后会有一个标量的反馈信号，即回报。通过最大化回报以获得一个最优策略。因此强化学习一般是多步决策，并且样本之间有强的相关性。
区别于监督学习和无监督学习，强化学习是一种半监督的学习方式，其通过智能体主动与环境进行交互，然后由环境给出反馈信息，智能体据此改进行动方案以适应环境，达到预期的目的。

确定性策略和随机策略比较？

确定性策略和随机策略是 model free 策略搜索的两类主要方法。
确定性策略，在相同的状态下，其输出的动作是确定的；而对于随机策略，对于相同的状态，其输出的状态并不唯一，而是满足一定的概率分布，从而导致即使是处在相同的状态，也可能输出不同的动作。
另外，就两者的优缺点来说，确定性策略的优点是能够利用确定性梯度优化策略，所以不需要太多的采样数据，计算效率也很快。缺点是由于每次面对同一状态其输出的动作是唯一的，无法讨论一些其它动作的效果，不具有自学习的能力；而随机策略的优点是能够探索好的经验，然后集成到一个策略中。而且随机策略比较成熟，现有的轮子比较多。而缺点是需要采样的数据量较大，学习比较慢；

探索-利用问题？

探索是对环境无知识时，利用一些试探性的动作，然后看环境的反馈
利用是根据当前的知识，看采取哪些知识时，回报会高或者低，然后可以对应地去利用这些知识
强化学习要去平衡探索与利用之间的关系
对一些未探索的可能需要采取一些探索的动作。

同策略与异策略方法的优缺点？

- **同策略（on policy）方法**：如果采样（行动）策略是$\pi^{\epsilon}(s)$，不断改进策略也是$\pi^{\epsilon}(s)$而不是目标策略$\pi(s)$。这种采样策略与目标策略相同（即都是$\pi^{\epsilon}(s)$）的学习方法叫做同策略方法　　直接了当，速度快，但不一定找到最优策略

- **异策略（off policy）方法**：如果采样策略是$\pi^{\epsilon}(s)$，而优化目标是策略$\pi(s)$，采样与改进分别使用不同策略的强化学习方法叫做异策略方法

  收敛慢，更为强大和通用，确保了数据的全面性，所有行为都能覆盖，可找到最优解

采样策略（行动策略）：产生样本用于评估的策略
目标策略：需要学习和改进的策略

同策略（on-policy）的代表算法 Sarsa，亦称 on-line Q-learning，其采样的策略（用于执行，behavior policy）和更新 Q 值的策略（用于评估，target policy）一样,行为策略和目标策略均为贪心策略。Sarsa 的每次 Q 值更新需要知道前一步的状态（state）、前一步的动作（action）、奖赏值（reward）、当前状态（state）、将要执行的动作（action），由此得名 **Sarsa 算法**。
同策略指生成样本的策略与网络更新参数策略相同,其基于当前的策略直接执行下一次动作选择，然后用这个样本更新策略，生成样本的策略和学习时的策略相同。
优点：每一步都可以更新，这是显然，学习速度快；可面对没有结果的场景，应用范围广。
缺点：遭遇探索-利用的矛盾；只利用已知的最优选择，可能学不到最优解；收敛到局部最优，加入探索而降低学习效率。

异策略（off-policy）的代表算法 Q-learning，亦称 SarasMax，其采样的策略（用于执行,behavior policy）和更新 Q 值的策略（用于评估,target policy）不一样，行为策略为贪心策略，而 target policy 为确定性策略，即选择最 Q 值最优的 action

时间差分、蒙特卡罗方法、动态规划方法的区别与联系？

蒙特卡洛方法：利用所有回报的累积估计值函数（值函数最原始的定义）。
动态规划和时间差分：用一步预测方法计算当前状态值函数。
不同：1.动态规划方法利用模型得到后继状态；2.时间差分方法利用实验得到后继状态。

时序差分的信用分配机制？

引入值函数逼近的原因？优缺点？

直接策略搜索与值函数方法的比较？