计算方法计算大题常考点总结(初稿)

引言:本文主要根据教材,考试范围与考试常考点总结重点难点大题的解题详细步骤。因此本文中并不会涉及过多的最基本的概念，只是在掌握最基本概念上的一个针对期末考试的总结。

第一章 绪论

绪论主要介绍绝对误差、相对误差、有效数字、误差的传播与度量等概念，该部分考点很少，在期末考试中出现的内容极大概率是有效数字的求解。

有效数字问题求解步骤(必须掌握):

$$\begin{cases} ①x^* = \pm 10^m \times 0.a_1a_2\cdots a_p \to \text{确定}m\text{的值} \\ ②|x^* - x| \leqslant \dfrac{1}{2}\times 10^{m-n} \to \text{确定}m-n\text{的值} \to \text{确定}n\text{的值} \end{cases}$$

即有n位有效数字.

第二章 非线性方程组数值解法

本章必然会有一道大题，二分法考的很少(因为步骤过于简单)但是不能完全排除，极大概率是简单迭代法与牛顿迭代法两个中的一个。第二章在考试的时候出的大题属于比较简单的一类，多算几遍力争算对

对于迭代法的一些个人理解:

迭代法是一种通过反复重复的计算来逐步逼近某个理论值的方法。在迭代法中，每一次的迭代都会利用上一次计算的结果进行新的计算，知道达到一定的收敛条件为止。

在进行近似计算的时候，每一步的计算都会产生误差，迭代法的精髓之处在于每一次的迭代都用到了上一次计算的结果，因此每一次迭代都在修正上一次计算中存在的误差，并且不断逼近真实的解，当迭代次数趋于无穷大时，迭代法会达到收敛状态，当然这是需要一定条件的，并不是所有的迭代格式都是收敛的，因此很显然，**探究的重点就是迭代格式的构造以及迭代收敛的条件**

本章所介绍的是简单迭代法，这种迭代方法可以通过画图直观的体现迭代法的思想

一、二分法解题步骤

题目通常已知函数$f(x)$，区间$[a,b]$，并且规定了误差ε

$$\begin{cases} ①\text{计算}f(a),f(b),f\left(\dfrac{a+b}{2}\right)\text{的值} \\ ②\text{根据给定误差确定最小二分次数} \to \text{二分次数}k \geqslant \dfrac{\ln(b-a)-\ln\varepsilon}{\ln 2} \\ ③\text{重复进行二分计算函数值} \end{cases}$$

注意:二分法唯一需要注意的点就是需要记忆二分次数和误差估计的公式

$$|x_k - x^*| \leqslant \frac{b-a}{2^k}、k \geqslant \frac{\ln(b-a)-\ln\varepsilon}{\ln 2}$$

二、简单迭代法解题步骤(必须掌握):

题目通常已知函数$f(x)$,区间$[a,b]$，并且规定了误差ε

$$\begin{cases} ①\text{由}f(x)=0 \xRightarrow{\text{通常经过简单变换即可}} x=\varphi(x) \xRightarrow{\text{构造迭代格式}} x_{k+1}=\varphi(x_k) \\ ②\text{验证迭代序列收敛,步骤为}\begin{cases} ①x\in[a,b],\varphi(x)\in[a,b] \\ ②0<L<1,\text{使得}|\varphi'(x)|\leqslant L(\text{导数最大值小于1即可}) \end{cases} \\ ③\text{误差估计}\begin{cases} \text{后验误差估计}|x_k-x^*|\leqslant \dfrac{L}{1-L}|x_k-x_{k-1}| \\ \text{先验误差估计}|x_k-x^*|\leqslant \dfrac{L^k}{1-L}|x_1-x_0| \end{cases} \leqslant \varepsilon(\text{一般用先验误差估计公式先确定迭代次数}) \end{cases}$$

若满足迭代序列全局收敛性条件，即可使用①的迭代格式进行迭代，若不满足条件需要重新构造

注意:要懂得根据题意变化进行步骤的调整，误差估计公式除了可以用来估计误差还可以在给定误差的情况下反求迭代次数。若题目给定了误差ε，就需要先求迭代次数再进行迭代

三、*Newton* 迭代法解题步骤

Newton 迭代法本质上也是简单迭代法的一种，只是它给出了一种相对固定的构造迭代格式的方式，因此直接记住公式就行

$$\begin{cases} ①\text{构造迭代格式}:f(x)=0\Rightarrow x_{k+1}=x_k-\dfrac{f(x_k)}{f'(x_k)},\text{开始迭代} \\ ②\text{误差估计}:|x_n-x_{n-1}|\leqslant\varepsilon \end{cases}$$

第三章 线性代数方程组的解法

本章在考试的计算题部分也会有一个大题，大概率是考线性方程组的LU分解，或者是考线性方程组的迭代法，并且考试出现的线性方程组是四阶的；高斯顺序消去法和高斯列主元方法几乎不会考，因为这属于线性代数的内容，在线性代数期末考试中已经考察过了，这部分可以自行看书。并且本章的大题都是一个类型，没有任何变化，所以这个大题也是千万不能失分的！

一、三角分解法求线性方程组的精确解

三角分解法主要包括:

Doolittle 分解与 *Cholesky* 分解,*Cholesky* 分解注意使用条件是：系数矩阵必须为n阶对称正定矩阵

1. *Doolittle* 分解解题步骤

对于线性方程组$Ax=b$

$$\begin{cases} ①A=LU,\text{即}\begin{pmatrix} a_{11} & a_{12} & \cdots & a_{1n} \\ a_{21} & a_{22} & \cdots & a_{2n} \\ \vdots & \vdots & & \vdots \\ a_{n1} & a_{2n} & \cdots & a_{nn} \end{pmatrix}=\begin{pmatrix} 1 & & & \\ l_{21} & 1 & & \\ \vdots & \vdots & \ddots & \\ l_{n1} & l_{n2} & \cdots & 1 \end{pmatrix}\begin{pmatrix} u_{11} & u_{12} & \cdots & u_{1n} \\ & u_{22} & \cdots & u_{2n} \\ & & \ddots & \vdots \\ & & & u_{nn} \end{pmatrix} \\ ②Ly=b\Rightarrow\text{求解}y \\ ③Ux=y\Rightarrow\text{求解}x \end{cases}$$

注意:由于三角分解求出的是方程的精确解，所以建议考试时算的时候认真一些，并且将算出来的结果带入题给方程组检验等式是否成立，这是不能失分的题！

前提：A可做唯一的LU分解的充要条件是A的各阶顺序主子式不为零

2. *Cholesky* 分解解题步骤

注意:$Cholesky$ 分解的使用条件:系数矩阵为对称正定矩阵!

对于线性方程组 $Ax=b$

$$\begin{cases} ①A=LL^T,\text{求解出}L\text{以及}L^T \\ ②Ly=b\Rightarrow \text{求解}y \\ ③L^T x=y\Rightarrow \text{求解}x \end{cases}$$

注意:同样记得带入题给方程组验算!

三、迭代法求线性方程组的近似解

注：迭代法求线性方程组的近似解在考试中出证明题的可能性比较大，部分期末考试试卷也有计算题出现，所以要求这部分需要对基础知识掌握得更为细致，之前的大题都是单一的考法，在这个部分考法可能会有很多种变化，所以这里简单总结相关知识点与易错点。

1. 矩阵范数与向量范数

矩阵范数与向量范数的相同点是两者都具有正定性、齐次性以及都满足三角不等式。

①向量范数:

对于向量 $\overline{x}=(x_1,x_2,\cdots,x_n)$

$$1-\text{范数：}\ \|\overline{x}\|_1=|x_1|+|x_2|+\cdots|x_n|$$

$$2-\text{范数：}\ \|\overline{x}\|_2=\sqrt{x_1^2+x_2^2+\cdots x_n^2}$$

$$\infty-\text{范数：}\ \|\overline{x}\|_\infty=\max_{1\leqslant i\leqslant n}|x_i|$$

②矩阵范数

对于矩阵 $A=(a_{ij})_{n\times n}$

$$\|A\|_1=\max_{1\leqslant j\leqslant n}\sum_{i=1}^{n}|a_{ij}|\ (\text{矩阵的}1-\text{范数即为每一列元素绝对值之和的最大值})(\text{列范数})$$

$$\|A\|_\infty=\max_{1\leqslant i\leqslant n}\sum_{j=1}^{n}|a_{ij}|\ (\text{矩阵的}\infty-\text{范数即为每一行元素绝对值之和的最大值})(\text{行范数})$$

$$\|A\|_2=\sqrt{\lambda_{\max}(A^T A)}(\text{谱范数})$$

$$\|A\|_F=\sqrt{\sum_{i=1}^{n}\sum_{j=1}^{n}a_{ij}{}^2}(F-\text{范数，矩阵各个元素的平方和})$$

谱半径：矩阵各个特征值绝对值的最大值. $\rho(B)=\max\limits_{1\leqslant i\leqslant n}|\lambda_i|$

2. 线性方程组迭代法的构造思路简写

$$Ax=b\Leftrightarrow x=Bx+g\xrightarrow{\text{构造迭代格式}}x^{(k+1)}=Bx^{(k)}+g,(k=0,1,\cdots)$$

所以关键在于求出 B，g，下面给出简单迭代法收敛的条件

简单迭代法收敛的充要条件是 $\lim\limits_{k\to+\infty}B^k=0$，而 $\lim\limits_{k\to+\infty}B^k=0$ 的充要条件是谱半径小于 1. 综上所述，若 B 的谱半径小于 1(谱半径小于 1 即 B 的所有特征值的绝对值均小于 1，解题时求出特征值判断绝对值是否小于 1 即可)，则该迭代格式

对于任意的$x^{(0)}$都收敛。反之，谱半径大于等于 1 时候，有可能存在某个$x^{(0)}$使得该迭代法收敛。

3. 迭代法题目求解步骤：

两种迭代格式的求解步骤如下（其实就是求 B 和 g）：

$A = L + D + U$，即(分别对应)

$$\begin{pmatrix} a_{11} & a_{12} & \cdots & a_{1n} \\ a_{21} & a_{22} & \cdots & a_{2n} \\ \vdots & \vdots & & \vdots \\ a_{n1} & a_{2n} & \cdots & a_{nn} \end{pmatrix} = \begin{pmatrix} 0 & & & \\ a_{21} & 0 & & \\ \vdots & \vdots & \ddots & \vdots \\ a_{n1} & a_{2n} & \cdots & 0 \end{pmatrix} + \begin{pmatrix} a_{11} & & & \\ & a_{22} & & \\ & & \ddots & \\ & & & a_{nn} \end{pmatrix} + \begin{pmatrix} 0 & a_{12} & \cdots & a_{1n} \\ & 0 & \cdots & a_{2n} \\ & & \ddots & \vdots \\ & & & 0 \end{pmatrix}$$

则对于$Jacobi$迭代：

$$B_J = -D^{-1}(L+U), g_J = D^{-1}b$$

对于基于$Jacobi$迭代的$Gauss-Seidel$迭代(JGS)：

$$B_{JGS} = -(D+L)^{-1}U, g_{JGS} = (D+L)^{-1}b$$

按题目要求的方法求出 B 之后若要求判断是否对于任意初始向量均收敛，则求出 B 的所有特征值判断其绝对值是否大于 1，若存在绝对值大于 1 的特征值则不是对于任意初始向量均收敛；若特征值绝对值均小于 1，则对任意初始向量均收敛

第四章 函数插值

本章在考试中也是有一道大题，并且形式比较固定——**给定一组自变量因变量取值，要求构造插值函数并求其他某一个点的近似解，**需要掌握的有$Lagrange$和$Newton$插值法、$Hermite$插值法，其中极大概率是$Lagrange$和$Newton$插值法，小概率为$Hermite$插值法，**最多**考到三次多项式(对四个点进行插值)后续以四个点为例

对于插值法的一些个人理解：

插值是一种数学方法，其基本思想是在已知的数据点之间构造一个连续的函数，以便能够通过这个函数来预测任意点的函数值。插值的精髓在于通过已知数据点的函数值，构造出一个连续的函数，并使用这个函数来预测其他点的函数值。因此我们可以归纳插值的基本步骤：

1. 给定一组已知的数据点，包括自变量因变量取值。
2. 构造一个连续的函数，该函数在这些数据点上的函数值与已知数据点的函数值相同。
3. 利用这个连续的函数来预测其他点的函数值。

因此可以知道，插值法的关键点是函数的构造，并且对于这个函数，我们有一个最基本的要求——连续性，在满足这个基本条件的前提下，衍生出了各种各样的插值方法，比如只考虑连续性的情况下，有$Lagrange$和$Newton$插值法，对于精度更高的场景，还需要在考虑连续性的情况下考虑光滑度(考虑导数值)，考虑一阶导数的有$Hermite$插值等等。

一、*Lagrance* 插值法解题步骤(以四点插值为例)

$$\begin{cases} \text{①写出}\,Lagrance\,\text{插值多项式:}L_3(x)=l_0(x)y_0+l_1(x)y_1+l_2(x)y_2+l_3(x)y_3 \\ \text{②根据插值多项式系数公式计算}\begin{cases} l_0(x)=\dfrac{(x-x_1)(x-x_2)(x-x_3)}{(x_0-x_1)(x_0-x_2)(x_0-x_3)} \\ l_1(x)=\dfrac{(x-x_0)(x-x_2)(x-x_3)}{(x_1-x_0)(x_1-x_2)(x_1-x_3)} \\ l_2(x)=\dfrac{(x-x_0)(x-x_1)(x-x_3)}{(x_2-x_0)(x_2-x_1)(x_2-x_3)} \\ l_3(x)=\dfrac{(x-x_0)(x-x_1)(x-x_2)}{(x_3-x_0)(x_3-x_1)(x_3-x_2)} \end{cases} \\ \text{③代入插值多项式，并代入题目代求值计算近似值} \\ \text{④误差估计：误差估计式}R_n(x)=f(x)-L_n(x)=\dfrac{f^{(n+1)}(\xi)}{(n+1)!}\omega_{n+1}(x),\text{其中}\,\omega_{n+1}(x)=(x-x_0)(x-x_1)\cdots(x-x_n) \end{cases}$$

注：

$f^{(n+1)}(\xi)$ 中的 $\xi\in[a,b]$，因此其值不确定，在误差估计的时候需要求的是 $f^{(n+1)}(x)$ 在 $[a,b]$ 上的最大值 $\max[f^{(n+1)}]$,误差估计则使用放缩， $R_n(x)=\dfrac{f^{(n+1)}(\xi)}{(n+1)!}\omega_{n+1}(x)\leqslant\dfrac{\max[f^{(n+1)}]}{(n+1)!}\omega_{n+1}(x)$,

Newton 插值法的误差估计同理

二、*Newton* 插值法解题步骤(以四点插值为例)

$$\begin{cases} \text{①写出}\,Newton\,\text{插值多项式:}N_3(x)=f[x_0]+f[x_0,x_1](x-x_0)+f[x_0,x_1,x_2]\omega_2(x)+f[x_0,x_1,x_2,x_3]\omega_3(x) \\ \text{②画差商表，计算差商} \\ \text{③差商计算结果代入插值多项式，并代入题目代求值计算近似值} \\ \text{④误差估计：误差估计式}R_n(x)=f(x)-L_n(x)=\dfrac{f^{(n+1)}(\xi)}{(n+1)!}\omega_{n+1}(x),\text{其中}\,\omega_{n+1}(x)=(x-x_0)(x-x_1)\cdots(x-x_n) \end{cases}$$

且对于差商

1. 差商计算公式

$$f[x_0]=f(x_0),f[x_0,x_1]=\frac{f(x_1)-f(x_0)}{x_1-x_0},f[x_0,x_1,x_2]=\frac{f[x_1,x_2]-f[x_0,x_1]}{x_2-x_0},f[x_0,x_1,x_2,x_3]=\frac{f[x_1,x_2,x_3]-f[x_0,x_1,x_2]}{x_3-x_0}$$

2. 差商性质：

①对称性： $f[x_i,x_j,x_k]=f[x_j,x_k,x_i]=f[x_j,x_i,x_k]=\cdots$

② $f(x)$ 在 $[a,b]$ 上 n 阶导数存在，则有 $f[x_0,x_1,\cdots,x_n]=\dfrac{f^{(n)}(\xi)}{n!},\xi\in[a,b]$

③ $f[x_0,x_1,\cdots,x_k]=\displaystyle\sum_{i=0}^{k}\frac{f(x_i)}{\omega'_{k+1}(x_i)},k=1,2,\cdots,n$

注意： *Newton* 插值法解题步骤前两步是关键，插值多项式的形式与差商表差商值一定要算对. 后两步与 *Lagrance* 同理.

三、*Hermite* 插值

注： *Hermite* 插值 是一种通过已知数据点的函数值和导数值来构造连续函数的插值方法。其基本思想是在已知数据点的基础上，通过构造一些辅助函数来确定插值函数。正式由于他的特点，所以本文不针对 *Hermite* 插值 写具体的单一

的解题步骤，因为没有固定的方法，需要根据题目所给函数值的特征进行具体的辅助函数的构造，可以参考教材 73 页例题 4.3 与 4.4 的两种典型解法。
这里给出三种典型的插值方法：
方法一：待定系数法，具体步骤如下：

$$\begin{cases} ①\text{考虑满足函数值的}Newton\text{插值多项式}:N_2(x)=f[x_0]+f[x_0,x_1](x-x_0)+f[x_0,x_1,x_2]\omega_2(x) \\ ②\text{再考虑导数的关系：设}H_3(x)=N_2(x)+K(x-x_0)(x-x_1)(x-x_2) \\ ③H_3(x)\text{求导并将导数值条件代入求得待定系数}K\text{即可} \end{cases}$$

注：这种方法的适用条件是题目所给条件只有一个导数值相等的条件，因为可以看出，由 *Newton* 插值所得到的插值多项式只有一个待定系数，如果导数值条件多余一个则会使得方程组变为超定方程组，无法求解
方法二：建立带有重节点的差商表

x	$f(x)$	一阶差商	二阶差商	三阶差商
x_0	$f(x_0)$			
x_0	$f(x_0)$	$f[x_0,x_0]$		
x_1	$f(x_1)$	$f[x_0,x_1]$	$f[x_0,x_0,x_1]$	
x_2	$f(x_2)$	$f[x_1,x_2]$	$f[x_0,x_1,x_2]$	$f[x_0,x_0,x_1,x_2]$

其中 $f[x_0,x_0]=f'(x_0)$（考虑差商定义式的极限），即可得

$$N_3(x)=f[x_0]+f[x_0,x_0](x-x_0)+f[x_0,x_0,x_1](x-x_0)(x-x_0)+f[x_0,x_0,x_1,x_2](x-x_0)^2(x-x_1)$$

注：可用于多个导数值条件下的插值求解
方法三：使用插值基函数进行构造（教材例题 4.4）（使用最麻烦的一种）
注：*Hermite* 插值几乎没考过，推荐把重心放在前两种插值方法
想要掌握更加牢固，*Lagrance* 和 *Newton* 插值的推导过程建议理解掌握

第五章 曲线拟合的最小二乘法

本章也会涉及到一道大题并且形式比较单一——给定一组数据，要求使用最小二乘法将数据进行拟合，并且通常会给出一个经验公式（求解经验公式中的待定系数即可）

关于曲线拟合的一些个人理解：

曲线拟合和函数插值的基本思想都是通过已知数据点来描述一个函数，并利用这个函数来预测其他未知点的函数值。但是函数插值得到的近似函数必须要过插值点，而曲线拟合只需要保证拟合函数与已知点相近即可，而对于相近的界定就是多种多样的了在进行曲线拟合时，通常会根据实际问题的特点选择适当的拟合函数形式，并利用统计学方法来求解拟合曲线的系数。常见的拟合算法包括最小二乘法、局部加权回归、非线性最小二乘法、贝叶斯拟合等。本章学习的就是曲线拟合的最小二乘法
曲线拟合的最小二乘法的精髓是将拟合问题转化为一个最小化误差的优化问题，从而得到最优的拟合曲线。
引入：若 $Ax=b$ 为超定方程组（未知数个数少于方程个数，即无解，从而产生矛盾，故又叫矛盾方程组），则它的正规方程组为

$$Cx=A^TAx=A^Tb=d$$

此时正规方程组有唯一解，这个解便是矛盾方程组的**最小二乘解**（误差最小的一组解）

拟合问题解题步骤

①确定拟合函数的形式(非常关键，但是题目会给)，通常为多项式函数、指数函数、对数函数等

②确定ϕ_0，$\phi_1,\cdots,\phi_k$(明确写在试卷上，防止算错)

③根据题目条件给出$A=\begin{pmatrix}\phi_0(x_0) & \phi_1(x_0) & \cdots & \phi_k(x_0)\\ \phi_0(x_1) & \phi_1(x_1) & \cdots & \phi_k(x_1)\\ \vdots & \vdots & & \vdots\\ \phi_0(x_n) & \phi_1(x_n) & \cdots & \phi_k(x_n)\end{pmatrix}$,并求解$A^TAx=A^Tb$，解出拟合曲线的待定系数

注:1. 关于拟合函数形式的问题，细看教材 87 页下方两种处理方式与技巧

在许多实际问题中，待拟合的曲线并不一定是一组已知函数的线性组合，比如随年份变化的人口模型为一种指数模型 $y=ae^{bx}$. 在拟合这类问题时，一般应将关于参数的非线性关系线性化.

如对于指数模型 $y=ae^{bx}$，通过两边取自然对数得到

$$\ln y=\ln a+bx.$$

这样 $\ln y$ 是函数 1 和 x 的线性组合，组合系数为 $\ln a$ 和 b.

再如 $y=\frac{ax}{b+x}$，可以将形式改写为

$$\frac{1}{y}=\frac{1}{a}+\frac{b}{a}\frac{1}{x}.$$

这样 $\frac{1}{y}$ 就成了 1 和 $\frac{1}{x}$ 的线性组合，组合系数为 $\frac{1}{a}$ 和 $\frac{b}{a}$.

§5.3 曲线最小二乘拟合 87

2. 步骤③具体如下(熟记)

正规方程组如下

$$\begin{pmatrix}(\phi_0,\phi_0) & (\phi_0,\phi_1) & \cdots & (\phi_0,\phi_m)\\ (\phi_1,\phi_0) & (\phi_1,\phi_1) & \cdots & (\phi_1,\phi_m)\\ \vdots & \vdots & & \vdots\\ (\phi_m,\phi_0) & (\phi_m,\phi_1) & \cdots & (\phi_m,\phi_m)\end{pmatrix}\begin{pmatrix}c_0\\ c_1\\ \vdots\\ c_m\end{pmatrix}=\begin{pmatrix}(\phi_0,b)\\ (\phi_1,b)\\ \vdots\\ (\phi_m,b)\end{pmatrix}$$

其中

$$(\phi_i,\phi_j)=\sum_{i=0}^{n}\phi_k(x_i)\phi_j(x_i)\qquad(\phi_i,b)=\sum_{i=0}^{n}\phi_k(x_i)b$$