选择题目：Quadratic Optimization Assignment（选题 4 中的第 3 题 QP Assignment 部分）

1. 问题分析与模型建立（第一问回答）：

这个问题和实际的学校登记有关，其中原题第一问要求建立离散时间的（按照年份记载的）学生注册登记的地推关系式，以及毕业生的计算式子，只有建立了这些关系式子，才能在后面进行转化成优化目标函数，从而进行优化求解。所以，构建正确的递推关系式，对于后面的优化设计非常重要，同时，对于题目中给出的表格数据，要将其与实际的各个变量值对应，从而方便后面的分析。

所以这一部分主要是建立递推式模型，回答题目中的第一问；同时对于表格中的数据进行分析，找出其对应的题目中的具体变量。

题目中给出了每一年学生登记数量、学生毕业升学、毕业工作、退学以及其他学生流动的关系，如图 1 所示，其中要注意的是 b_4, c_4 这两个分支，学生在读完硕士或者博士以后，又转而学习学士或者硕士，这是比较少见的，对应在现实中应该是类似于转专业的特殊情况，依据实际情况，这一部分的数值应该是比较小的，可以作为对于优化最终求解结果的验证。

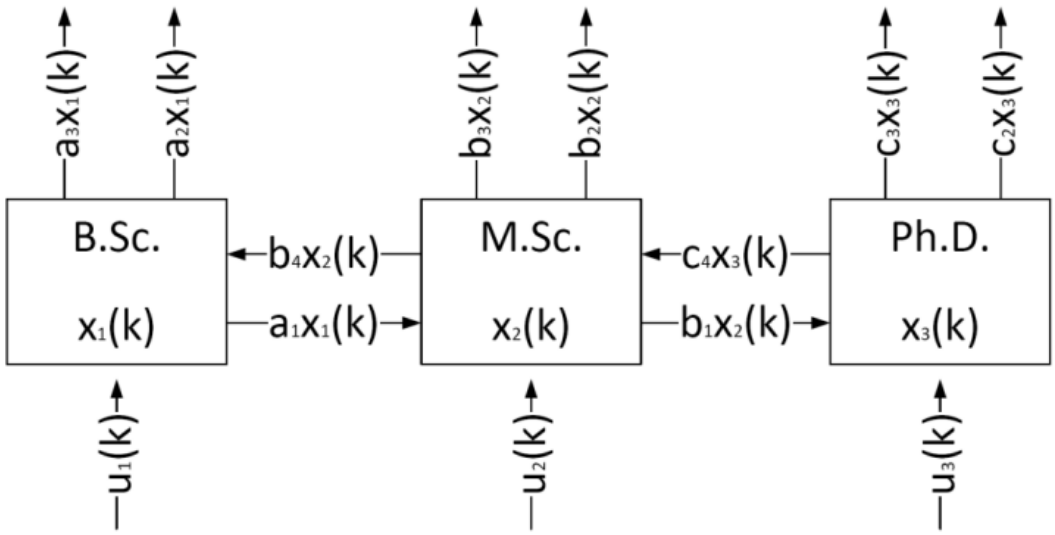

图 1: 题目中的各个学位同学之间的流动关系

根据题目中的描述，以及图 1 中的人员流动，可以构建如下的方程组：

$$\begin{cases} x_1(k+1) = x_1(k) + u_1(k) - (a_1 + a_2 + a_3)x_1(k) + b_4 x_2(k) \\ x_2(k+1) = x_2(k) + u_2(k) - (b_1 + b_2 + b_3 + b_4)x_2(k) + a_1 x_1(k) + c_4 x_3(k) \\ x_3(k+1) = x_3(k) + u_3(k) - (c_2 + c_3 + c_4)x_3(k) + b_1 x_2(k) \end{cases}$$

从方程组中可以看出，实际过程中的三个变量 $x_1(k), x_2(k), x_3(k)$ 的更新方程中，实际上是三个变量之间相互作用的，单独的对于一个变量的参数更新而忽略其他的变量是有问题的，所以，对于接下来的建模

中，需要考虑的是整个三个变量的所有参数的估计，这一块将在接下来的优化问题分析部分中详细展示。同时，对于输出部分，要求计算出每一年的每一个学位的毕业的人数，根据题目意思，具体如下式所示：

$$\begin{cases} y_1(k+1) = a_1 x_1(k) + a_2 x_1(k) \\ y_2(k+1) = b_1 x_2(k) + b_2 x_2(k) + b_4 x_2(k) \\ y_3(k+1) = c_2 x_3(k) + c_4 x_3(k) \end{cases}$$

同时，由于其中的参数 D_1, D_2, D_3 已经给出，所以实际上 a_3, b_3, c_3 这三个和退学有关的参数实际上是可以直接给定的，这对于后面的参数估计是减轻了负担的，具体如下所示：

$$\begin{cases} a_3 = 0.05 + \frac{D_1}{200}, D_1 = 3 \\ b_3 = 0.05 - \frac{D_2}{200}, D_2 = 4 \\ c_3 = 0.10 + \frac{D_3}{200}, D_3 = 5 \end{cases}$$

对于每一部分的递推关系式子已经确定，题目中还提供了表格的部分，对于表格中的数据，需要对应实际的变量，这一部分有一点争议，因为对于 current 那一栏的数据，难以确定到底是每一年的年初进行登记，还是每一年的年底进行登记，因为，如果是年初登记的话，即此时的 current 看成是 $x(k)$，而同时每一行的新生、毕业的数据看成 $u(k), x(k)$ 乘上相关的参数，但是这样计算的话，带入 09 年，10 年的数据会算出来 b_4 会出现负值的情况，这样显然是不对的；但是如果看成是年底登记的话，此时的 current 看成是 $x(k+1)$，而同时每一行的新生、毕业的数据看成 $u(k), x(k)$ 乘上相关的参数，这样计算的话，带入 09 年，10 年的数据是没有问题的，但是带入后面年份的数据就是非常离谱的出现 b_4 的负值的，所以这里的表格数据可能是有问题的。

这一部分后来和身边同样选择这一题的同学讨论，最终还是商量之后决定不管 b_4, c_4 的计算值，将此时的 current 看成是 $x(k+1)$，而同时每一行的新生、毕业的数据看成 $u(k), x(k)$ 乘上相关的参数，也就是说，表格中的第 k 年的 current 对应的是 $x_1(k+1)/x_2(k+1)/x_3(k+1)$，也就说，认为这是登记的是第 k 年年底的学生人数，已经经过了第 k 年中的新生入学、老生毕业、退学这一系列的事件；而表格中第 k 年的 New 对应的是 $u_1(k)/u_2(k)/u_3(k)$，Graduated 分别对应的是 $(a_1+a_2)x_1(k)/(b_1+b_2+b_4)x_2(k)/(c_2+c_4)x_3(k)$。具体如下表所示：

time/degree	B.Sc.	M.Sc.	Ph.D.
current	$x_1(k+1)$	$x_2(k+1)$	$x_3(k+1)$
new	$u_1(k)$	$u_2(k)$	$u_3(k)$
Graduated	$(a_1+a_2)x_1(k)$	$(b_1+b_2+b_4)x_2(k)$	$(c_2+c_4)x_3(k)$

2. 优化问题分析

这一部分是在原来的建模的基础上，对于上面的地推式子的深入转换，能够更加深入准确地回答第一问中的递推关系式的问题，使其能够成为优化模型的一部分；同时，这一部分将给出具体的优化目标函数和约束条件，为后面的求解做好准备。

对于上面的递推关系式，将其用矩阵的形式表示，设 $X(k) = [x_1(k), x_2(k), x_3(k)]^T, U(k) = [u_1(k), u_2(k), u_3(k)]^T$，这样的话，递推式可以表示为：

$$X(k+1) = AX(k) + U(k)$$

用矩阵的形式，进一步表示可以写为：

$$X(k+1) = [A, I] \times \begin{bmatrix} X(k) \\ U(k) \end{bmatrix}$$

其中 I 矩阵是一个单位矩阵，其中的 A 矩阵如下所示：

$$A = \begin{bmatrix} 1 - a_1 - a_2 - a_3 & b_4 & 0 \\ a_1 & 1 - b_1 - b_2 - b_3 - b_4 & c_4 \\ 0 & b_1 & 1 - c_2 - c_3 - c_4 \end{bmatrix}$$

这一部分其实更加深入的回答了第一问的递推关系式，使用矩阵的形式，这样更加简洁明了，也为后面的优化函数做好准备。

对于优化的目标函数，这是一个参数估计问题，这里使用上课讲的参数估计的 ARX 模型 (Auto Regressive eXogenous input)，这个模型具体可以见老师上课讲的 QP 部分的 PPT 的第九页和第十页。在这个题目里面，我们将上面的矩阵 X(k)，U(k) 作为输入，计算 $X(k+1)$ 和 $AX(k) + U(k)$ 之间的误差，并且让误差最小化，寻找使得误差最小化的最优参数，这是一个 QP 求解的问题。这里的误差，将使用所有数据求和得到的累计的误差，类似于现在深度学习中的计算所有的训练样本的累计误差，将这个累计误差作为目标函数进行优化。具体的计算如下：

设矩阵 $B = [A, I]$，矩阵 $h(k) = \begin{bmatrix} X(k) \\ U(k) \end{bmatrix}$，这样的话递推关系式可以写作：

$$X(k+1) = B \times h(k)$$

设第 k 组数据的误差为：

$$e(k) = X(k+1) - B \times h(k)$$

计算误差的二范数平方为：

$$e(k)^T e(k) = X(k+1)^T X(k+1) - X(k+1)^T Bh(k) - h(k)^T B^T X(k+1) + h(k)^T B^T Bh(k)$$

对于误差的求和为：

$$\sum_{k=1}^{8} e(k)^T e(k) = \sum_{k=1}^{8} X(k+1)^T X(k+1) - X(k+1)^T Bh(k) - h(k)^T B^T X(k+1) + h(k)^T B^T Bh(k)$$

注意这里实际上能用的年份的数据只有 8 组，因为 2009 年和 2018 年的数据实际上是残缺的，同时这个是对于参数的优化问题，所以上面的损失函数还要转化成一个是以参数为变量的目标函数，这一部分的工作量较大，但是这里只要保留和参数有关的多项式就好，可以省略很多无关的多项式，所以没有想象中那么繁琐。

将原来的损失函数转化成关于参数的优化目标函数，设 $x = [a_1, a_2, b_1, b_2, b_4, c_2, c_4]$，经过转化之后的目标函数如下所示：

$$\min_x \frac{1}{2} x^T Hx + 2c_1 + c_2$$

$$Dx \leq b$$

$$lb \leq x \leq ub$$

其中对于矩阵 H，矩阵 c_1, c_2 是转化之后的参数矩阵，其具体表示为：

$$H = 2\begin{bmatrix} 2x_1(k)^2 & x_1(k)^2 & -x_1(k)x_2(k) & -x_1(k)x_2(k) & -2x_1(k)x_2(k) & 0 & x_1(k)x_3(k) \\ x_1(k)^2 & x_1(k)^2 & 0 & 0 & -x_1(k)x_2(k) & 0 & 0 \\ -x_1(k)x_2(k) & 0 & 2x_2(k)^2 & x_2(k)^2 & x_2(k)^2 & -x_2(k)x_3(k) & -2x_2(k)x_3(k) \\ -x_1(k)x_2(k) & 0 & x_2(k)^2 & x_2(k)^2 & x_2(k)^2 & 0 & -x_2(k)x_3(k) \\ -2x_1(k)x_2(k) & -x_1(k)x_2(k) & x_2(k)^2 & x_2(k)^2 & 2x_2(k)^2 & 0 & 0 \\ 0 & 0 & -x_2(k)x_3(k) & 0 & 0 & x_3(k)^2 & x_3(k)^2 \\ x_1(k)x_3(k) & 0 & -2x_2(k)x_3(k) & -x_2(k)x_3(k) & 0 & x_3(k)^2 & 2x_3(k)^2 \end{bmatrix}$$

$$c_1^T = \begin{bmatrix} x_1(k)x_1(k+1) - x_1(k)x_2(k+1) \\ x_1(k)x_1(k+1) \\ x_2(k)x_2(k+1) - x_2(k)x_3(k+1) \\ x_2(k)x_2(k+1) \\ x_2(k)x_2(k+1) - x_2(k)x_1(k+1) \\ x_3(k)x_3(k+1) \\ x_3(k)x_3(k+1) - x_3(k)x_2(k+1) \end{bmatrix}$$

$$c_3^T = \begin{bmatrix} -1.87x_1(k)^2 - 2u_1(k)x_1(k) + 2x_1(k)u_2(k) \\ -1.87x_1(k)^2 \\ -2u_2(k)x_2(k) - 1.94x_2(k)^2 + 2x_2(k)u_3(k) \\ -2u_2(k)x_2(k) - 1.94x_2(k)^2 \\ 1.87x_1(k)x_2(k) + 2x_2(k)u_1(k) - 2x_2(k)u_2(k) - 1.94x_2(k)^2 \\ -2x_3(k)u_3(k) - 1.75x_3(k)^2 \\ 2x_3(k)u_2(k) - 2x_3(k)u_3(k) - 1.75x_3(k)^2 \end{bmatrix}$$

$$D = \begin{bmatrix} 1 & 1 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & 1 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 & 1 \end{bmatrix}$$

对于约束部分的 b、lb、ub 矩阵，这里的参数需要根据实际情况进行设置，具体的将在求解部分进行阐述。

3. 优化问题求解（第二问回答）

这一部分是对第二部分构建的优化目标函数进行求解，得到一组比较好的参数估计，其中，对于模型的约束进行了改进，通过计算毕业率均值的方式，确定大致的约束范围。由于目标函数的最优解并不是在约束范围内的，这里智能得到一组较好的可行解。将使用三种方法进行求解：内点法、有效集法、序列二次规划法（Sequential Quadratic Programming，简称 sqp 法），三种方法的结果和迭代步骤将会展示。

对于目标函数，在没有约束的情况下，其最优解不在这个问题的约束范围之内，也就是说，这个问题本质上是在约束范围内寻找一组比较好的可行解，而单纯的二次规划的最优解在这里没有实际意义。对于约束部分的 b、lb、ub 矩阵，这里使用毕业率的平均值来估计其上限。有第一部分的对于表格的分析可以得知，将毕业人数除以那一年的年初的登记人数（上一年年末的登记人数）$x_1(k)/x_2(k)/x_3(k)$，实际上可以得到的是 $(a_1+a_2)/(b_1+b_2+b_3)/(c_2+c_4)$ 在第 k 年的一组值，这里将每一年的 $(a_1+a_2)/(b_1+b_2+b_3)/(c_2+c_4)$ 看成随机变量，可以估计这个随机变量的均值方差，通过计算可以得出 $(a_1+a_2)/(b_1+b_2+b_3)/(c_2+c_4)$

的均值分别为 0.30/0.46/0.11。这样的话，可以将这一组均值，再在均值的基础上加上一些可以调整的数值（这些数值可以看成超参数），合适的调整这些超参数，由此就可以确定 b 矩阵，从而得到比较好的解。同理，对于上下界 lb，ub，都可以看成超参数来进行调整，其中上界 ub 可以全部设置成 1，因为按照实际意义，学生流动的比率不会超过 1，对于下界，要注意的是，下界不能全部设成 0，这样的话会有平凡解，下界可以根据求解的情况和实际意义进行调整，从而最终确定下界 lb。

关于求解器的选择，这里的求解器我们选择内点法、有效集法和序列二次规划法，单纯形法由于这里参数实在是太多了，计算量实在太大，而且 matlab 的单纯形法求解器和上课讲的求解器之间是有矛盾的，这一点在当时的作业中是体现出来的，所以这里只是考虑上面三种求解器，三种求解器都是考虑迭代十次，三种求解器都收敛到了相同的最优解。具体的求解迭代如下表所示：
内点法：

time	a_1	a_2	b_1	b_2	b_4	c_2	c_4
1	0.0400	0.2800	0.0700	0.4100	0.0700	0.2033	0.0133
2	0.0401	0.2799	0.0700	0.4100	0.0700	0.2033	0.0133
3	0.0978	0.2509	0.1022	0.3984	0.0598	0.2033	0.0134
4	0.0978	0.2509	0.1022	0.3984	0.0598	0.2033	0.0134
5	0.0978	0.2509	0.1022	0.3984	0.0598	0.2033	0.0134
6	0.0978	0.2509	0.1022	0.3984	0.0598	0.2033	0.0134
7	0.0978	0.2509	0.1022	0.3984	0.0598	0.2033	0.0134
8	0.0978	0.2509	0.1022	0.3984	0.0598	0.2033	0.0134
9	0.0978	0.2509	0.1022	0.3984	0.0598	0.2033	0.0134
10	0.0978	0.2509	0.1022	0.3984	0.0598	0.2033	0.0134

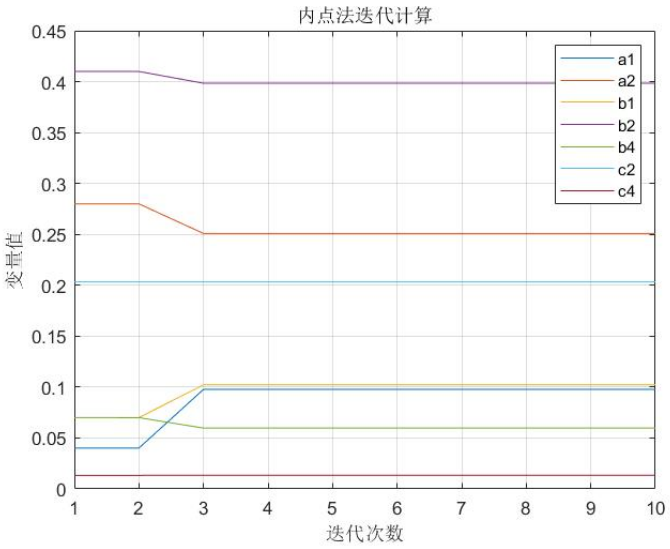

图 2: 内点法的迭代过程

有效集法：

time	a_1	a_2	b_1	b_2	b_4	c_2	c_4
1	0.0400	0.2800	0.0700	0.4100	0.0700	0.2033	0.0133
2	0.0401	0.2799	0.0700	0.4100	0.0700	0.2033	0.0133
3	0.0978	0.2509	0.1022	0.3984	0.0598	0.2033	0.0134
4	0.0978	0.2509	0.1022	0.3984	0.0598	0.2033	0.0134
5	0.0978	0.2509	0.1022	0.3984	0.0598	0.2033	0.0134
6	0.0978	0.2509	0.1022	0.3984	0.0598	0.2033	0.0134
7	0.0978	0.2509	0.1022	0.3984	0.0598	0.2033	0.0134
8	0.0978	0.2509	0.1022	0.3984	0.0598	0.2033	0.0134
9	0.0978	0.2509	0.1022	0.3984	0.0598	0.2033	0.0134
10	0.0978	0.2509	0.1022	0.3984	0.0598	0.2033	0.0134

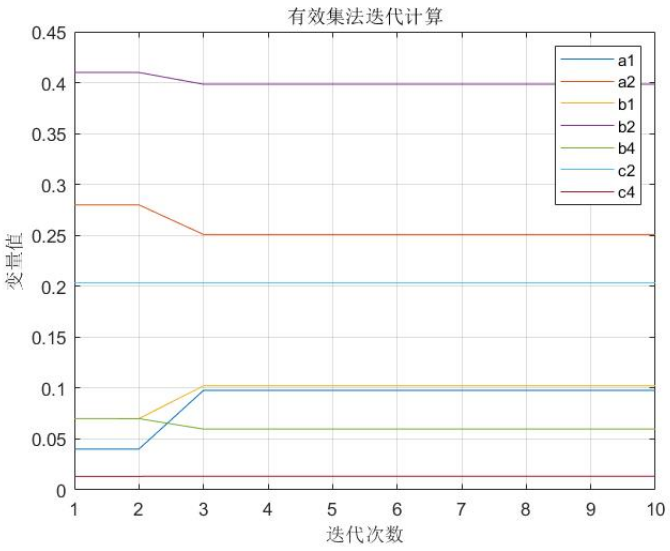

图 3: 有效集法的迭代过程

SQP 法：

time	a_1	a_2	b_1	b_2	b_4	c_2	c_4
1	0.0400	0.2800	0.0700	0.4100	0.0700	0.2033	0.0133
2	0.0401	0.2799	0.0700	0.4100	0.0700	0.2033	0.0133
3	0.0978	0.2509	0.1022	0.3984	0.0598	0.2033	0.0134
4	0.0978	0.2509	0.1022	0.3984	0.0598	0.2033	0.0134
5	0.0978	0.2509	0.1022	0.3984	0.0598	0.2033	0.0134
6	0.0978	0.2509	0.1022	0.3984	0.0598	0.2033	0.0134
7	0.0978	0.2509	0.1022	0.3984	0.0598	0.2033	0.0134
8	0.0978	0.2509	0.1022	0.3984	0.0598	0.2033	0.0134
9	0.0978	0.2509	0.1022	0.3984	0.0598	0.2033	0.0134
10	0.0978	0.2509	0.1022	0.3984	0.0598	0.2033	0.0134

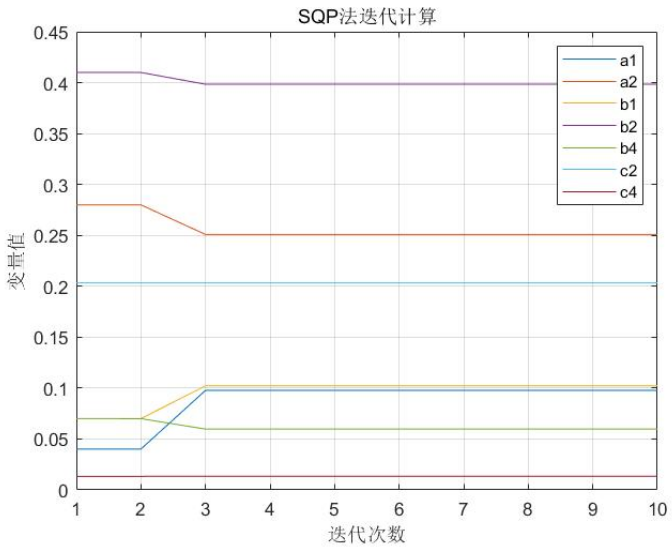

图 4: SQP 法的迭代过程

有意思的是，这里的单重优化方法的输出是完全一致的，但是在调试参数的过程中，使用三种方法的输出并不是一致的，而且最终的收敛的值也有 0.01 左右的误差的。但是在最终确定了上下界和约束的超参数的时候，发现竟然收敛的计算式一模一样的，这里猜想大致原因，由于没有设置初始值，所以在 matlab 里面程序自动设置，从而三种方法初始值一致，所以导致最后的收敛效果也接近一致。三种方法都是求解出来了一致的解，都是在很小的步骤以内收敛，但是注意的是，这三种方法求出来的都是在约束情况下的解，和这个二次规划的无约束下的最优解是有很大的差距的，所以这里的求解实际上是一个约束下的近似最优解，对于整个二次规划的函数来说是一个次优解，或者说是一个局部的最优解，用来解决这里的实际中的问题。

选择的这三种方法都可以比较好的解决在约束下的最优的求解问题，而且这里三种方法的输出结果完全一致，都是收敛到了相同的解，最终的解：

$$x = [a_1, a_2, b_1, b_2, b_4, c_2, c_4] = [0.0978, 0.2509, 0.1022, 0.3984, 0.0598, 0.2033, 0.0134]$$

可以看到这一组解相对而言比较符合实际情况，而且对于后面的估计效果也相对较好，具体的估计效果在下一部分展示。

4. 学生登记人数估计与分析（第三问回答）

这一部分主要是回答第三问，对上一部分的求解出来的参数进行验证，并且对于验证效果进行分析。对于原题中的第三问进行作答，在只有 2009 年的数据的初始点的情况下，以及有每一年新生人数的情况下，对于每一年的登记人数进行估计，同时对于毕业生进行估计。

对于上面第一第二部分对于递推关系式的建模，结合表格中的数据，可以进行模拟每一年的学生人数，最终模拟出来的结果如下表所示。

year	B.Sc.-current	B.Sc.-new	M.Sc.-current	M.Sc.-new	Ph.D.-current	Ph.D.-new
2010	2837	976	746	504	277	47
2011	2814	989	688	418	298	60
2012	2801	981	661	386	304	64
2013	2806	976	642	370	306	66
2014	2822	978	632	360	308	66
2015	2848	984	640	354	308	66
2016	2908	993	646	358	308	66
2017	2868	1014	653	362	311	66
2018	-	1000	-	366	-	67

从估计的数值上看，对比原来的表格，可以发现估计和表格的实际差别在 100 以内，但是实际上差距还是不小的，尤其时候硕士和博士部分，只能说基于最小二乘法参数估计是一种折中（trade-off）的参数估计，是的各个方面的误差都不是太大，但是实际上还是不小的，说明题目中要求的基于 QP 的参数估计法并不是最优的参数估计方法，关于此将在下一部分讨论。

5. 结果分析

在第一部分提出，这里的表格的数据实在难以确定，所以第四题最终没有去做（原题要求至少做三道题），但是这一部分想探讨一下方法改进的思路，改进思路值给出猜想，不给出具体的细节，如果以后有空的话，可以深入的去改进方法。

这里的每一年的参数可以看做一个是和时间有关的变量，时间因素也应该考虑进去，也就是说，这里的 $x = [a_1, a_2, b_1, b_2, b_4, c_2, c_4]$ 应该看做一个和时间有关的变量 $x(k)$，如果这样的话，可以借鉴模式识别中的判别式模型的思路，可以将 $x(k)$ 设为关于时间的线性表达式子，然后在结合实际的数据对于表达式进行更新，这样的话，基于每一年的数据，参数就是一个动态变换的值，有着更加准确的预测效果。其难点在于如何确定参数表达式，这是一个问题，以及确定表达之后如何确定最佳参数，这样是比较麻烦的。

上面是借鉴了判别式模型的思路，也可以是借鉴模式识别中生成模型的思路，局域统计的思路来进行参数估计，可以将各个参数看做是一个随机变量，假设其服从的是高斯分布，可以根据极大似然估计或者贝叶斯估计来进行参数估计，事实上上面用到的求解均值的思路就是极大似然估计。深入思考的话，可以借鉴随机过程的思路，将参数看成和时间有关的随机变量，当然这样的估计会更加繁琐，求解出每一个参数的均值函数，然后使用均值函数进行参数估计。这一部分计算量更大，也更加繁琐，恐怕不是一个作业报告能够搞定的。

所以只能说是使用 QP 的方法是相对简单可行的，如果使用其他的参数估计的思路，那会比较繁琐和困难，远远超出了大作业的工作量了，所以这个题目要求使用的还是比较简单的 QP 思路。

6. 总结

本报告第一部分主要构建了递推关系式以及表格中数据的分析，回答了第一问；第二部分构建优化模型，深入回答了第一问并且为第二问做好准备；第三部分使用三种方法进行参数估计，并且展示三种方法的结果并且进行结果分析；第四部分主要是给出了人数的估计，回答了第三问，并且对于估计进行了分析；第五部分主要是对方法改进的思考（第四题因为表格数据疑问没有做）。回答了原问题的前三问，给出了比较好的参数估计结果，解决了问题。