《系统工程导论》

大作业

系统工程方法
在交通数据处理的应用

（独立完成）

姓名：卢志

班级：自 43 班

学号：2014011497

2016 年 5 月 11 日

目录

1. 实验大纲

1.1 实验名称

系统工程方法在交通数据处理的应用

1.2 实验类型

综合设计型实验

1.3 实验目的

利用系统工程方法对实际数据完成建模和分析，锻炼自学能力。

1.4 实验方法

提供北京市路网交通流原始数据，要求学生基于课堂讲授方法并自学其他方法，完成数据的建模与分析。

2. 实验内容

2.1 黑箱建模

2.1.1 内容要求

基于课堂讲授的黑箱建模方法，对上述数据进行预处理后，建立交通流预测模型，以最后两天的数据为预测值，之前的数据为训练值，给出分时段（5 分钟，10 分钟和 15 分钟）预测结果，并给出预测精度（平均绝对误差百分比，平均相对误差等指标）。

2.1.2 设计思路

a. 数据预处理

现有的数据为 50 个检测器 16 天（2006.10.22～2006.11.6）的数据，每天的数据点为 288 个（5 分钟一个数据）。相当于 50 个道口 16 天每天 288 个时段的数据。

而黑箱建模的作用是通过拟合，进行预测。那么可以假定 1 个道口的过车数据为 Y，用其他 49 个道口的数据作为 X，则可将入定义为一个 49 维的变量，输出定义为一个 1 维的变量。

而建模中所需的黑箱参数，需要通过前 14 天的数据进行训练得到，而将第 15 天和第 16 天进行预测。而在训练过程中，可以认为前 14 天的数据是等价的，所以可以将前 14 天的数据在每一个采集时刻取平均，得到了 50 个道口，每个道口 288 个数据，作为预处理后的数据。其中，有 1 个道口的数据记为目标值 Y，则 Y 为 1*288 的矩阵，把剩下 49 个道口的数据记为输入值 X，则 X 为 49*288 的矩阵。

b. 多元线性回归

数据预处理之后，得到了 X，Y 两个矩阵，可进行多元线性回归，即 n=49，N=288。

① 规范化自变量、因变量

按照定义，样本的均值和方差计算如下：

样本均值：$e(x_i)=\frac{1}{N}\sum_{i=1}^{N}x_i(t)$

样本方差：$\delta^2(x_i)=\frac{1}{N-1}\sum_{i=1}^{N}(x_i(t)-e(x_i))^2$

对应的归一化后数据 $\overline{x}_i(t)=\frac{x_i(t)-e(x_i)}{\sqrt{\delta^2(x_i)}}$，同理 $\overline{y}(t)=\frac{y(t)-e(y)}{\sqrt{\delta^2(y)}}$

② 判断问题是否病态

将上一步求取的规范化数据整理为（其中，输入 X 为 n 维向量，有 N 组数据）：

$$Y=\left[y(1)...y(N)\right]_{1\times N},X=\left[x(1)...x(N)\right]_{n\times N}$$

对矩阵进行分解：$XX^T=Q\Lambda Q^T,XY^T$，其中$\Lambda=diag(\lambda_1,\lambda_2...\lambda_n)$，即为$XX^T$全部的特征值，将特征值按照从大到小的顺寻排序。

设定阈值ε，当$\frac{\sum_{i=m+1}^{n}\lambda_i}{\sum_{i=1}^{n}\lambda_i}\geq\varepsilon$时，说明所需要保留的特征值只有(n-m)个，即为$\lambda_{m+1}...\lambda_n$。若m>0，则说明问题是病态回归问题，需要降维。本题要求阈值ε自定，本次作业设置ε=0.01。

③ 降维处理

若前述步骤说明需要降维，降维处理选择较小的n-m个特征值：

$$\lambda_{m+1}...\lambda_n$$

设XX^T从大到小特征值所对应的特征向量为：

$$Q=\left[q(1),q(2)...q(n)\right]$$

选出较大m个特征值对应特征向量组成：

$$Q_m=[q(1),q(2),\dots,q(m)]$$

则

$$(ZZ^T)^{-1}=\begin{bmatrix}1/\lambda_1 & 0 & 0\\ 0 & \dots & 0\\ 0 & 0 & 1/\lambda_m\end{bmatrix}$$

④ 病态回归方程求解

直接利用下述公式计算，其中$\bar{y},\bar{x}$都是经过了归一化处理后的结果：

$$\hat{d}=(ZZ^T)^{-1}ZY^T=\Lambda_m^{-1}Q_m^{\ T}XY^T$$

$$\hat{c}=Q_m\hat{d}$$

$$\bar{y}=\hat{c}^T\bar{x}$$

由上式可得到病态回归方程。

⑤ 反归一化

再将求得的回归参数代入原来的方程：

$$\frac{y(t)-e(y)}{\sqrt{\delta^2(y)}}=\sum \hat{c}\frac{x_i(t)-e(x_i)}{\sqrt{\delta^2(x_i)}}$$

$$\hat{b}=\delta(y)\sum_{i=1}^{n}\frac{c_i}{\delta x_i}, a=e(y)-\delta(y)\sum_{i=1}^{n}\frac{e(x_i)}{\delta(x_i)}$$

即可求得最终参数。

⑥ 显著性检验

利用 F 检验法进行统计性检验，根据讲义，统计量

$$F=\frac{ESS/f_E}{RSS/f_R}$$

服从自由度为(n,N-n-1)的 F 分布，对于给定的显著性水平 alpha，本次建模中选取 alpha=0.01，可查 F 分布表，做假设检验。可利用 finv 函数，用法如下：

x=finv(p,v1,v2)。分子自由度为 v1，分母自由度为 v2，求置信度（显著性水平）为 1-p 的情况下得到的 F 的值即为 x。此时分子为解释平方和，自由度 1；分母为剩余平方和，自由度为 N-2（N 为数据总数）。

⑦ 预测精度（求取置信区间）

F检验的结果为x与y存在线性关系，因此求取置信区间。S_δ 为y的剩余均方差，表示变量y偏离回归直线的误差：

$$S_\delta=\sqrt{\frac{\sum_{i=1}^{N}(y_i-\hat{y})^2}{N-2}}=\sqrt{\frac{RSS}{f_R}}$$

S_δ服从正态分布，可用norminv函数求取置信区间。norminv函数用法如下：

x=norminv(p,u,sigma)。norminv是正态分布的反函数，已知概率求百分位点（方差前的系数）。P为概率，u为正态分布的均值，sigma为正态分布的方差。由于本次中使用的是标准正态分布，故u=0，sigma=1。语句为：

z=norminv(1-alpha/2,0,1)%即得到正数的z进行置信区间计算。

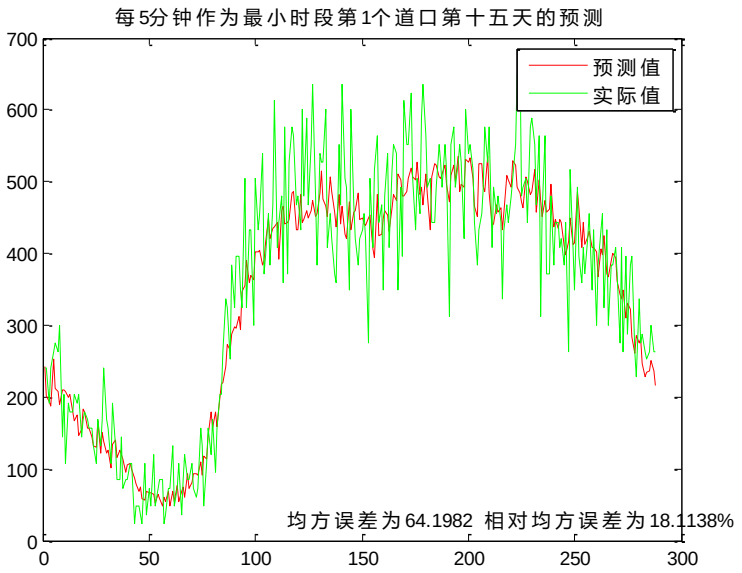
c. 分时段预测结果（5 分钟、10 分钟）

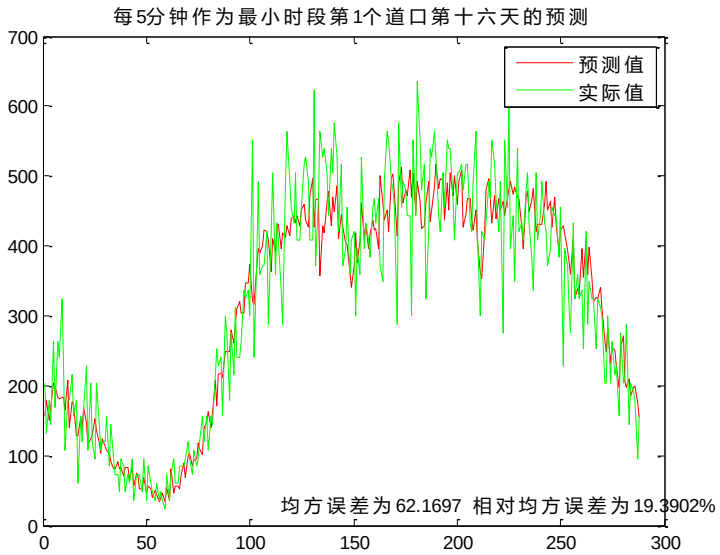
在数据预处理时，通过数据压缩，得到 10 分钟基准和 15 分钟基准的数据，重复 b 操作，得到回归结果。

2.1.3 运行结果和分析

① 时间间隔为 5 分钟

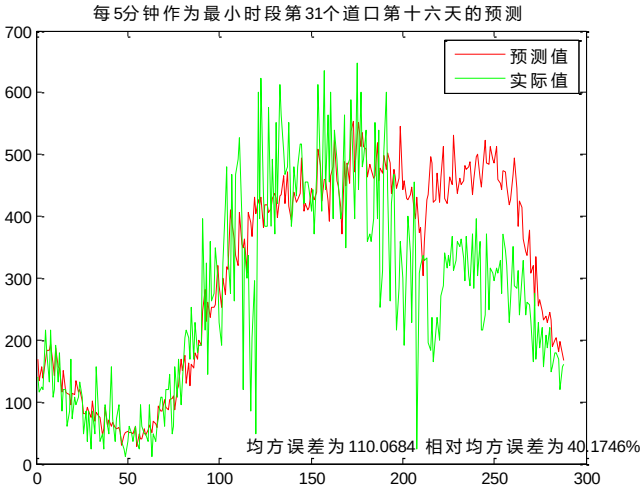
首先使用第一个道口做实验，设定数据间隔为 5 分钟预测结果如下：





可见预测效果理想，相对误差小于 20%。在间隔为 5 分钟的条件下，多做几组实验，经检验，测试的效果均很不错，置信度也可达 99%（设定 alpha=0.01，满足假设性检验）。

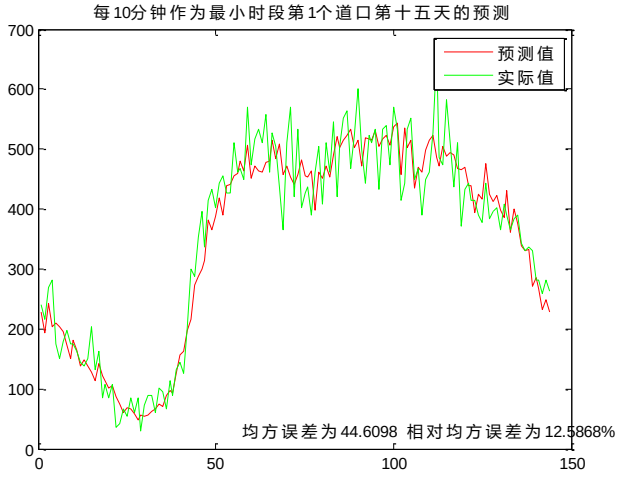
但也存在不理想的情况，如预测第 31 个道口第十六天的数据时，在后半部分，数据出现了偏移。如下图：

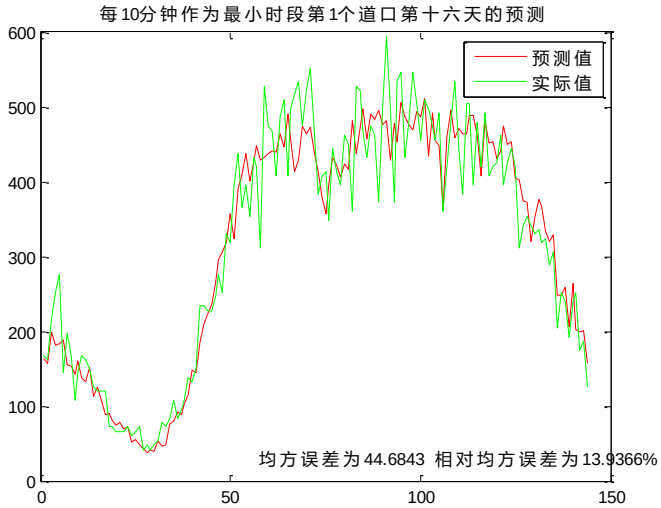


由此可见，预测效果在 200 个数据点之后有明显的偏移。而由 100-200 数据点剧烈变化我们也可以看出，当日肯定有一些特殊事件，比如在 120 和 210 数据采集点左右流量很小，很可能是车祸，进行了交通管制。

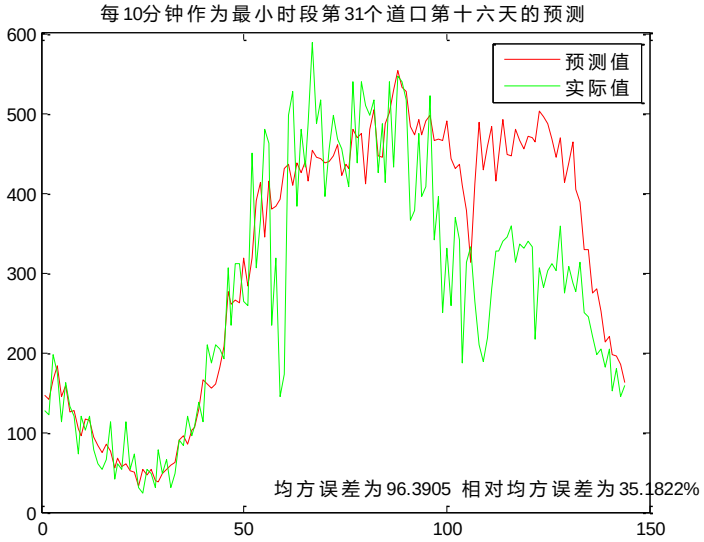
② 时间间隔为 10 分钟

由于时间间隔增加，数据量减小，为 144 个采集时刻。所以会使得预测曲线趋于平缓，绝对误差有所增大，但会造成拟合效果变差，置信概率减小。



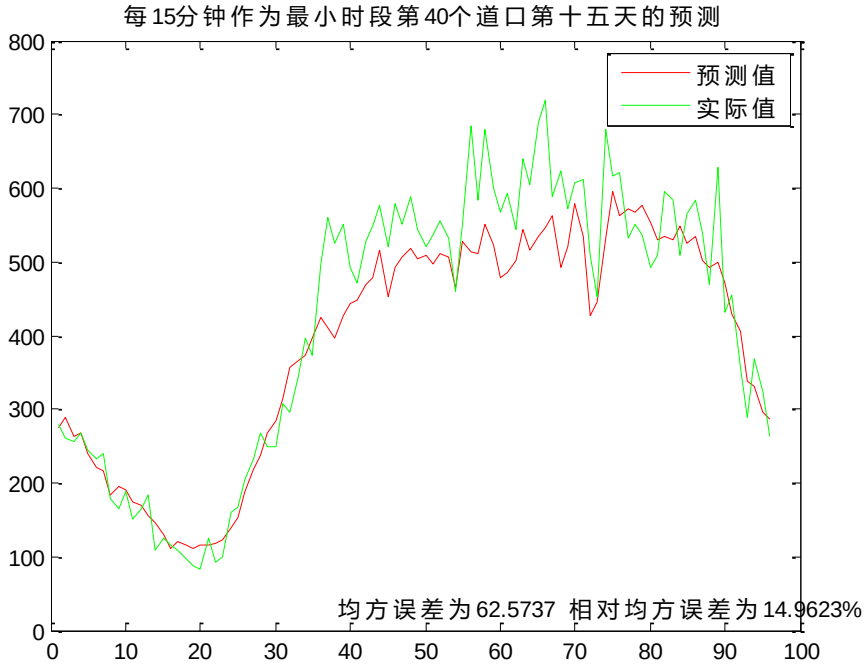


而①中 31 分钟的预测情况依然没有好转，如下：

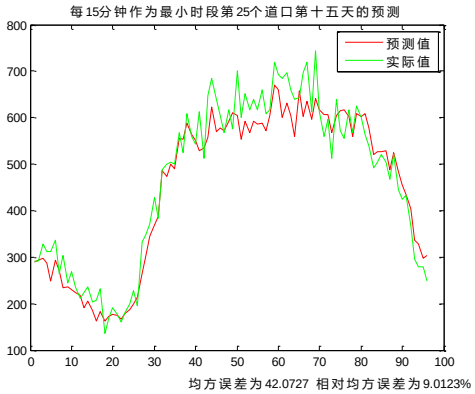


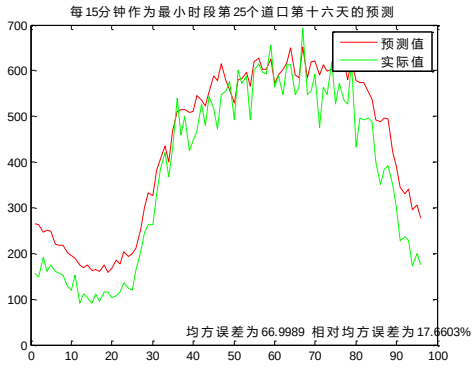
③ 时间间隔为 15 分钟

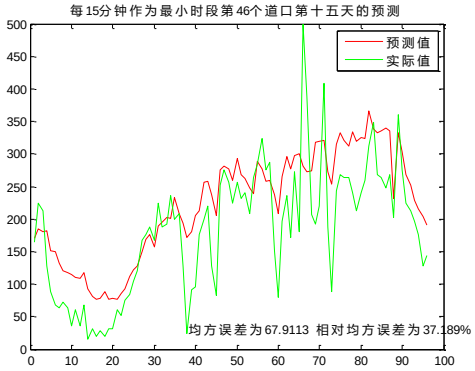
尺度进一步增大，这也将导致拟合的效果进一步变差，效果如下：

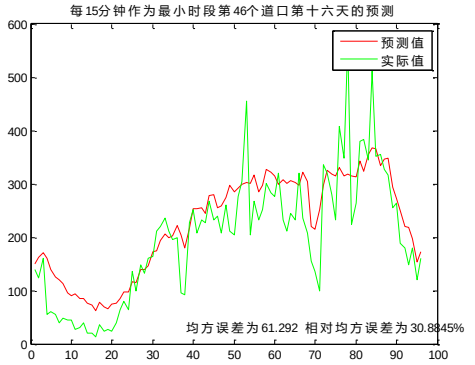


以第 40 道口为例，可见在中段有明显的偏差。展示更多样例，如下：









分析原因：由于采集的时间间隔增大，这将导致训练所用的数据量减小，从而也将导致预测的准确性下降，可靠性降低，此时不能再认为有可靠的预测把握了。

2.2 主成分分析

2.2.1 内容要求

基于课堂讲授的主成分分析法，对上述数据进行压缩和解压缩，并给出压缩比、压缩精度等参数。

2.2.2 设计思路

此部分工作和任务一中的病态分析类似，根据任务一中数据预处理后，可以得到一个 288*50*16 的矩阵。对其进行 PCA，并压缩数据。

PCA 的原理和任务一中的病态分析类似，这里简要说明：先进行数据的规范化处理，然后求取自相关矩阵的特征根，根据误差的阈值，进行截断，得到主成分，并计算压缩结果和压缩率。

而误差的计算需要回到 50 维空间进行处理。

2.2.3 运行结果和分析

① 将相对误差界限ε设为 5%，则根据下式

$$\frac{\sum_{i=m+1}^{n}\lambda_i}{\sum_{i=1}^{n}\lambda_i}\geq\varepsilon$$

进行压缩。得到结果为：

```
result =

相对误差界限是5%


result =

压缩恢复后与原始数据的误差是11.9672%.


result =

压缩比是0.52608
```

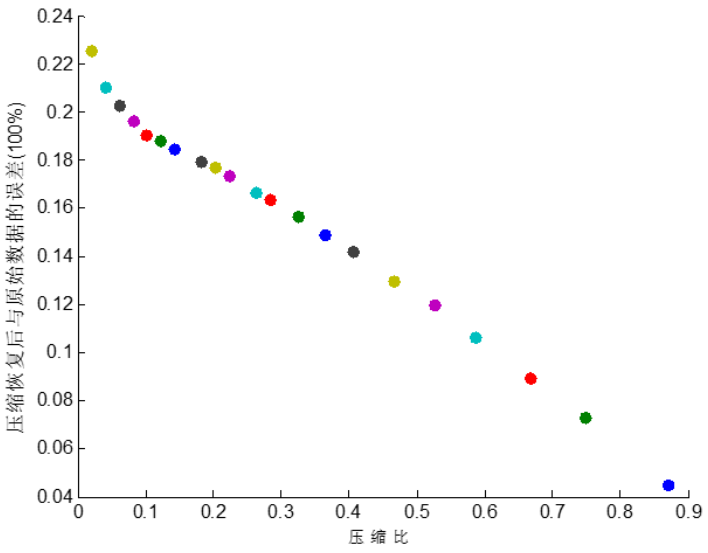
此时可利用压缩比公式倒推主成分 m：

$$\eta = \frac{m \times N + m \times n + n + n}{n \times N}$$

② 将相对误差界限ε设为 10%，得到结果为：

```
result =

相对误差界限是10%


result =

压缩恢复后与原始数据的误差是16.3553%.


result =

压缩比是0.28347
```

③ 将压缩比和恢复误差做成散点图的形式，可以得到：



分析：由①②③，可以分析得知，由于 PCA_compress 和 PCA_reconstruct 并不是完全的逆运算关系，所以会导致相对误差界限 rerr 和压缩恢复后与原始数据的误差并不等价。

而由曲线可以看出，随着压缩比降低，恢复误差是不断增加的，这也意味着压缩精度下降。所以说，压缩这一操作是以丢失数据信息为牺牲的。而作为研发者，应该在压缩精度和压缩比之间进行权衡，折中选择适合工程的方法。

2.3 聚类分析

2.3.1 内容要求

基于课堂讲授的 K-means 或系统聚类等聚类分析方法，选取早高峰时段（早 7:00-9:00）的数据，对相同时段各个路口的交通流量进行聚类分析（将路段进行聚类分析研究）；要求：若选择 K 均值聚类，则聚类数目可变化；如选择系统聚类，则要求绘制聚类谱系图。

2.3.2 设计思路

a. 数据预处理

本任务中要求的数据为早高峰时段（早 7:00-9:00）的数据，5 分钟为 1 时段，则有一共 24 个采集点，50 个道口则有 50 组这样的采集点。

由于有 16 天的数据，可以认为 16 天的数据是等价的，所以可以将 16 天的数据在每一个采集时刻取平均，得到一个 50*24 的矩阵，将 50 个道口作聚类分析。

b. 聚类分析

根据聚类分析 CA，N=50，m=24，聚类数目为 k 可变。

函数思路如下：

① 选取初始中心点，根据分类数目 k，挑选 k 个初始中心点，这里采取随机数生成的方式。

② 逐个利用每个样本修改中心点，对所有样本进行下述计算

a. 将其归入与其最近的中心点所在的类

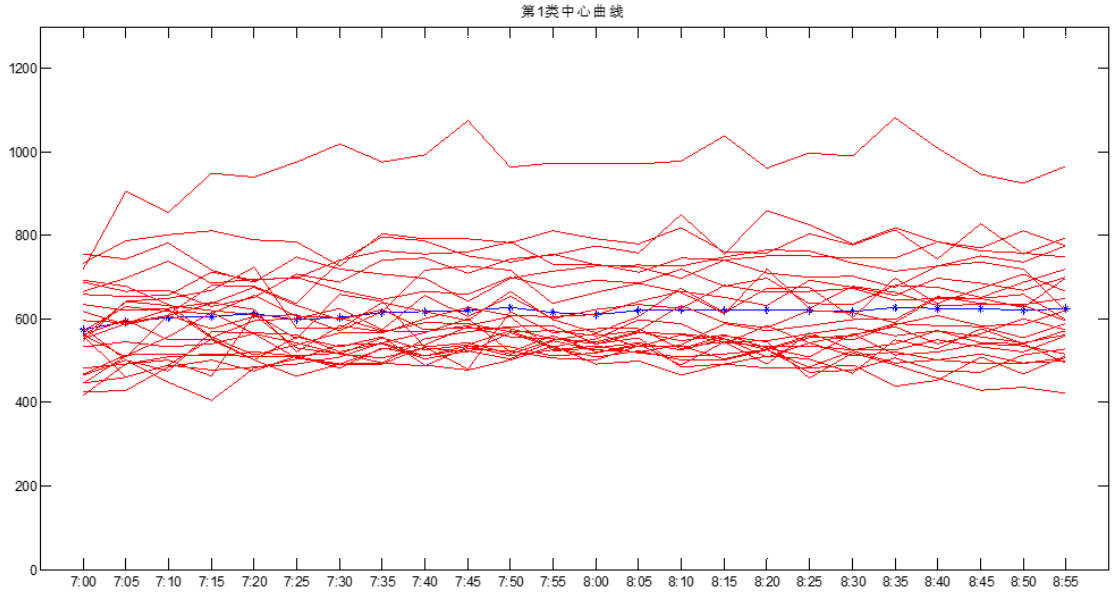
b. 重新计算该类的中心点，并用新的中心点替换原先中心点。

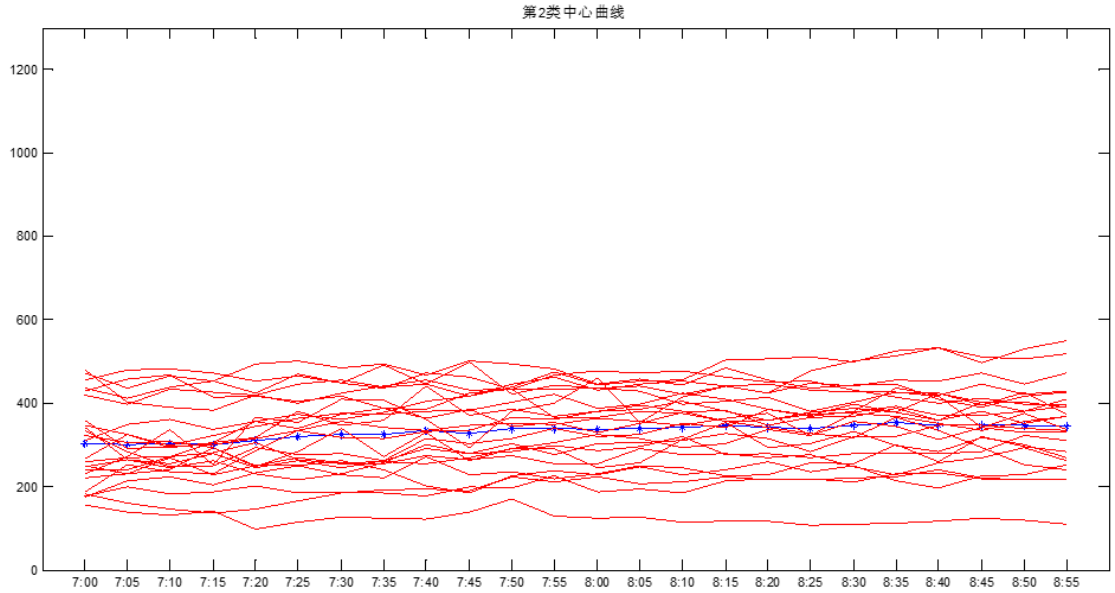
③ 停止准则：如果上述步骤开始时和结束后的中心点差别很小，停止分类。这里可以利用距离之和来体现差别。

2.3.3 运行结果和分析

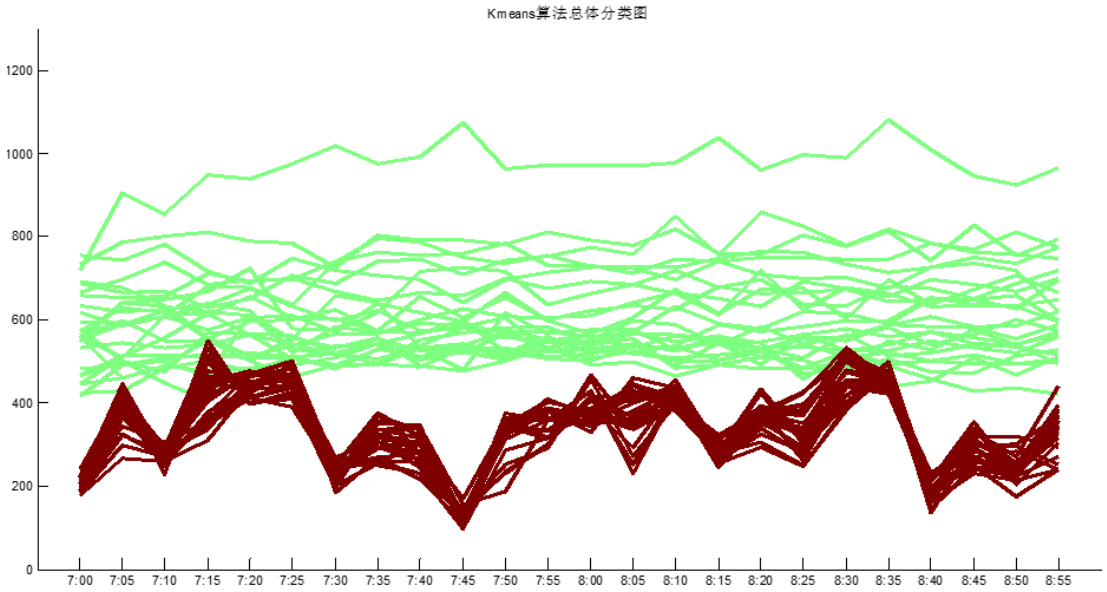
① 未归一化聚类结果

若选择聚类数目为 2，则可得两类的中心曲线如下：



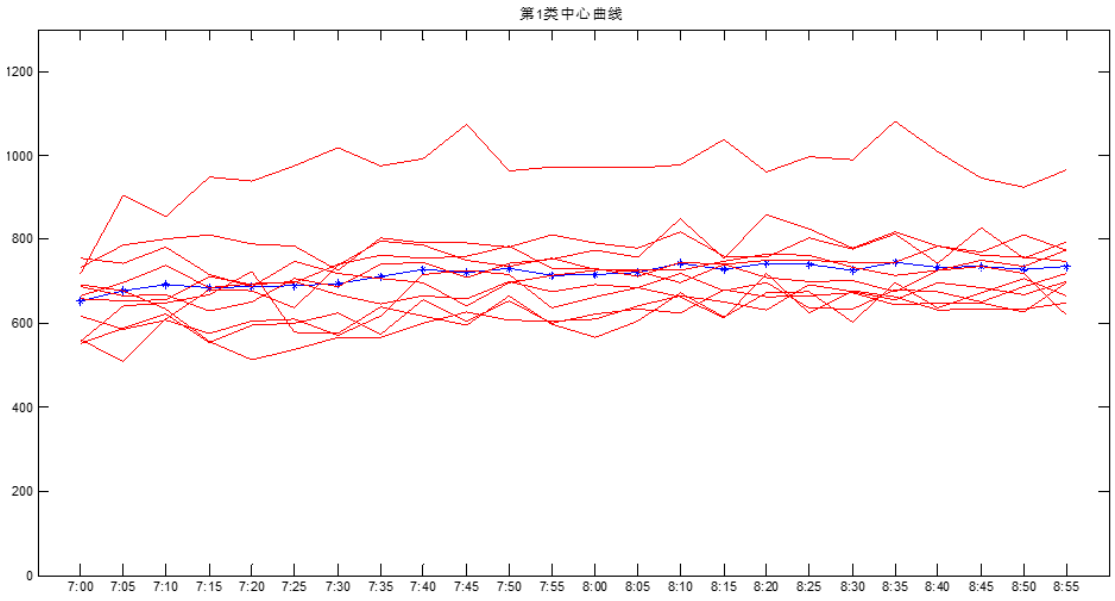


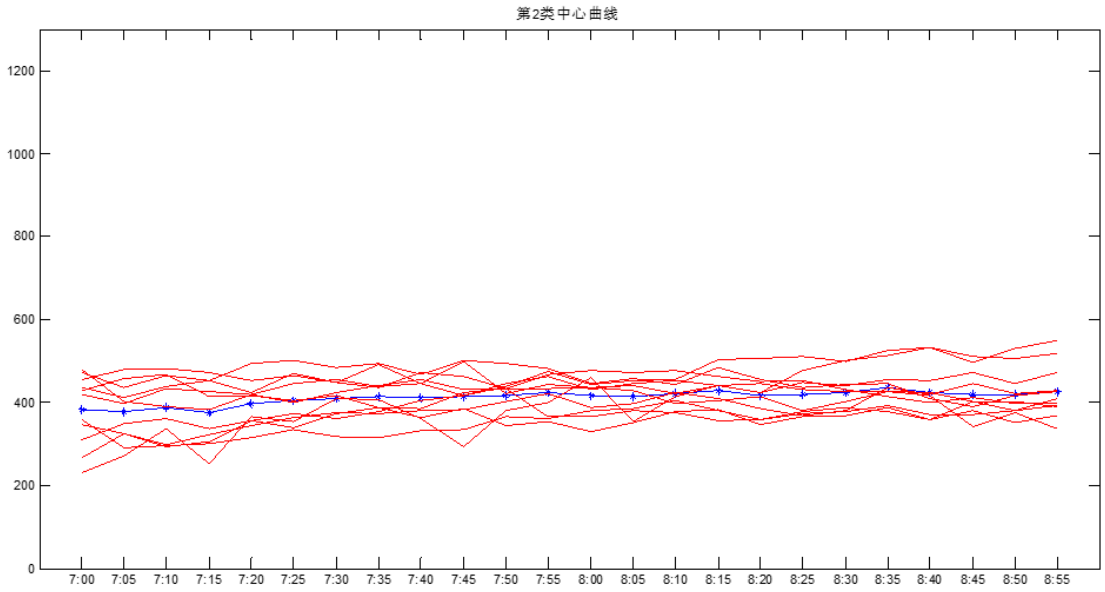
将想同类的曲线标记成同色、加粗，并将不同类的曲线放在同一张图上，如下：

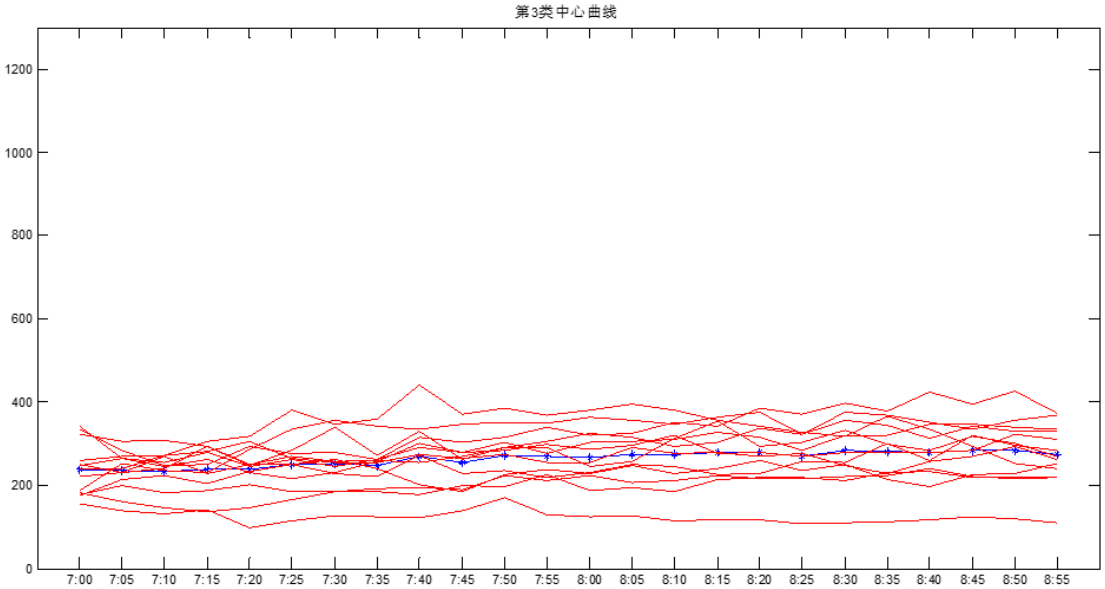


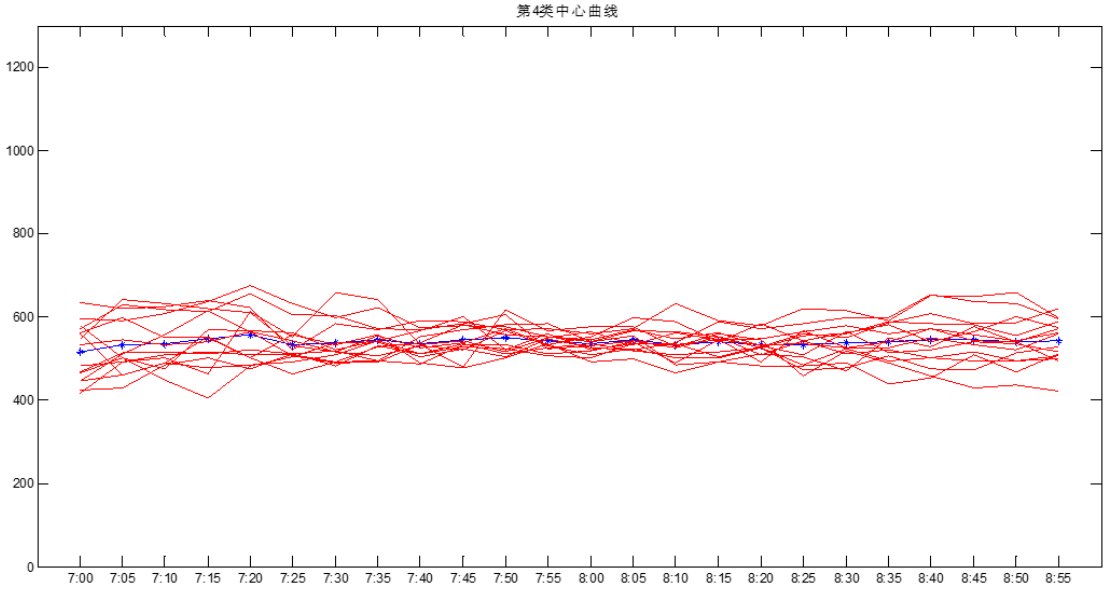
可见，聚类分析起到了一定的效果。下面再展示几组数据：

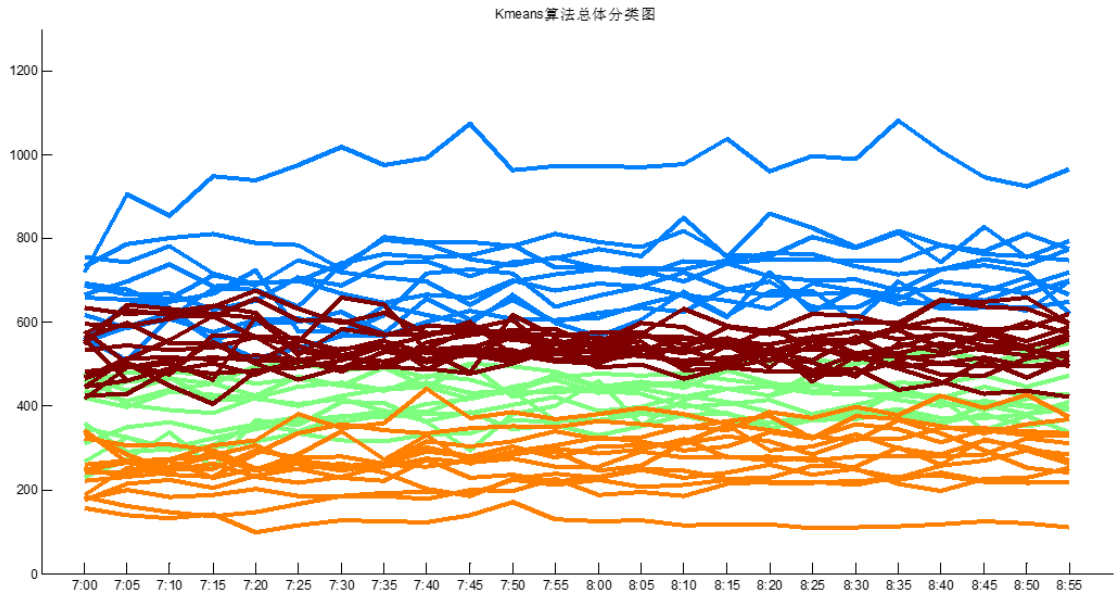
聚类数目为 4：



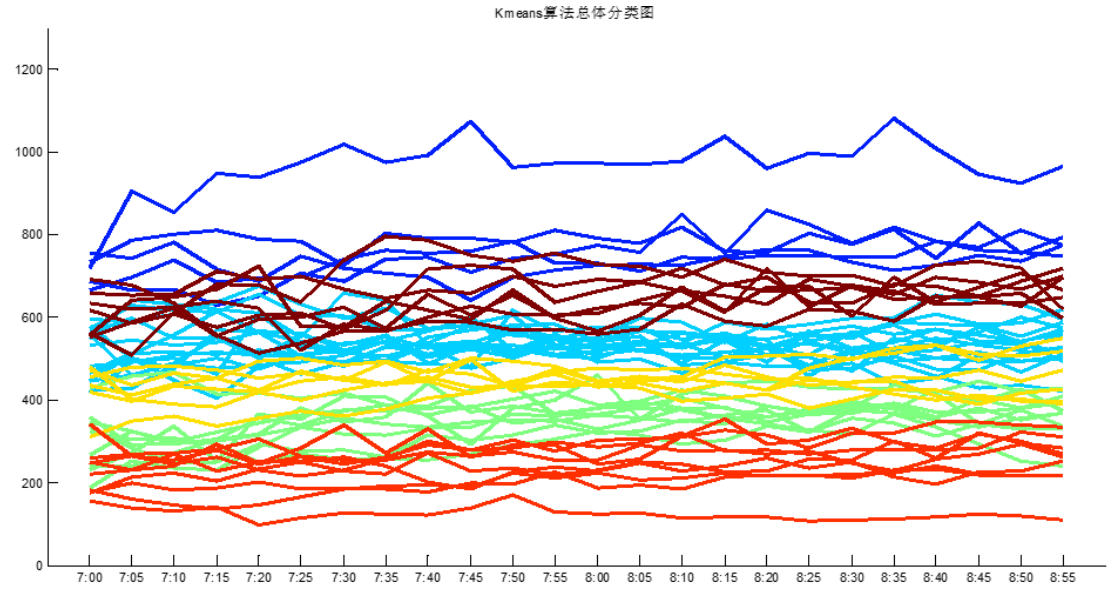

第2类中心曲线
1200
1000
800
600
400
200
0
7:00 7:05 7:10 7:15 7:20 7:25 7:30 7:35 7:40 7:45 7:50 7:55 8:00 8:05 8:10 8:15 8:20 8:25 8:30 8:35 8:40 8:45 8:50 8:55


第3类中心曲线
1200
1000
800
600
400
200
0
7:00 7:05 7:10 7:15 7:20 7:25 7:30 7:35 7:40 7:45 7:50 7:55 8:00 8:05 8:10 8:15 8:20 8:25 8:30 8:35 8:40 8:45 8:50 8:55


第4类中心曲线
1200
1000
800
600
400
200
0
7:00 7:05 7:10 7:15 7:20 7:25 7:30 7:35 7:40 7:45 7:50 7:55 8:00 8:05 8:10 8:15 8:20 8:25 8:30 8:35 8:40 8:45 8:50 8:55



聚类数目为 6：



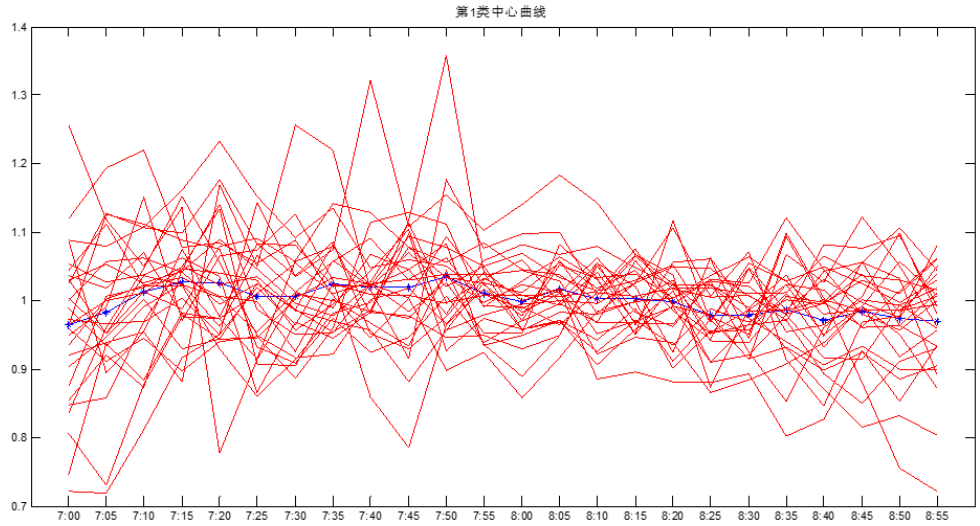
发现问题：当聚类数目增加时，分类结果是按照平均车流量进行分类，在上图体现为画了 5 条横线，把道口分成了 6 种。

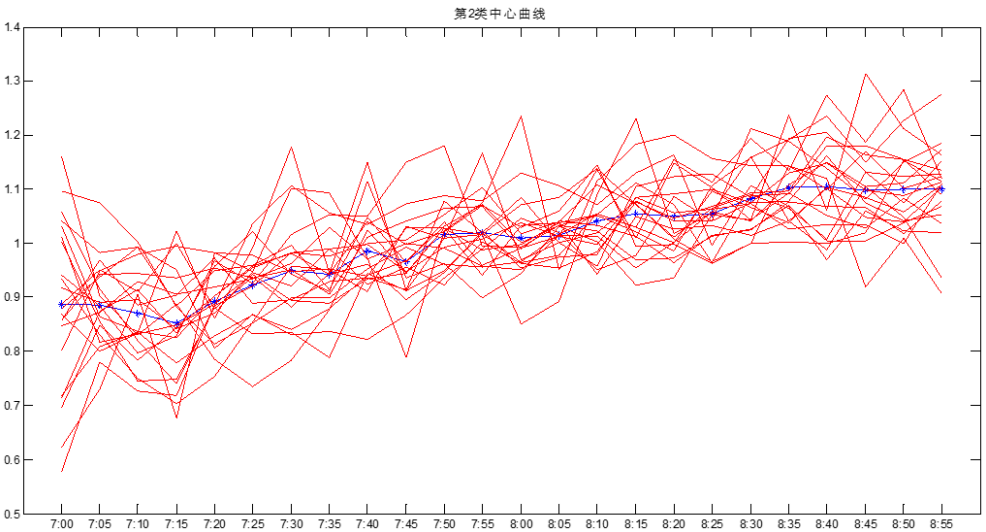
思考原因，这是因为车流大的道口和车流小的道口之间车流数量的差距真是非常大，几乎可以掩盖了单个道口在一天之内的车流变化。

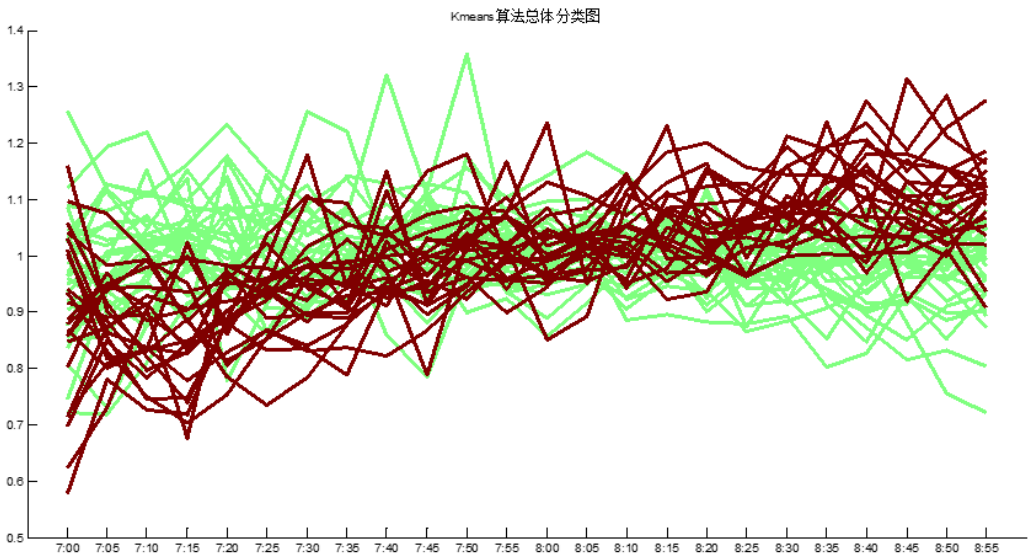
因此，如果一定要根据道口在一天之内的车流变化，可以采用现将左右道口的均值归一化，再进行聚类的方法来获得。

② 归一化处理后的聚类结果

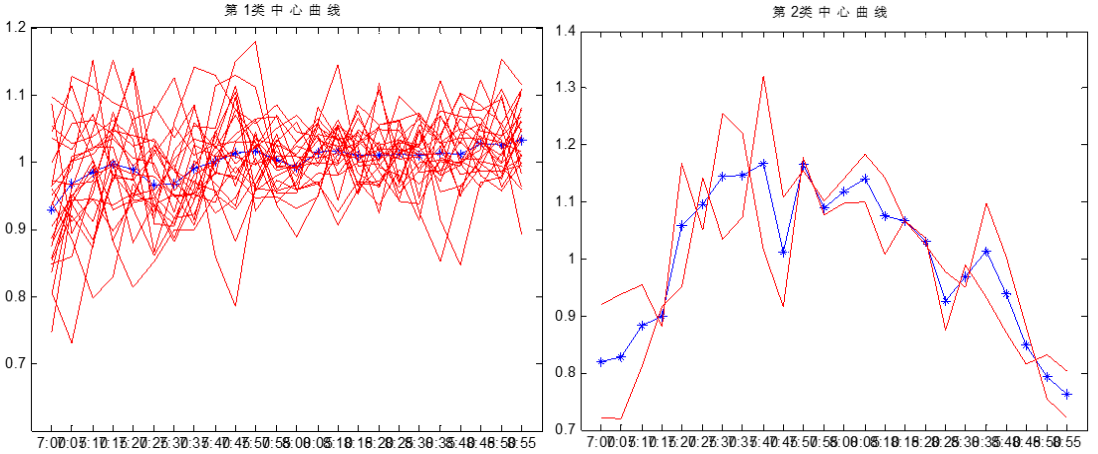
首先将数据进行归一化后，令聚类数目为 2，得到结果：

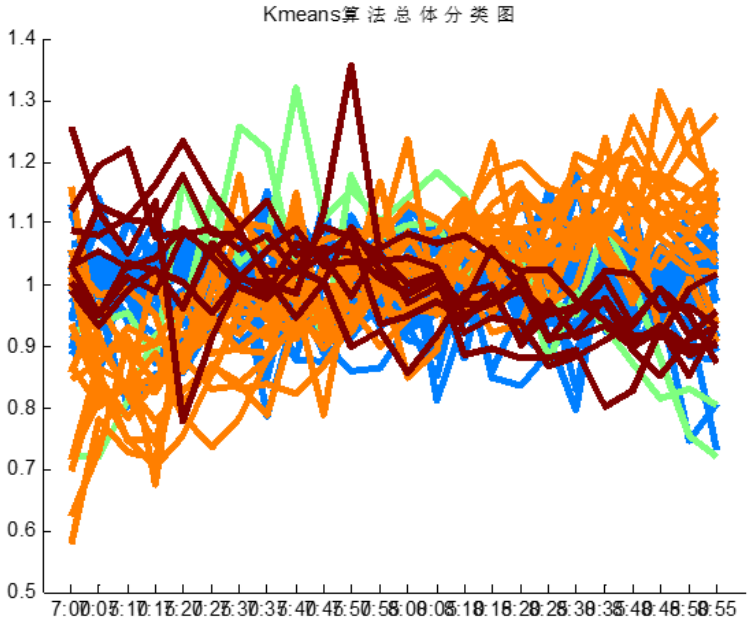




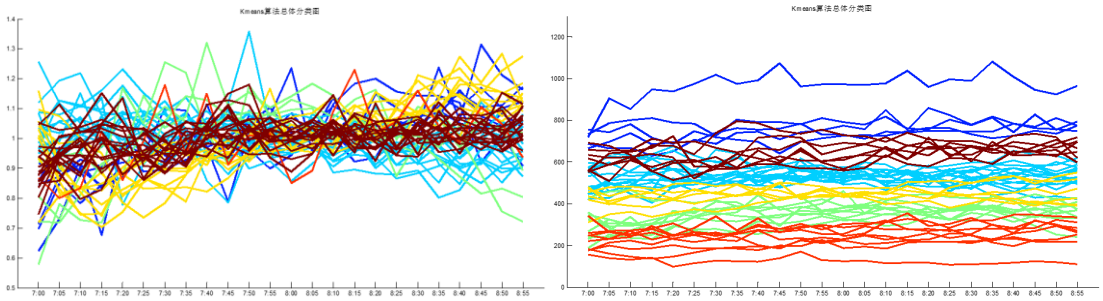


聚类数目为 4：





以聚类数目 6 为例，与非归一化结果对比：



左图为归一化的结果，右图未进行归一化。可见，归一化之后才能得到良好的聚类的结果。

由此可见，归一化处理在聚类分析中的重要性

2.4 【选做】meanshift 聚类分析

2.4.1 内容要求

自学至少一种新的聚类分析方法（可以是 SOM 聚类方法），对同一时段各个路口的交通流量进行聚类分析。

在这里，我选用了 meanshift 聚类方法。

2.4.2 设计思路及算法介绍

Meanshift 算法全称是均值漂移聚类算法。

利用不断尝试在某一半径内的均值的梯度方向，逐步移动选定区域，直到均值达到一个极值，找到密度最大的点。即先算出当前点的偏移均值，移动该点到其偏移均值，然后以此为新的起始点，继续移动，直到满足一定的条件结束。

Meanshift 是一个局部最优算法，因为空间均值是连续可导的，所以 meanshift 算法一定会收敛。其原理如下：

Meanshift 的核心思想是在一个迭代的过程中，先计算出当前点的偏移平均值，移动该点到其偏移平均值，然后以此为新的起始点，继续移动直到满足一定的阈值条件后结束。

在参考的文件中，作者将Mean Shift算法运用到特征向量的分析上，在图像的平滑和分割中进行了应用，并给出了几个实例应用。

Mean Shift算法的公式推导主要如下：

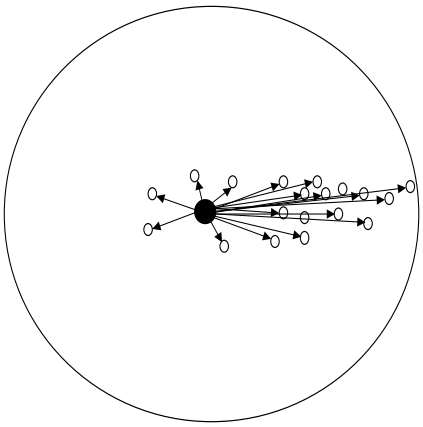
给定的d维空间R^{d}中的n个样本点$\mathrm{x}_1,\mathrm{x}_2,x_3,\cdot\cdot,x_n$。首先定义MS向量是：

$$\mathrm{M_h}(x)=\frac{1}{k}\sum_{x_i\in S_k}(x_i-x)$$

其中，S_h是一个半径为h的高维球区域，满足以下关系的y点的集合：

$$\mathrm{S_h}(x)\equiv\{\mathrm{y}:(\mathrm{y}-\mathrm{x})^{\mathrm{T}}(y-x)\le h^2\}$$

同时x是这个球域的中心，因此$(\mathrm{x_i}-\mathrm{x})$成为了一个偏离中心程度的衡量，所以mean shift向量就描述了落在区域S内的点到区域中心的偏移量的均值。同时，我们定义f是概率密度函数，通过区域内的采样获得，那么当x向f梯度方向移动时，$\mathrm{M_h}(x)$会趋向最优.



如右图所示，圆圈表示广义高斯球，黑点表示基准点x，因此x应当向密度函数梯度方向移动，可以使这个球$\mathrm{S_h}$得到更多的k和更小的$\mathrm{M_h}(x)$。在这之中，通常需要定义核函数，来配合距离的定义，从而使得算法更快收敛到局部最优解。记$\mathrm{K}(x)=k(\|x\|^2)$为核函数，其需要能满足：

(1) k是非负的

(2) k是非增的，即如果a<b那么$\mathrm{k(a)}\ge\mathrm{k(b)}$

(3) k是分段连续的，并且$\int_0^\infty k(r)dr<\infty$

一般我们都采用高斯核函数或者单位均匀核函数。也就是$\mathrm{N(x)}=\mathrm{e}^{-|\mathrm{x}|^2}$或者：

$$\mathrm{F(x)}=\begin{cases}1 & if\|x\|<1\\ 0 & if\|x\|\ge 1\end{cases}$$

通过核函数，我们可以实现，距离中心点不同的点，对$\mathrm{M_h}(x)$的贡献是不同的。在本次代码实现中，对于核函数采用以下形式：

$$k_N(x) = \exp\left(-\frac{1}{2}x\right) \qquad g(x) = -k'(x)$$

则在计算平均偏移向量时，所采用的函数为：

$$m_{h,G}(X) = \frac{\sum_{i=0}^{n} x_i g\left(\left|\frac{x - x_i}{h}\right|^2\right)}{\sum_{i=0}^{n} g\left(\left|\frac{x - x_i}{h}\right|^2\right)} - X$$

根据上述推导，可以得到算法流程为：

1） 首先随机选择初始点作为聚类中心点

2） 计算每个点到聚类中心点的聚类

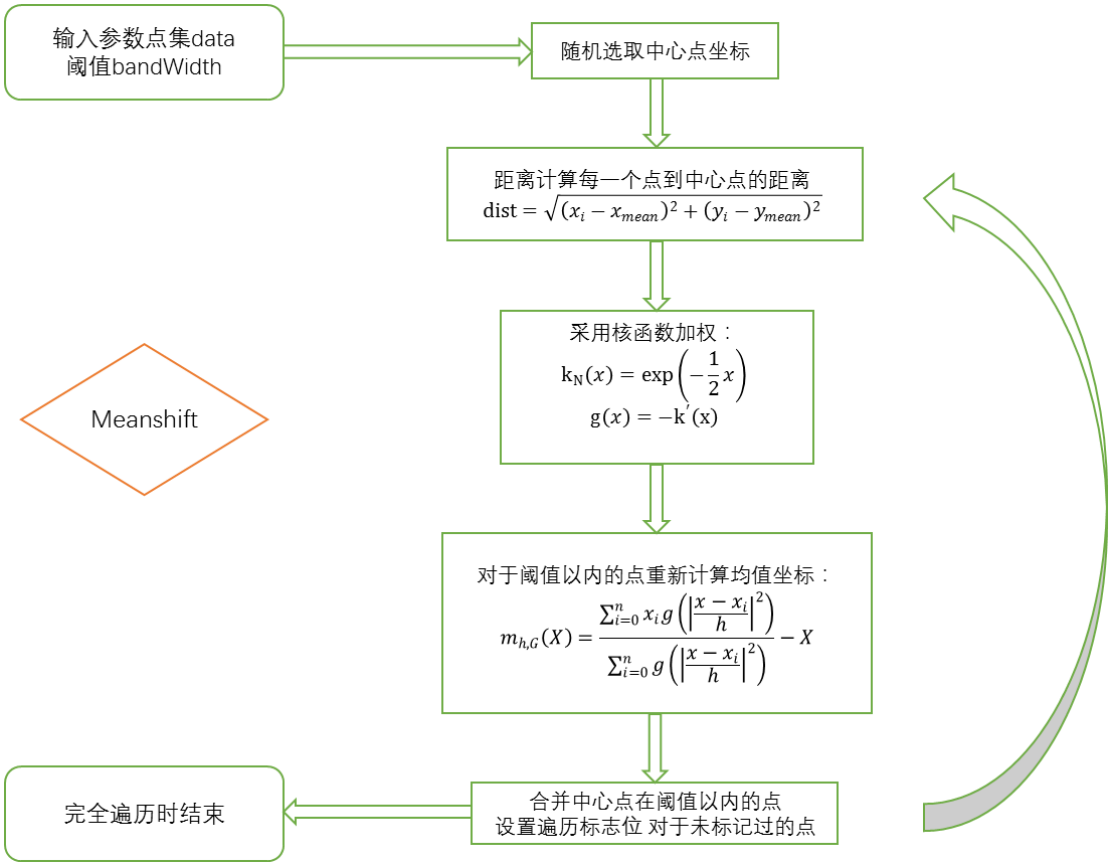
3） 根据核函数加权获得新的聚类中心点

4） 将距离新的聚类中心点阈值范围内的点纳入聚类结果

5） 当两类聚类中心点距离在阈值内合并成一类，否则形成新的一类

6） 如此循环 k 次，根据每个点所属类别的次数票选择每一个数据所处的类别。

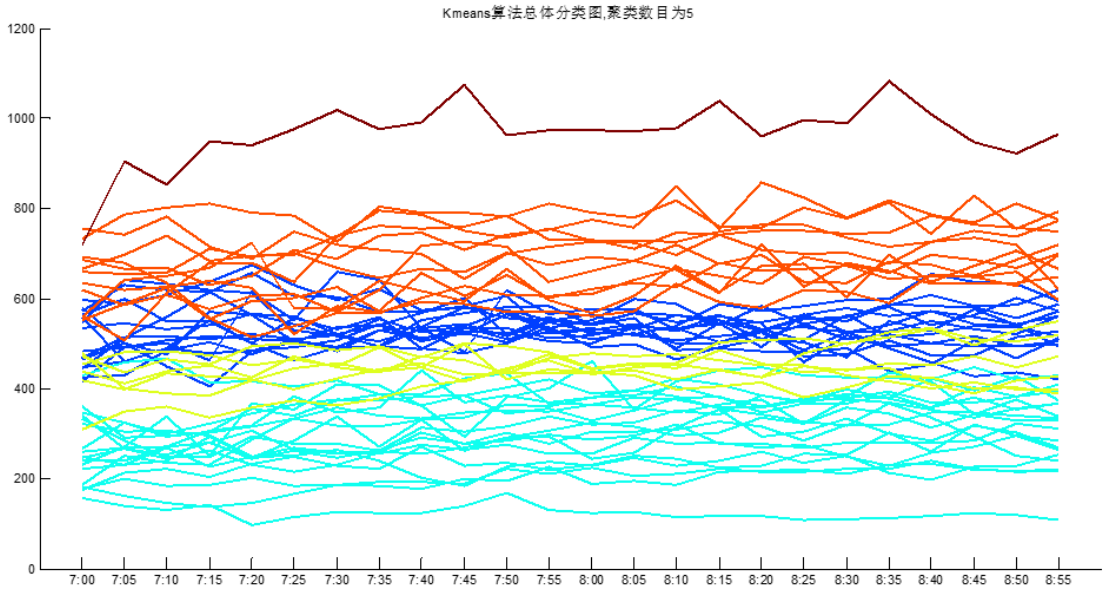
进一步可做出流程图如下：



2.4.3 运行结果和分析

① 数据未先归一化处理的聚类结果

选择合适阈值，如 500，得到聚类结果如下，此时一共聚成 5 类：



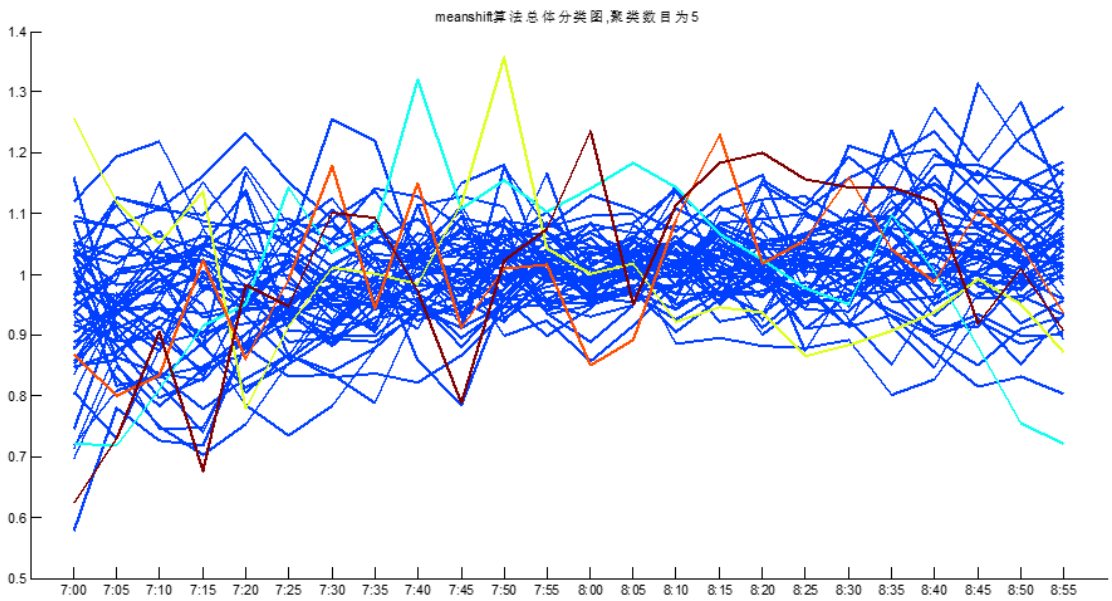
这里的问题和 kmeans 插值类似，当聚类数目增加时，分类结果是按照平均车流量进行分类，在上图体现为画了 4 条横线，把道口分成了 5 种。

思考原因，这是因为车流大的道口和车流小的道口之间车流数量的差距真是非常大，几乎可以掩盖了单个道口在一天之内的车流变化。

因此，如果一定要根据道口在一天之内的车流变化，可以采用现将左右道口的均值归一化，再进行聚类的方法来获得。

② 数据先归一化处理后的聚类结果

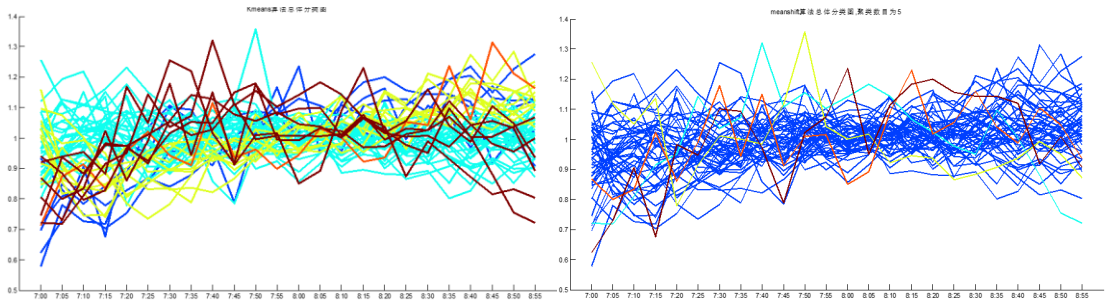
此时归一化后，数据的大小为 1 附近，减小阈值为 0.45，结果如下：



此时聚类效果较好。

③ 与 kmeans 方法对比

以聚类数目 5 为例，左边是 kmeans 方法结果，右边是 meanshift 方法结果：



由结果可见,meanshift 算法相比于 kmeans 算法,有其中一类占了大多数 ，及其中大多数之间的差别较小，都属于紧凑型，因此 meanshift 将它们全部归为一类，而剩下的很多类都只有 1-2 个道口。

所以说 Meanshift 在处理这组数据上的表现并不好。

不过 meanshift 的优势在于其是一个局部算法，复杂度较低，很高效，在算法执行过程中，kmeans 需要约 1 秒，而 meanshift 只需要三分之一的时间就可以完成。

3. 作业问题与体会

3.1 作业中遇到的问题

① 黑箱建模中数据如何预处理

在黑箱建模中，我得到的初始数据是一个 50*288*16 的数据，而我们在系统工程课程中并未涉及到这样的数据，而且我也调研了相关文献，没有发现很好的解决办法。

解决办法：经过我仔细分析，我意识到通过前 14 天的数据预测第 15 天和第 16 天的，而每一天的数据实质上是由 50 个道口的数据组成，每一个道口有 288 个时刻的数据量。而根据黑箱建模的方法，必须要有足够的训练值，而这些训练值则要求有 X 和 Y,根据多元函数回归的关系,可以让 Y 代表一个要预测的道口。

那么我们的目的更加明确了，即预测第 15 天和第 16 天该道口的车流量，所以，其实前 14 天的数据是等价的，应该综合利用，所以可以取平均，得到 50*288 个训练数据，完成数据的预处理。

② 黑箱建模中病态回归的阈值选取

在黑箱建模中，最重要的一部分是多元线性回归，然而在回归时容易遇到问题，就是有些特征值较小，导致矩阵约为不可逆，使得最后的预测结果出现了很大 的数值。

解决方法：为了解决这一点，在作业中我加入了病态回归的处理。但是阈值选取成为了难题。因为如果阈值过高，则会使得过多的特征值被消掉，维数下降过多，使得预测的精准度降低；而阈值过低，还是会出现特征值较小的情况，影响预测效果。所以，最终我通过不断尝试，在预测精度和阈值之间不断比较，得到了较好的效果。

③ 黑箱建模中哪里进行假设性检验

这也是第四章作业的一个问题，病态线性回归问题中，显著性检验是在自变量降维去线性之前，还是之后，还是前后都检验？

解决方法：最终我采用了在降维之后进行检验。因为对于病态问题，在降维之前，特征矩阵接近奇异矩阵，所以其逆的特征值较大，从而导致剩余平方和较大，检验数 F 的值较小，可能会接受原假设，认为变量是非线性的。其实病态问题产生的本质就是存在了一些冗余的变量，如果把这些变量去除之后，其他变量本身的线性度较好，能够通过显著性检验。所以应该在降维之后（把这些冗余变量去除之后）进行显著性检验。

而且以 F 检验为例，其检验要求是各变量之间是线性无关的，才能使得变量 F 满足自由度为(n,N-n-1)的 F 分布。在降维前不满足条件，无法进行检验。

④ 在主成分分析方法中，进行线性回归之后，如何将其映射回 x 域

一开始的时候，我认为这是很难做到的，因为变换矩阵 Q 为(n×m)，是将 n 维变量变为 m 维变量。直接的思路是将其直接乘以 Q 的逆，进行反变换。但是在编程中求解一个非方阵的逆是没有途径的。

解决方法：在仔细研究原理后，我惊奇地发现原来〖QQ〗^T=I，这是根据 PCA 变换矩阵每一列向量都是单位正交列向量的性质，其左乘或右乘其自身的转置，得到的都是单位矩阵，所以其转置即为其逆矩阵。基于此，可以得到 n 维向量。

⑤ 聚类过程中会把车流量相似的曲线聚成一类

在一开始聚类时，我发现得到的效果是按照平均车流量进行分类，在上图体现为画了 4 条横线，把道口分成了 5 种。

解决方法：思考原因，这是因为车流大的道口和车流小的道口之间车流数量的差距真是非常大，几乎可以掩盖了单个道口在一天之内的车流变化。

因此，如果一定要根据道口在一天之内的车流变化，可以采用现将左右道口的均值归一化，再进行聚类的方法来获得。

3.2 作业收获和体会

本次大作业综合运用了之前学习到的线性回归分析、病态问题分析、PCA 主成分分析、kmeans 聚类分析等多种分析方法，基本综合了前大半学期的全部所学，全都倾洒至这次作业中，让我感觉有些“练成神功”的感觉。

而本次大作业也基于了一个和课堂完全不同的对象，即真正的交通流数据，也让我们提升了实际解决生活中工程问题的能力，感受到了系统工程的魅力，让我感受到了数学工具和数据分析的用处。

系统工程的这些数据分析方法也十分有用，其实在很久以前我们就接触到了这些知识，比如一元的拟合问题，它的本质就是只有两个未知数，但给了我们很多组的数据,不能直接用这些数据去解出两个未知数,因为方程是超定的。那么，为了使得一个多维的解变成了二维的解，这就需要一个投影的过程，相当于一个映射，而这个映射本身就是我们系统工程里讲述的经典公式：

$$(XX^T)^{-1}X^TY$$

假设 X 为 N×n，Y 为 N×1，则最后的结果是一个长度为 n 的向量，这就体现了投影的过程。

最后，也非常感谢这次大作业的机会，让我总结了这学期数据分析几类函数的 matlab 库，以后需要，调用起来也十分方便。

4. 参考资料

[1]Dorin Comaniciu, Mean Shift A Robust Approach Toward Feature Space Analysis,IEEE

[2] Changjiang Yang, Ramani Duraiswami and Larry Davis, Efficient Mean-Shift Tracking via a New Similarity Measure

[3]胡坚明，清华大学系统工程导论课件

[4]第七章课后阅读材料《Cluster Analysis Basic Concepts and Algorithms》

5. 代码文件说明

打开“源代码”文件夹，将本次作业的 4 个部分均分别放在不同文件夹中，均可打开 PARTi.m 直接运行。

“PART1 黑箱建模”中，PART1.m 为任务一的主函数，data_preproceecing.m 为数据预处理函数，linear_regression.m 为多元病态线性回归函数。

“PART2 主成分分析压缩数据”中，PART2.m 为任务二 PCA 压缩的主函数，PCA_compress.m 为 PCA 压缩函数，PCA_reconstruct.m 为 PCA 解压缩函数。

“PART3Kmeans 聚类分析”中，PART3 为任务三 Kmeans 聚类分析的主函数，data_preproceecing.m 为数据预处理函数，kmeans_clustering.m 为 kmeans 聚类分析函数。

“PART4meansshift 聚类分析”中，PART4.m 是任务四：选做内容的主函数，在这里选用了 meanshift 聚类方法,data_preproceecing.m 为数据预处理函数，meanshift_cluster.m 为 meanshift 聚类操作函数。