

# **《系统工程导论》 课程设计**

## **系统工程方法在交通数据处理 的应用**

### **课题报告**

**班级： 自 45**

**姓名： 林子坤**

**学号： 2014011541**

## 目录

一、课题说明.....	3
二、文件说明.....	3
三、必做任务.....	4
3.1 黑箱建模方法.....	4
3.1.1 题目要求.....	4
3.1.2 方法原理.....	4
3.1.3 步骤与结果.....	5
3.2 主成分分析法.....	6
3.2.1 题目要求.....	6
3.2.2 方法原理.....	6
3.2.3 步骤与结果.....	7
3.3 K-MEANS 聚类方法.....	8
3.3.1 题目要求.....	8
3.3.2 方法原理.....	8
3.3.3 步骤与结果.....	9
四、选做任务.....	10
4.1 使用 SVR 方法进行交通流预测.....	10
4.1.1 题目要求.....	10
4.1.2 方法原理.....	10
4.1.3 步骤与结果.....	11
4.1.4 对比分析.....	12
4.2 核主成分分析法.....	12
4.2.1 题目要求.....	12
4.2.2 方法原理.....	12
4.2.3 步骤与结果.....	14
4.2.4 对比分析.....	15
4.3 SOM 聚类方法.....	16
4.3.1 题目要求.....	16
4.3.2 方法原理.....	16
4.3.3 步骤与结果.....	17
五、总结与感受.....	22
参考文献.....	23

# 系统工程方法在交通数据处理的应用

## 课题报告

林子坤 （自动化系 自 45 班 2014011541）

[摘 要] 在本学期的系统工程课程中，我们学习到了许多数据处理的方法，例如解释性结构建模、黑箱建模、主成分分析方法、因子分析法、聚类分析等等。在本次课题中，我们将这些方法加以运用与综合，将复杂的北京市路网交通流原始数据进行建模分析处理，并探索其它系统工程方法在这一领域的应用与魅力所在。

[关键词] 黑箱建模 主成分分析 聚类 核主成分分析 SOM 聚类方法 交通数据流

## 一、课题说明

本次课题单人独立完成，完成了课题要求中所有的 3 个必做任务和 3 个选做任务。

本次课题使用 MATLAB R2017a 进行编写，所有代码可以在任何安装有 MATLAB 程序的计算机中运行。

## 二、文件说明

由于代码量较大，文件较多，因此对附在“code”文件夹中的文件作说明如下。

所属题目	文件名	内容说明
数据集	data_16d.mat	flow_50link: 流量数据，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）
		occ_50link: 占有率数据，采集方式同上
		link_info: 50 个检测器的相关信息，包括检测器 id，路口号，路口名等信息
		time_link: 16 天每天 288 个数据对应的检测时间
第 1 题	compAss1.m	必做任务 1 的主函数
	linear_regression.m	线性回归黑箱建模
第 2 题	compAss2_1.m	必做任务 2 的主函数 1
	compAss2_2.m	必做任务 2 的主函数 2 (两个主函数执行一个即可，效果不同，在报告中予以说明)
	pca_compress.m	PCA 数据压缩
	pca_reconstruct.m	PCA 数据恢复
第 3 题	compAss3.m	必做任务 3 的主函数
	kmeans_clustering.m	k 均值聚类

第 4 题	optAss4 文件夹	optAss4.m 为主函数，其余文件为运行 libsvm 所需要的程序包
第 5 题	optAss5_1.m	必做任务 2 的主函数 1
	optAss5_2.m	必做任务 5 的主函数 2 (两个主函数执行一个即可，效果不同，在报告中予以说明)
	kpca_compress.m	核主成分分析方法数据压缩
	kpca_reconstruct.m	核主成分分析方法数据恢复
第 6 题	optAss6_1.m	必做任务 6 的主函数 (对时段进行聚类)
	optAss6_2.m	必做任务 6 的主函数 (对路口进行聚类)

## 三、必做任务

### 3.1 黑箱建模方法

#### 3.1.1 题目要求

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）。

#### 3.1.2 方法原理

以下为考虑病态情况的多元线性回归的实现过程。

##### (1) 数据归一化

为了统一样本的统计分布特性，需要对样本进行归一化。直接调用 Matlab 函数 `zscore` 即可。

```
X_zscore=zscore(X)';
```

##### (2) 获得去线性化的维数 $m$

在自变量降维去线性之后，相互之间线性无关的自变量个数应缩减为  $m$  个，且  $m$  应使得相对逼近误差可以接受。在本程序中，相对逼近误差不超过 3% 视为可以接受，即：

$$\frac{\sum_{i=m+1}^n \lambda_i}{\sum_{i=1}^n \lambda_i} \leq 5\%$$

##### (3) 确定各个参数矩阵

使用 Matlab 函数

```
[Q,~]=eig(A); D=eig(A);
```

得到特征值矩阵和特征向量矩阵。

从  $Q$  矩阵中取出特征值最大的  $m$  列获得  $Q_m$  矩阵，并根据以下公式求得  $Z$ 、 $d$ 、 $c$  矩阵。

```
Z=Qm'*X_zscore;
d=inv(Z*Z')*Z*Y_zscore';
c_zscore=Qm*d;
```

$c=c\_zscore.*sqrt(var(Y))./sqrt(var(X))'$ ;

最后，根据变量间的约束关系求得常数项 $c_0$ 。

#### (4) 误差检验

对于得到的模型，通过求取平均相对误差的方式进行效果评估。公式为：

$$ARE = \frac{1}{m} \sum_k \frac{x - \mu}{\mu} \times 100\%$$

其中 $x$ 为预测结果， $\mu$ 为实际数值， $m$ 为总时段数。

#### 3.1.3 步骤与结果

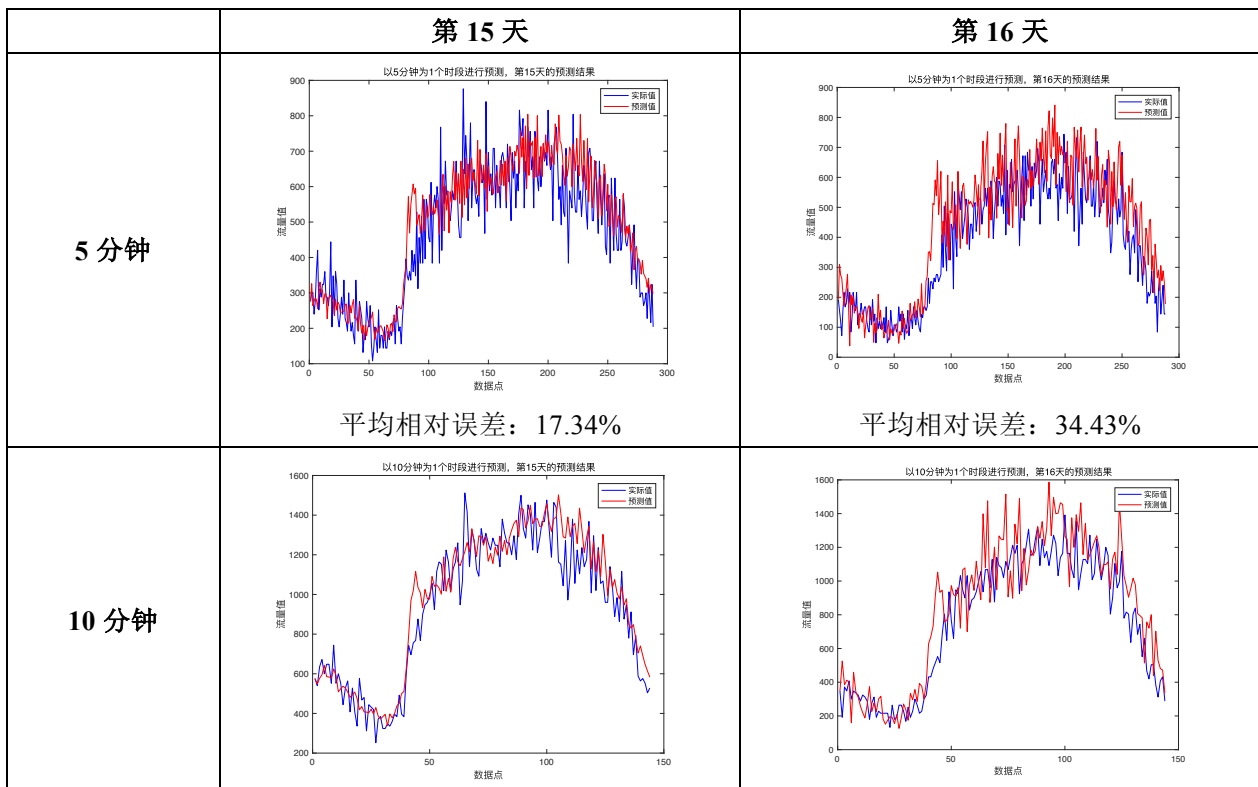
实验中使用的黑箱建模方法是多元线性回归（且考虑病态情况），需确定其自变量和因变量。假设每天第 $k$ 个时段的流量完全取决于所有之前时段的流量，则将第 $k$ 个时段的流量设置为因变量，将之前所有时段的流量设置为自变量，则对于每个时段的流量有 14 个训练样本（14 天的流量数据），将其进行线性回归分析。回归方程形式如下：

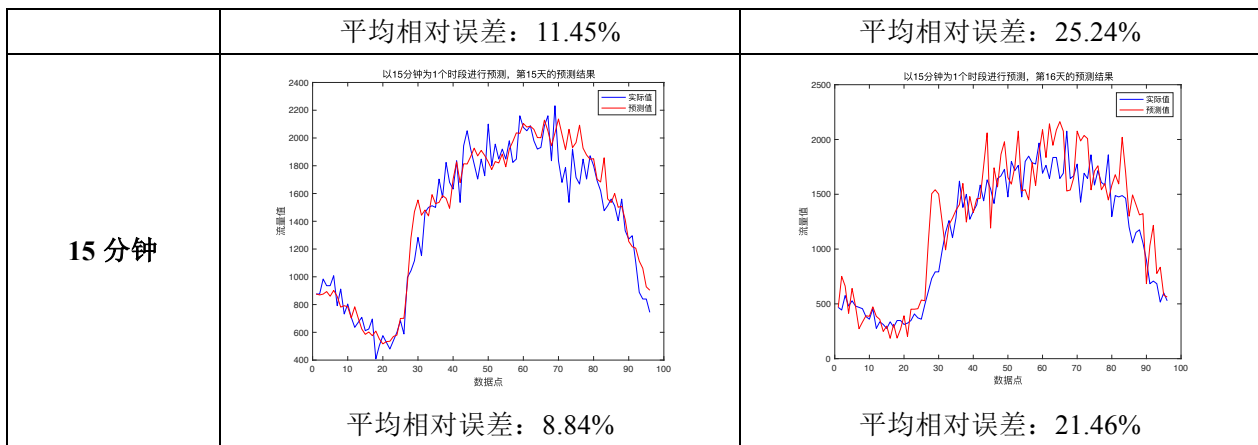
$$Y_k = \sum_{i=1}^{k-1} c_i X_i$$

其中 $Y_k$ 代表第 $k$ 个时段的流量， $X_i$ 代表第 $i$ 个时段的流量（ $i < k$ ）。

运行程序，以第 25 个路口为例，获得结果如下所示。

请输入你想要进行预测的路口：25  
以5分钟为1个时段进行预测，第25个路口：第15天的预测平均相对误差17.341848%，第16天的预测平均相对误差为34.434484%  
以10分钟为1个时段进行预测，第25个路口：第15天的预测平均相对误差11.446903%，第16天的预测平均相对误差为25.239123%  
以15分钟为1个时段进行预测，第25个路口：第15天的预测平均相对误差8.838720%，第16天的预测平均相对误差为21.461143%





通过以上结果可以看出：通过线性回归对原始数据的趋势预测很好，但是仍存在较大的误差。分析原因是：每天的交通流仍存在一定的随机性，在预测时只能对趋势进行较好拟合，但对于具体数值仍然会因为每天的特异性存在偏差。其次，如果拉长每次测量与预测的时间段，效果将会变好，这是因为延长时间段将会使得每个时间段的随机性减小（体现在图中是“锯齿”减少），从而减小了预测难度并增加了预测准确率。

## 3.2 主成分分析法

### 3.2.1 题目要求

基于课堂讲授的主成分分析法，选取不同主成分，对上述数据进行压缩和解压缩，并对比分析压缩比、压缩精度等参数。

### 3.2.2 方法原理

[A] 主成分分析法进行数据压缩的具体步骤如下所述。

#### (1) 数据归一化

为了统一样本的统计分布特性，需要对样本进行归一化。直接调用 Matlab 函数 `zscore` 即可。

$$X\_zscore = zscore(X)';$$

#### (2) 确定特征值矩阵和特征向量矩阵

使用 Matlab 函数

$$[Q, \sim] = eig(A); D = eig(A);$$

得到特征值矩阵和特征向量矩阵。

#### (3) 计算主成分

在自变量降维去线性之后，相互之间线性无关的自变量个数应缩减为  $m$  个，且  $m$  应使得相对逼近误差可以接受。在本程序中，相对逼近误差小于  $rerr$  视为可以接受，即：

$$\frac{\sum_{i=m+1}^n \lambda_i}{\sum_{i=1}^n \lambda_i} < rerr$$

取前  $m$  个特征向量，组成  $Q_m$  矩阵；根据以下公式计算各样本数据在主成分方向上的投

影:

$$y(t) = Q_m^T * X\_zscore(t)$$

#### (4) 输出结果

将以上步骤得出的结果置为函数的输出。其中:

输出量 pcs 代表各个主成分, 每一列为一个主成分, 即为  $Q_m$  矩阵;

输出量 cprs\_data 代表压缩后的数据, 每一行对应一个数据点, 即为  $y(t)^T$  矩阵;

输出量 cprs\_c 代表压缩时的一些常数, 包括数据每一维的均值和方差等, 第一行输出样本均值, 第二行输出样本标准差。

利用以上三个变量应当可以恢复出原始的数据。

**[B] 主成分分析法进行数据恢复的具体步骤如下所述。**

重建数据时, 先利用以下公式求出规范化后的样本成分:

$$X\_zscore(t) = y(t)^T Q_m^T$$

随后利用以下公式求出恢复出来的数据:

$$X(t) = X\_zscore(t) * \delta + \bar{X}$$

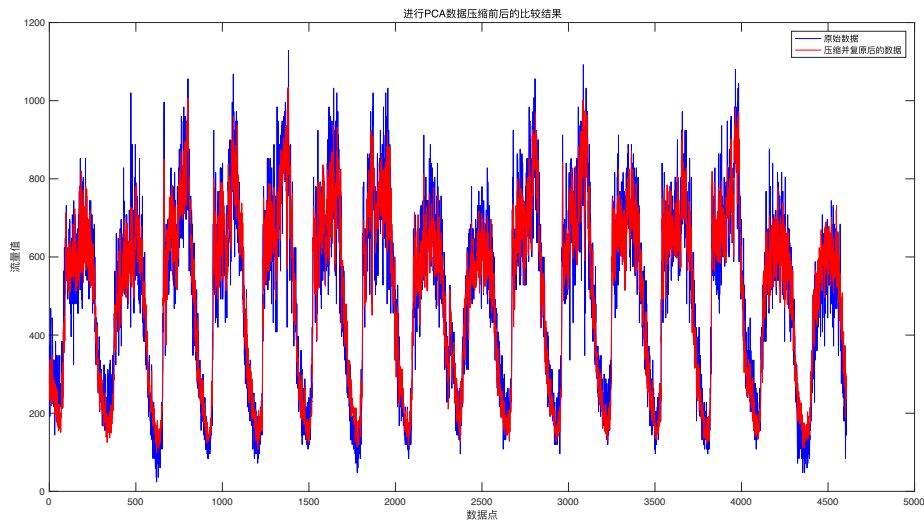
**[C] 最终, 对获得的结果进行压缩精度分析, 用来评估的参数为均方根误差(RMSE)。**

### 3.2.3 步骤与结果

为了尝试不同压缩方式, 并比较它们的压缩效果, 本人在编写程序时, 首先将每个路口 16 天的数据由  $288 \times 16 \times 50$  的三维矩阵转化为  $4608 \times 50$  的二维矩阵, 即将 16 天的流量数值按照时间顺序连接在一起, 这保证了 PCA 主成分分析程序能够对这个矩阵进行处理。

为了分析对  $4608 \times 50$  的矩阵进行处理和对  $50 \times 4608$  的矩阵进行处理的效果与性能差异, 本人也进行了两次实验, 并以路口 25 为例进行图像绘制, 获得比较结果如下所示。

#### (1) 使用 $4608 \times 50$ 的矩阵进行 PCA

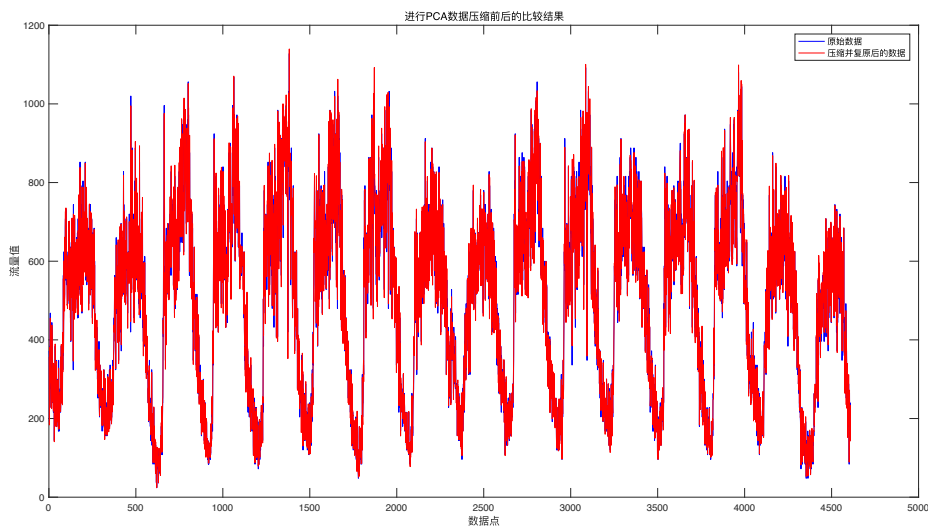


可以看出，进行 PCA 压缩与恢复后，基本上能够完整且不失真地恢复原来的数据。但是观察曲线可以发现，尖峰值得恢复效果不好，可以将尖峰值单独存储，这应该可以提高总体恢复的性能。

压缩比为1.900865，均方根误差为0.112033

可以看出，压缩比达到了 1.90，均方根误差也比较合理，说明该方法获得结果较为有效且成功。

## (2) 使用50×4608的矩阵进行 PCA



可以看出，进行 PCA 压缩与恢复后，基本上能够完整且不失真地恢复原来的数据，且尖峰值的恢复效果也较好。

压缩比为1.649957，均方根误差为0.084854

可以看出，进行这个步骤时所费时间较长，且压缩比略有下降。但是换来的是性能的提升——误差的下降和较好的高保真的尖峰值恢复效果。

## 3.3 K-means 聚类方法

### 3.3.1 题目要求

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。

### 3.3.2 方法原理

#### (1) 选取中心点

使用“密度法”选取初始点:取阈值 $\alpha_1 = 3$ ,对每个样本 $x(t)$ ,与其欧几里得距离小于 $\alpha_1$ 的点被定义为其“相邻点”。选取“相邻点”最多的点作为第一中心点。



随后，取阈值 $a_2 = 6$ ，把每个点的“相邻点”个数按照降序排列，从头到尾选取与之前中心点距离大于阈值的“相邻点”数较多的点作为随后的中心点。

### (2) 确定初始分类

使用“最近中心点”原则，对每个点进行分析，看它们离哪个中心点更近，那么就将其分到哪一类。

### (3) 修改中心点

对于每个分类，取其重心作为这一类新的中心点。

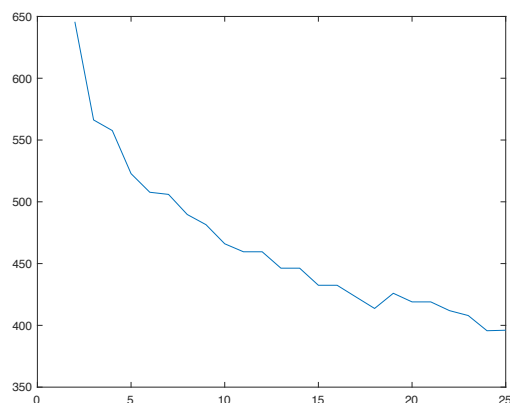
### (4) 停止迭代

如果修改前后的中心点的欧几里得距离小于阈值 $\delta = 0.01$ 就停止迭代。

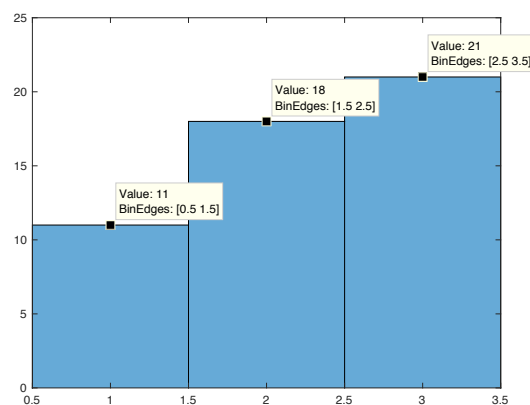
### 3.3.3 步骤与结果

提取出早高峰时段（早 7:00-9:00）的流量数据，并将其由 $24 \times 16 \times 50$ 的三维矩阵转化为 $384 \times 50$ 的二维矩阵。则这 50 个路口的特征为 384 维的向量。

首先先探索聚类数目和目标函数的关系如下图所示。



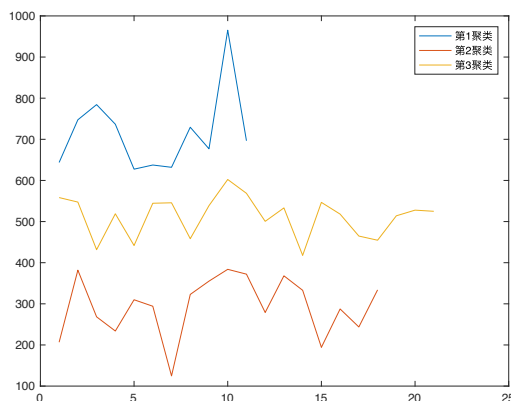
可以看出，目标函数在聚类数目从 2 变为 3 时出现了“肘点”，因此设置聚类数目为 3。运行程序，获得结果如下所示。



可以看出，有 11 个路口（4 10 14 15 18 19 26 27 42 47 49）被分在了第 1 聚类，有 18

个路口（2 5 6 16 17 20 21 22 28 29 31 35 36 37 43 46 48 50）被分在了第 2 聚类，有 21 个路口（1 3 7 8 9 11 12 13 23 24 25 30 32 33 34 38 39 40 41 44 45）被分在了第 3 聚类。

对每个路口的一天平均流量进行统计，并根据聚类在一张图像中进行表示，如下图所示。



由此可见，这个聚类结果能够较有效地反映出每个路口的属性特征（如每日平均流量），平均流量最大的一批（大于 600）被分在第 1 聚类，最小的一批（小于 400）被分在第 2 聚类，其余被分在第 3 聚类。通过这个验证步骤，可以看出之前使用的 Kmeans 方法的聚类效果较为合理。

## 四、选做任务

### 4.1 使用 SVR 方法进行交通流预测

#### 4.1.1 题目要求

自学至少一种新的交通流预测方法，仍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，与课堂讲授方法在预测精度方面进行对比分析。

#### 4.1.2 方法原理

SVR 方法即支持向量回归（Support vector regression）。以下为它的主要思想。

##### (1) 拟合函数

支持向量回归的拟合函数可表示为以下形式：

$$f(x) = \mathbf{w} \cdot \mathbf{x} + b$$

其中  $\mathbf{x}$  为向量， $f(x)$  为标量。

##### (2) 容忍回归值

支持向量回归具有一定的“容忍性”，即容忍回归值与目标函数间最多有  $\varepsilon$  的差。

$$\begin{cases} y_i - \mathbf{w} \cdot \mathbf{x}_i - b \leq \varepsilon + \xi_i \\ \mathbf{w} \cdot \mathbf{x}_i + b - y_i \leq \varepsilon + \xi_i^* \end{cases}$$

其中,  $\xi_i, \xi_i^* \geq 0, i = 1, \dots, n$

### (3) 优化目标

SVR 的优化目标是使得  $y - x$  曲线的斜率最小, 这个函数最接近于横轴, 也可以增加估计的鲁棒性。

$$\min_{\mathbf{w}, b} \frac{1}{2} \|\mathbf{w}\|^2 + C \sum_{i=1}^n (\xi_i + \xi_i^*)$$

### (4) 拉格朗日对偶法

最后, 我们使用“拉格朗日对偶法”

$$\begin{aligned} L = & \frac{1}{2} \|\mathbf{w}\|^2 + C \sum_{i=1}^n (\xi_i + \xi_i^*) - \sum_{i=1}^n \mu_i \xi_i - \sum_{i=1}^n \mu_i^* \xi_i^* \\ & + \sum_{i=1}^n \alpha_i (y_i - \mathbf{w} \cdot \mathbf{x}_i - \varepsilon - \xi_i) + \sum_{i=1}^n \alpha_i^* (\mathbf{w} \cdot \mathbf{x}_i - y_i - \varepsilon - \xi_i^*) \end{aligned}$$

求得优化参数后, 最终便可以求得 SVR 回归函数如下所示。

$$f(\mathbf{x}) = \sum_{i=1}^n (\alpha_i^* - \alpha_i) (\mathbf{x}_i \cdot \mathbf{x}) + b$$

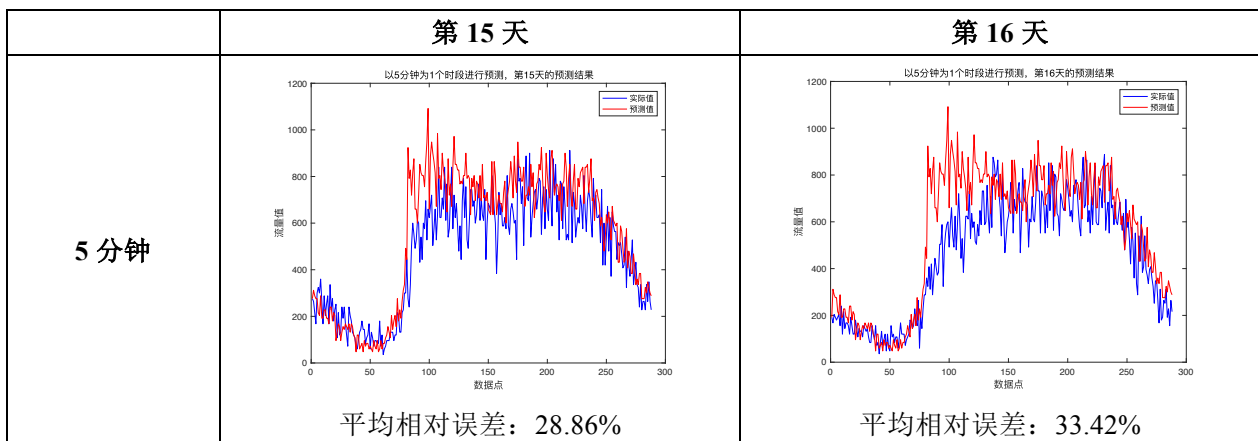
#### 4.1.3 步骤与结果

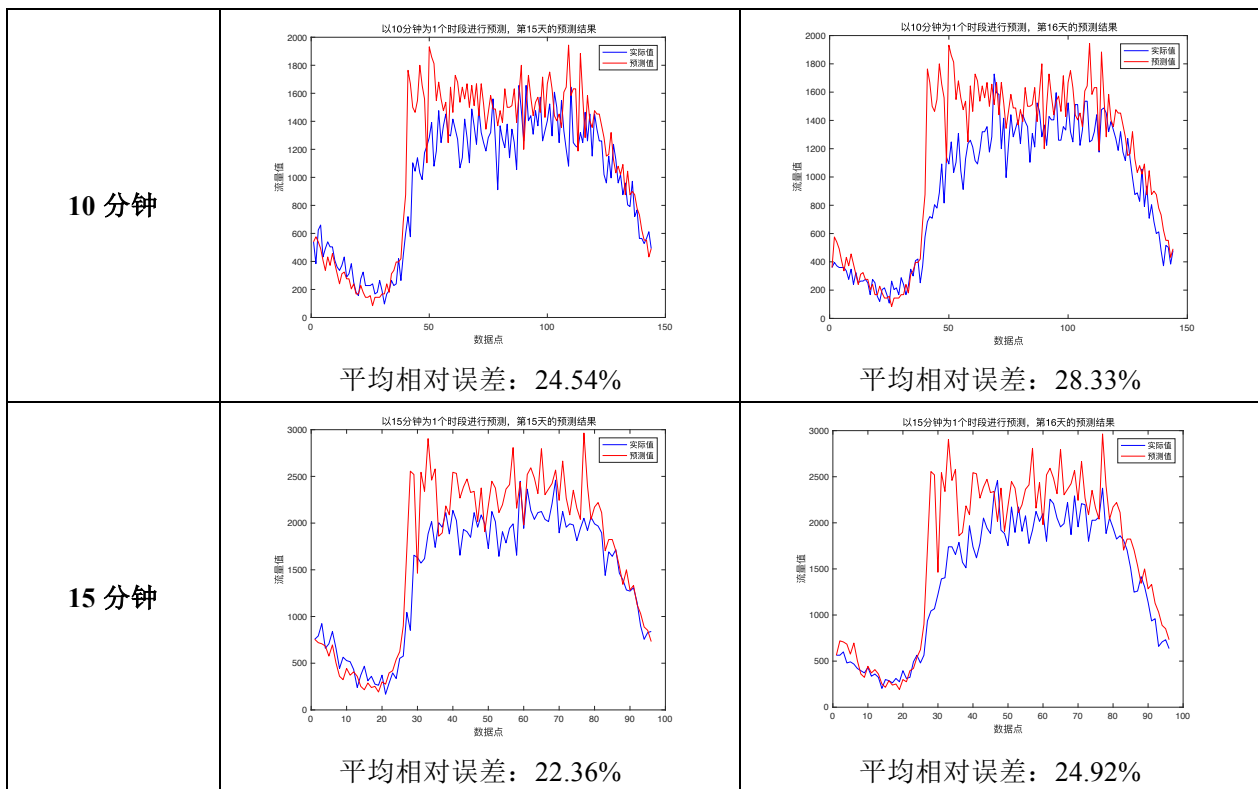
数据预处理的方法与必做任务 1 中的完全相同。在这里, 我们直接使用了由台湾大学林智仁(Lin Chih-Jen)教授等开发设计的 LIBSVM 软件包。并使用其中的 svmtrain 函数训练出模型。使用这个模型, 运行程序, 以第 10 个路口为例, 获得结果如下所示。

以 5 分钟为 1 个时段进行预测, 第 10 个路口: 第 15 天的预测平均相对误差 28.855914%, 第 16 天的预测平均相对误差为 33.421677%

以 10 分钟为 1 个时段进行预测, 第 10 个路口: 第 15 天的预测平均相对误差 24.536861%, 第 16 天的预测平均相对误差为 28.333664%

以 15 分钟为 1 个时段进行预测, 第 10 个路口: 第 15 天的预测平均相对误差 22.355755%, 第 16 天的预测平均相对误差为 24.918171%





通过以上结果可以看出：**SVR** 方法对原始数据的趋势预测较好，但是仍存在较大的误差。分析原因是：每天的交通流仍存在一定的随机性，在预测时只能对趋势进行较好拟合，但对于具体数值仍然会因为每天的特异性存在偏差。其次，如果拉长每次测量与预测的时间段，效果将会变好，这是因为延长时间段将会使得每个时间段的随机性减小（体现在图中是“锯齿”减少），从而减小了预测难度并增加了预测准确率。

#### 4.1.4 对比分析

将这个结果与之前使用考虑病态情况的多元线性回归方法进行对比，可以看出虽然 **SVR** 模型看起来更加复杂了，但是这并没有提升它的预测准确率，相反，在一些路口的预测准确率甚至还低于考虑病态情况的多元线性回归方法。

经过分析，我认为出现这个现象的原因是：**SVR** 模型的可调整参数比较多，对于每一个路口可能需要根据更多的样本，先行人工选择一个较为合适的参数组合，再进行预测的过程。而考虑病态情况的多元线性回归方法虽然原理较为简单、模型不是很复杂，但是在这个模型数据下面能够较好地拟合实际情况，因此获得的预测效果也比较好。

## 4.2 核主成分分析法

### 4.2.1 题目要求

自学概率主成分分析、贝叶斯主成分分析、核主成分分析等方法中的一种或者多种，对上述数据进行压缩和解压缩。与课堂讲授方法在压缩比、压缩精度等参数上进行对比分析。

### 4.2.2 方法原理

KPCA, 中文名称“核主成分分析”, 是对 PCA 算法的非线性扩展。言外之意, PCA 是线性的, 其对于非线性数据往往显得无能为力。受到支持向量机中通过核函数实现非线性变换的思想的影响, 核主成分分析方法得以发展。方法的基本思想是: 对样本进行非线性变换, 通过在变换空间进行主成分分析来实现再原空间的非线性主成分分析。利用与 SVM 中相同的原理, 根据可再生希尔伯特空间的性质, 在变换空间中的协方差矩阵可以通过原空间中的核函数进行运算, 从而绕过了复杂的非线性变换。算法的基本步骤分为通过核函数计算矩阵、主成分分析法进行数据压缩、数据恢复这三个。

#### [A] 通过核函数计算矩阵的具体步骤如下所述。

在这个步骤中, 使用高斯径向基函数(Radial Basis Function 简称 RBF), 这是一种种沿径向对称的标量函数。其形式为:

$$k(||x - x_c||) = \exp \left\{ -\frac{||x - x_c||^2}{2\sigma^2} \right\}$$

其中 $x_c$ 为核函数中心,  $\sigma$ 为函数的宽度参数, 控制了函数的径向作用范围。

#### [B] 核主成分分析法进行数据压缩的具体步骤如下所述。

为了能够直接使用主成分分析法的步骤与代码, 将以上步骤获得的结果直接设置为变量 X 进行之后步骤的操作。

##### (1) 数据归一化

为了统一样本的统计分布特性, 需要对样本进行归一化。直接调用 Matlab 函数 `zscore` 即可。

$$X\_zscore = zscore(X)';$$

##### (2) 确定特征值矩阵和特征向量矩阵

使用 Matlab 函数

$$[Q, \sim] = eig(A); D = eig(A);$$

得到特征值矩阵和特征向量矩阵。

##### (3) 计算主成分

在自变量降维去线性之后, 相互之间线性无关的自变量个数应缩减为  $m$  个, 且  $m$  应使得相对逼近误差可以接受。在本程序中, 相对逼近误差小于  $rerr$  视为可以接受, 即:

$$\frac{\sum_{i=m+1}^n \lambda_i}{\sum_{i=1}^n \lambda_i} < rerr$$

取前  $m$  个特征向量, 组成  $Q_m$  矩阵; 根据以下公式计算各样本数据在主成分方向上的投影:

$$y(t) = Q_m^T * X\_zscore(t)$$

##### (4) 输出结果

将以上步骤得出的结果置为函数的输出。其中:

输出量 `pcs` 代表各个主成分，每一列为一个主成分，即为 $Q_m$ 矩阵；

输出量 `cprs_data` 代表压缩后的数据，每一行对应一个数据点，即为 $y(t)^T$ 矩阵；

输出量 `cprs_c` 代表压缩时的一些常数，包括数据每一维的均值和方差等，第一行输出样本均值，第二行输出样本标准差。

利用以上三个变量应当可以恢复出原始的数据。

[C] 核主成分分析法进行数据恢复的具体步骤如下所述。

重建数据时，先利用以下公式求出规范化后的样本成分：

$$X\_zscore(t) = y(t)^T Q_m^T$$

随后利用以下公式求出恢复出来的数据：

$$X(t) = X\_zscore(t) * \delta + \bar{X}$$

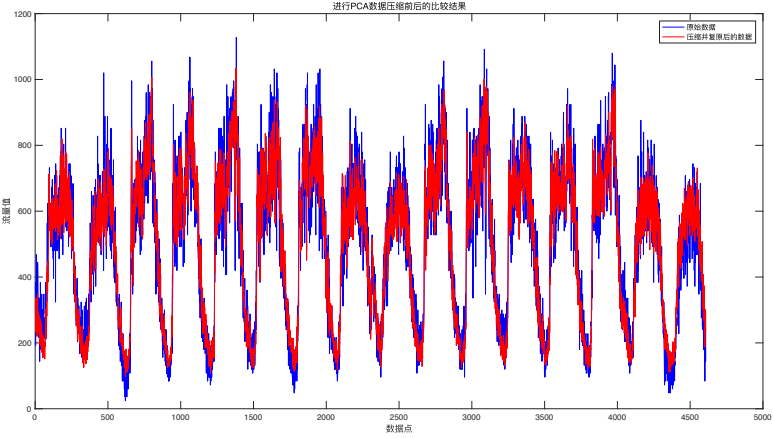
[D] 最终，对获得的结果进行压缩精度分析，用来评估的参数为均方根误差(RMSE)。

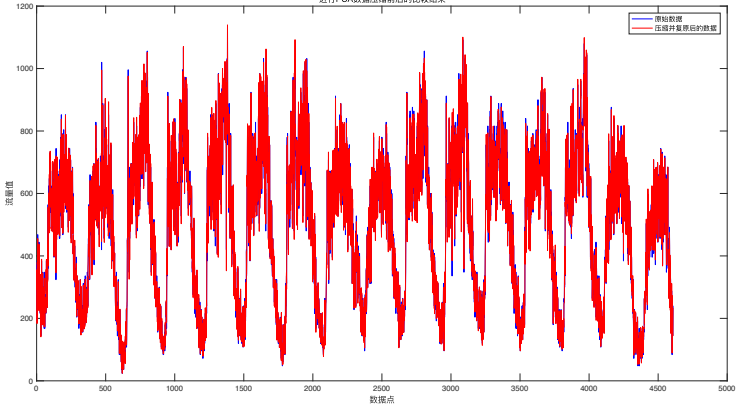
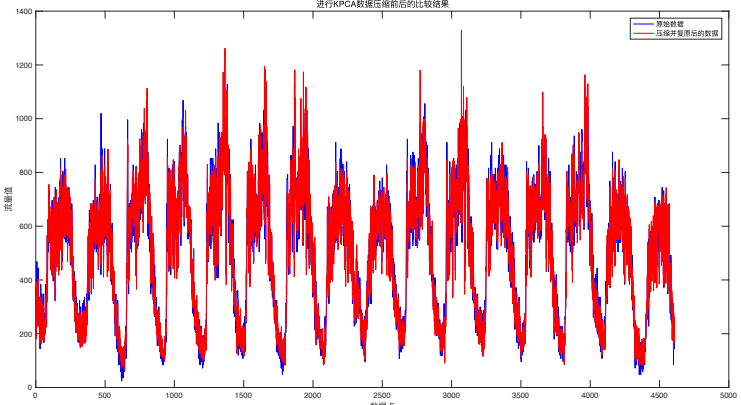
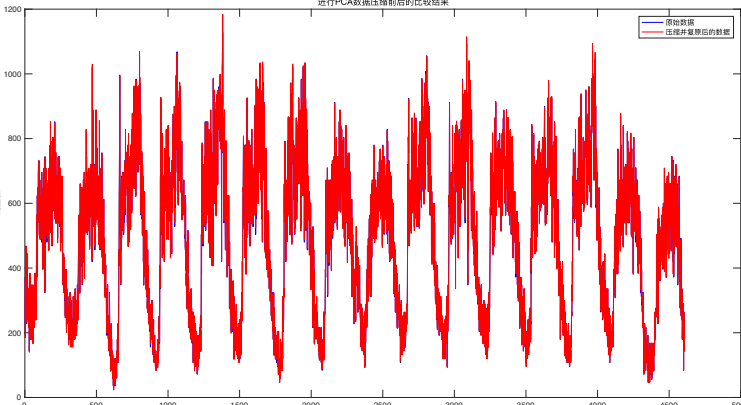
### 4.2.3 步骤与结果

设置参数 $rerr = 0.05$ ， $\sigma = 200$ 。

为了尝试不同压缩方式，并比较它们的压缩效果，本人在编写程序时，首先将每个路口 16 天的数据由 $288 \times 16 \times 50$ 的三维矩阵转化为 $4608 \times 50$ 的二维矩阵，即将 16 天的流量数值按照时间顺序连接在一起，这保证了 PCA 主成分分析程序能够对这个矩阵进行处理。

为了分析对 $4608 \times 50$ 的矩阵进行处理和对 $50 \times 4608$ 的矩阵进行处理的效果与性能差异，并和必做任务 2 中的结果进行对比，本人也进行了两次实验，并以路口 25 为例进行图像绘制，获得比较结果如下所示。

方法	矩阵维数	压缩比较图与数据分析
PCA	4608×50	<div></div>

	<b>50×4608</b>	<div>进行PCA数据压缩前后的比较结果</div>  <div>压缩比为1.649957, 均方根误差为0.084854</div>
<b>KPCA</b>	<b>4608×50</b>	<div>进行KPCA数据压缩前后的比较结果</div>  <div>压缩比为1.235587, 均方根误差为0.066426</div>
	<b>50×4608</b>	<div>进行PCA数据压缩前后的比较结果</div>  <div>压缩比为1.302401, 均方根误差为0.067298</div>

可以看出，进行 KPCA 压缩与恢复后，基本上能够完整且不失真地恢复原来的数据，且在 PCA 中恢复的不太好的尖峰值也能够较好地恢复出来。压缩比和均方根误差均较小，也处于相对合理的数值范围内。

4.2.4 对比分析

将 PCA 和 KPCA 的结果进行对比，可以得出以下的结论：

1、从运行效果上来看，使用相同的参数 ( $rerr = 0.05$ )，KPCA 的运行结果能够更加良好地恢复实际的数据，但是这是以较低的压缩比为代价的。

2、分析压缩比与压缩精度（指标为均方根误差），可以看出随着压缩比的增大，压缩精度相对减小，这说明在压缩过程中肯定会存在信息丢失的现象，并导致误差的出现。调整方法与参数可以改变压缩比的大小，但是压缩精度始终将会受到压缩比的牵制，因此在此考虑下不能为了节省存储空间无限制地增大压缩比。

## 4.3 SOM 聚类方法

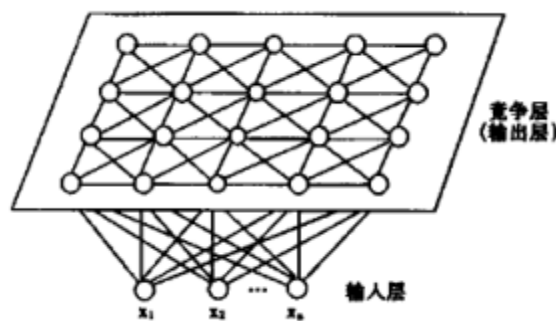
### 4.3.1 题目要求

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。

### 4.3.2 方法原理

SOM 聚类方法的灵感来自于人类的大脑结构。人们很早就了解到，人的大脑皮层是分区的，不同的区域对应着不同的功能，即对外部世界不同输入的响应。不仅不同类型的外界刺激在大脑皮层引起响应的区域有特定的规律，而且对于用一类型的刺激，大脑皮层对它的响应也明显表现出有组织的特点。高等动物大脑皮层对外界信号有规律的响应，有很大一部分是在不断接受外界信号刺激的过程下逐渐形成的，可以看作一种自学习的过程。

与前馈型神经网络不同，SOM 网络的神经元结点都在同一层上，在一个平面上呈规则排列。常见的排列形式包括方形网格排列或蜂窝状排列。样本特征向量的每一维都通过一定的权值输入到 SOM 网络的每一个节点上，构成下图所示的结构。



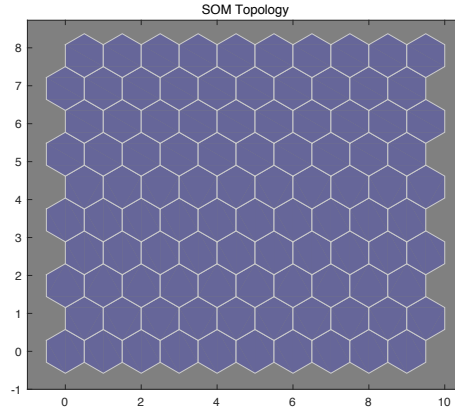
SOM 的自学习过程包括以下几个步骤：权值初始化、随机加入新样本、计算神经元响应、权值竞争学习、更新步长和邻域等。在经过了适当的自学习后，SOM 网络会表现出自组织现象。随着学习过程的进行，对于某个输入样本  $x$ ，对应的最佳相应节点会逐渐趋于固定，这个节点被称为该样本的像。这种自组织的过程完成的是从样本空间到二维平面上神经元网络的映射，这种映射是拓扑保持的，即在原空间中样本间的距离关系在只有有限个节点的平面网格上得到尽可能的保持。从这种意义上，SOM 是一种映射空间高度网格化的非线性特征变换。



### 4.3.3 步骤与结果

#### (1) 对时段进行聚类

原始数据集 flow\_50link 大小为 $288 \times 16 \times 50$ ，目标是建立一种分类，将数据聚类为一个 $10 \times 10$ 的结构，每个分类对应一种 50 个路口的流量分布模式。该神经结构如下图所示。

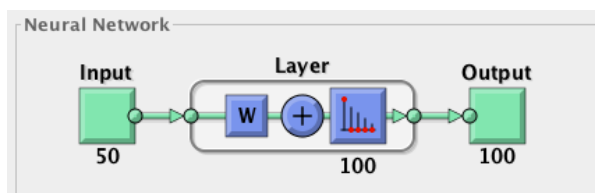


设置 SOM 输出层（竞争层）为 $10 \times 10$ 的二维结构，即将数据集划分为 100 个类。将 $288 \times 16 \times 50$ 的原始数据集进行降维并拆分大小为 $4320 \times 50$ 的训练集和 $288 \times 50$ 的测试集。

使用 MATLAB 中自带的 SOM 训练函数进行训练，设置迭代次数为 100 次，代码如下：

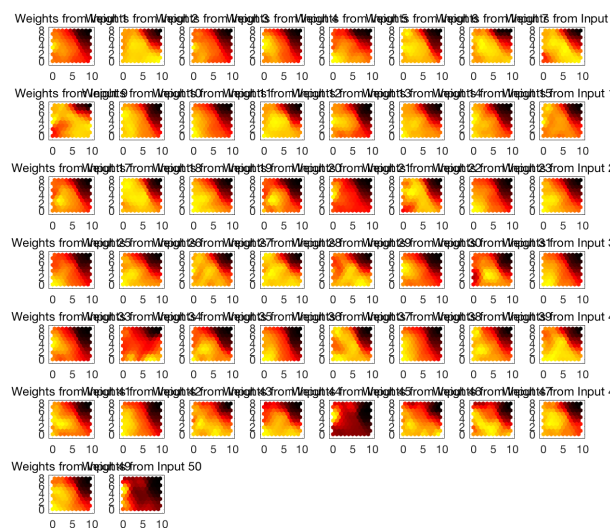
```
%% 设置SOM网络
N=10;
range=ones(50,2);
range(:,1)=zeros(50,1);
net=newsom(range,[N N]); %建立som神经网络: 10*10
net.trainParam.epochs=100; %设置迭代次数
net=train(net,train_a); %利用训练值进行训练
w=net.iw{1}; %对应神经元的权值分量，获得训练结果，为100*50，即每个类50个路口流量情况
```

SOM 学习过程如下所示：

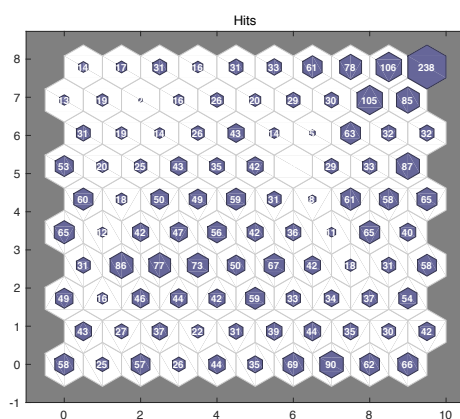


依据 SOM 算法的基本原理，本实验中神经元数量为 100 个，每个神经元维数为 50，即每个神经元对应于一种分类，该分类中 50 个路段的流量具有一定区别于其他分类的特性。

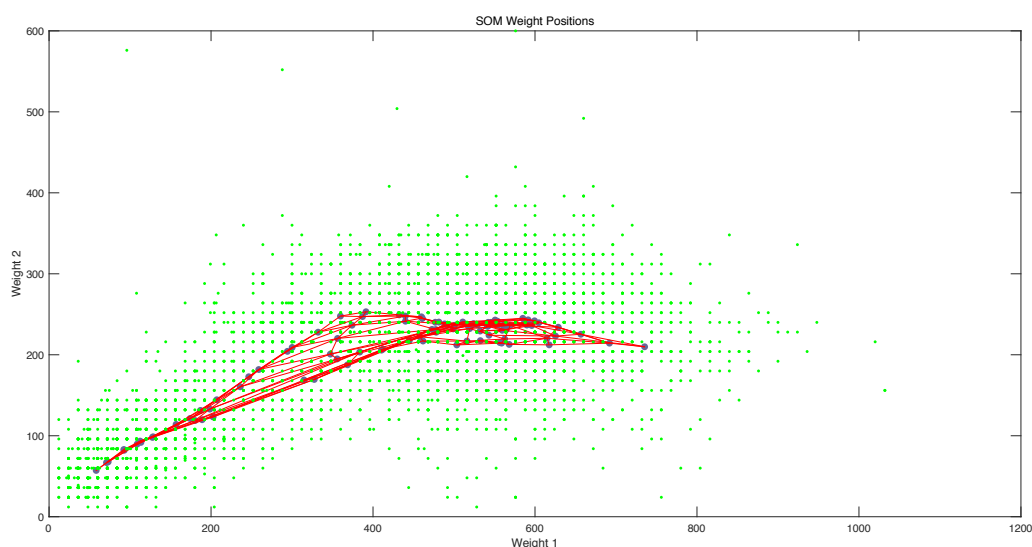
进行迭代聚类后，得到的 SOM( $10 \times 10$ )神经元 50 个权值向量大小示意图如下图所示。



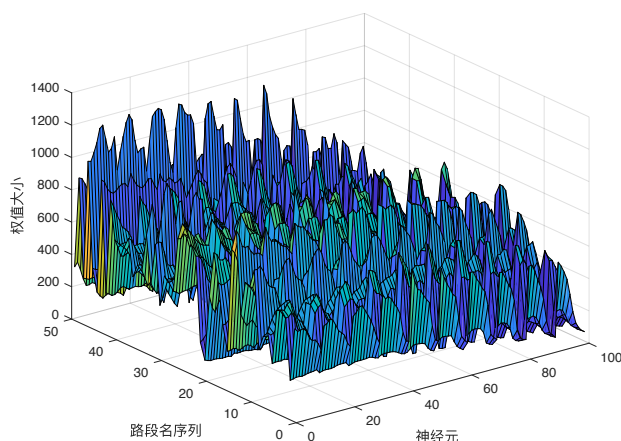
各个神经元对应的训练点数量统计如下图所示。



神经元位置与原始训练集点位置关系如下图所示。其中，绿点为训练集数据点在二维平面上的投影，蓝点和红点表示 SOM 网络在二维平面上的投影。



最终得到的 100 个神经元对应的 50 维权值向量权值图如下图所示。

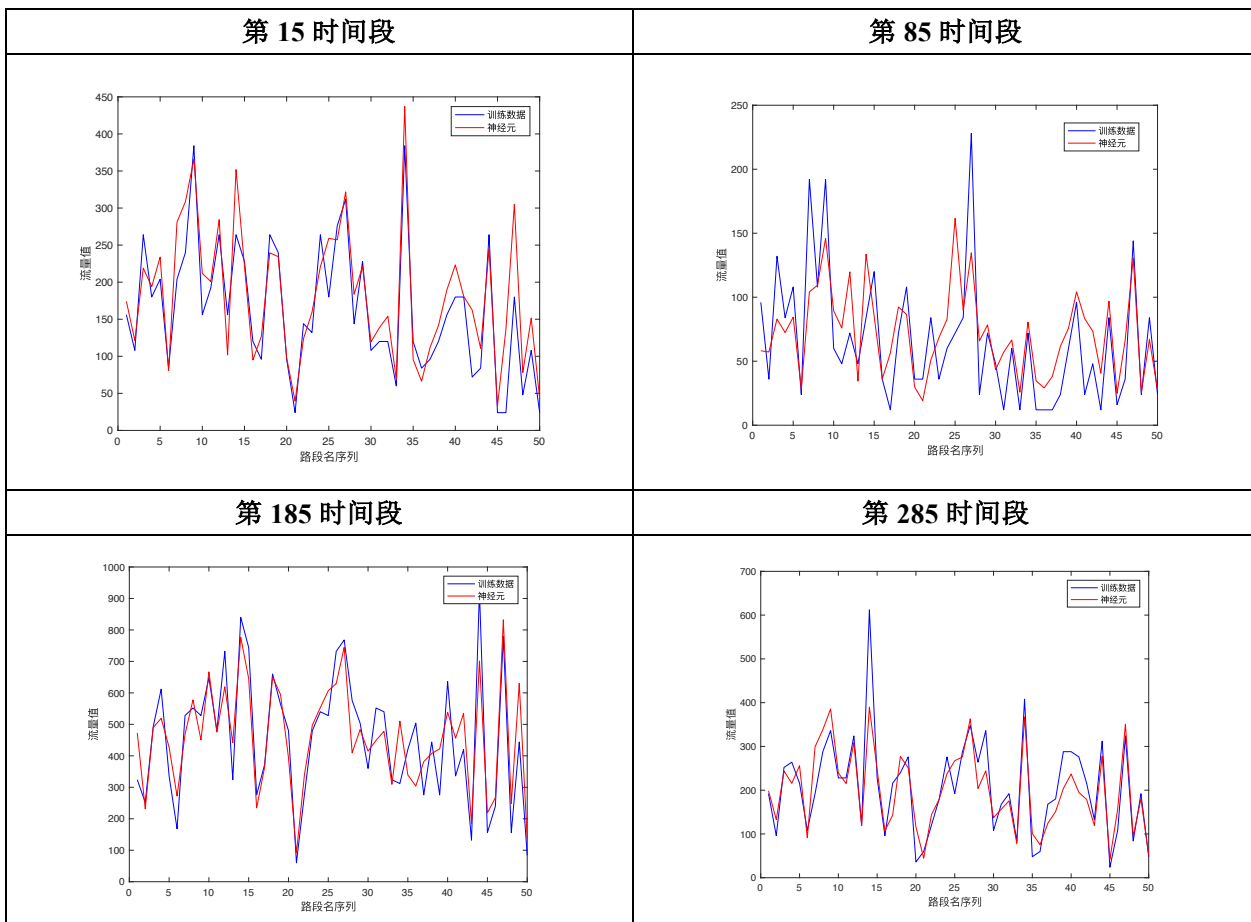


计算按照分类得到的路段流量与实际流量之间的均方根误差，获得结果如下所示。

聚类结果的均方根误差为：2.242852e-01

误差量达到了 22.4%，但是由于聚类方法与其他方法不同，它是综合了所有同类样本的信息而非每个样本独立的信息，因此代表的是样本的总体特征而非具体数值，因此这个误差值是可以接受的。

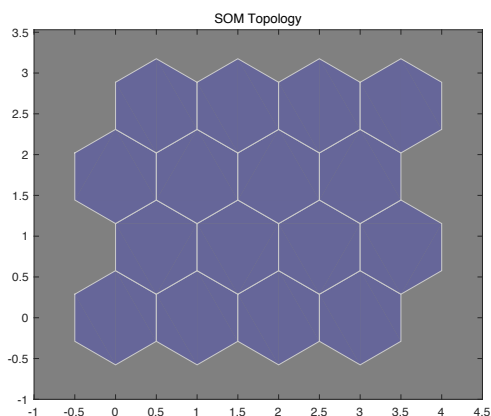
为了测试最终获得的聚类神经元是否真的能够代表实际的流量特征，我随机抽取了几个时间节点（第 15、85、185、285 个时间段），将神经元与各路口那一时刻的实际流量进行对比，获得结果如下所示。



可以看出，聚类结果基本上能够反映每个路口的实际属性，从而能够较好地预测第 16 天的交通状况，说明该聚类结果较为科学。

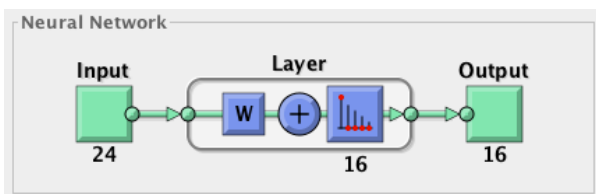
## (2) 对路口进行聚类

取同一个时段（7:00~9:00）的流量数据，目标是建立一种分类，将数据聚类为一个 $4 \times 4$ 的结构，每个分类对应该时段这一路口的流量走向趋势。该神经结构如下图所示。



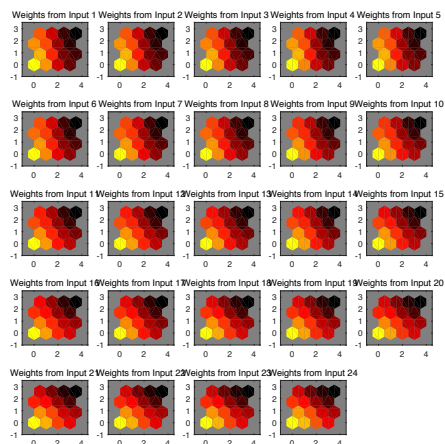
设置 SOM 输出层（竞争层）为 $10 \times 10$ 的二维结构，即将数据集划分为 100 个类。将 $288 \times 16 \times 50$ 的原始数据集进行降维并拆分大小为 $4320 \times 50$ 的训练集和 $288 \times 50$ 的测试集。

使用 MATLAB 中自带的 SOM 训练函数进行训练，设置迭代次数为 100 次，SOM 学习过程如下所示：

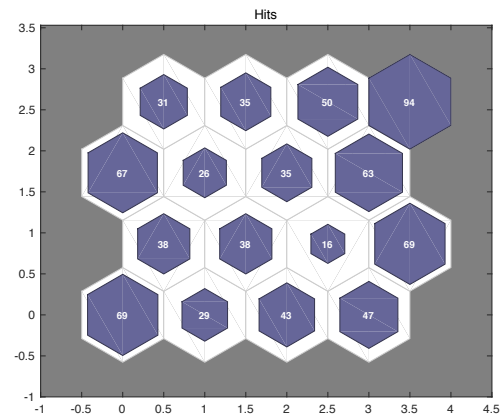


依据 SOM 算法的基本原理，本实验中神经元数量为 16 个，每个神经元维数为 24，即每个神经元对应于一种分类，该分类中 24 个时间段的流量数具有一定区别于其他分类的特性。

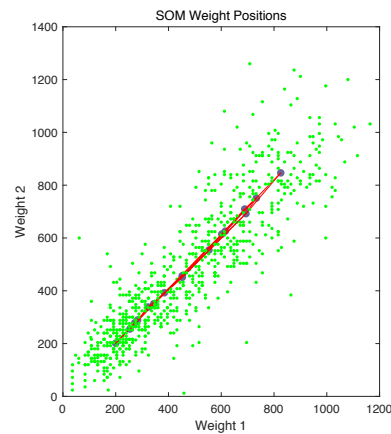
进行迭代聚类后，得到的 SOM( $10 \times 10$ )神经元 24 个权值向量大小示意图如下图所示。



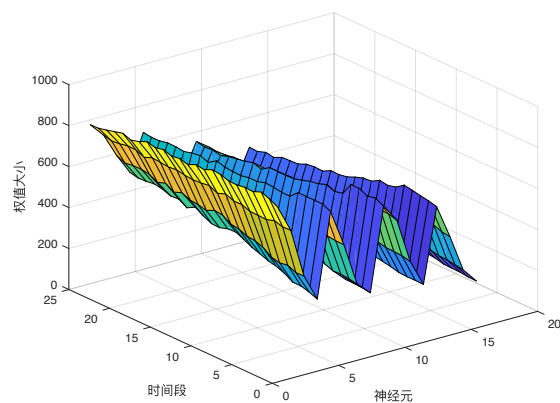
各个神经元对应的训练点数量统计如下图所示。



神经元位置与原始训练集点位置关系如下图所示。其中，绿点为训练集数据点在二维平面上的投影，蓝点和红点表示 SOM 网络在二维平面上的投影。



最终得到的 16 个神经元对应的 24 维权值向量权值图如下图所示。



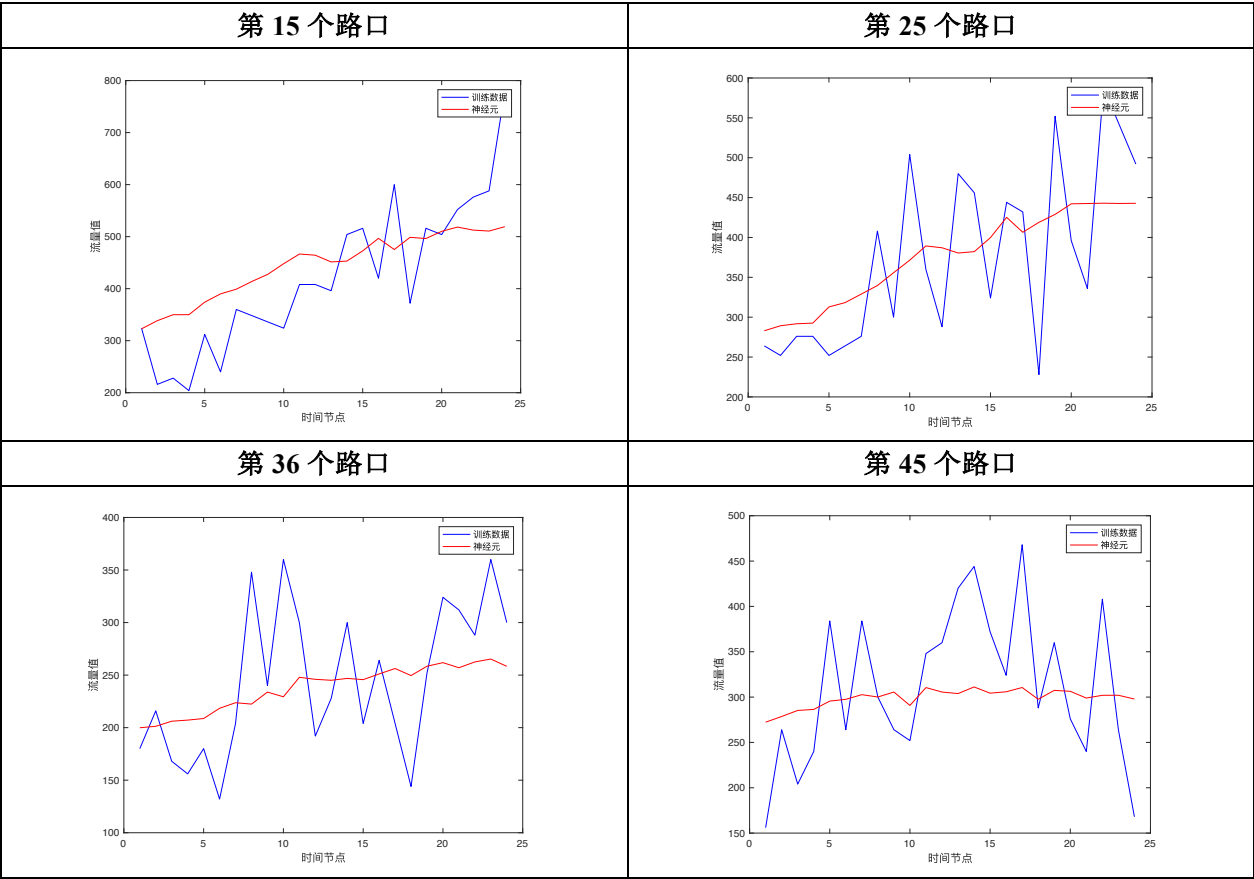
计算按照分类得到的路段流量与实际流量之间的均方根误差，获得结果如下所示。

聚类结果的均方根误差为：3.201051e-01

误差量约为 32.0%。

为了测试最终获得的聚类神经元是否真的能够代表该时段各路口实际的流量特征，我

随机抽取了几个路口（第 15、25、36、45 个路口），将神经元与该路口高峰时段的实际流量进行对比，获得结果如下所示。



从以上结果可以看出，SOM 聚类基本可以将高峰时段内具有同样趋势的路口聚类在一起，因此可以通过聚类神经元预测第 16 天的交通趋势。但是由于每天即使是同一时间段的流量数值也存在着较强的随机性，因此对于随机性（震荡性）较强的时间节点，最终的预测结果会产生一定的误差。

## 五、总结与感受

这是一次令人激动的课题经历。因为在此之前，我们对系统工程相关知识的理解还完全停留在理论和一些简单地数据中，而通过本次课题，我们揭开了系统工程与现实生活交通数据的联系，且通过实践对系统工程的理论有了更加深入的了解和认识。当我看到在改进方法或者参数后，回归聚类效果变好了、误差值减小了，我的激动之情难以言表。

尽管在完成课题的过程中还出现了一些不太尽如人意的地方，需要我们在今后的学习中进行进一步探索与研究，例如线性回归预测的误差较大、对路口进行聚类分析时能够反映出路口的均值特征，但是对其流量走向趋势并不能很好地拟合与表示。这些问题的解决可能超出了课程的要求，在今后的进一步学习中我也将会时时留心。

在最后，感谢老师和助教为我们精心设计的大作业题，让我们有机会接触到并深入探索系统工程与现实生活数据的关系，在完成课题的过程中更加深入地理解理论。

## 参考文献

- [1] 张学工. 模式识别（第三版）[M]. 北京: 清华大学出版社, 2010.
- [2] <http://www.csie.ntu.edu.tw/~cjlin/libsvm/>
- [3] <http://blog.csdn.net/ws998689aa/article/details/40398777>
- [4] [http://blog.csdn.net/zdy0\\_2004/article/details/50858864](http://blog.csdn.net/zdy0_2004/article/details/50858864)