中国研究生创新实践系列大赛

“华为杯”第十六届中国研究生数学建模竞赛

学　　校	桂林电子科技大学
参赛队号	**19105950041**
队员姓名	**1.　杨鲜**
	2.　郑朋
	3.　张瑞

中国研究生创新实践系列大赛
“华为杯”第十六届中国研究生数学建模竞赛

汽车行驶工况构建（D 题）

摘　　要：

随着汽车在我国越来越普及，由汽车引起的一系列问题也随之产生，比如新车排放达标率低，汽车排放污染状况日益严重等。为了有效地控制城市汽车污染物的排放，掌握汽车在实际路况上的排放量显得越来越重要。

针对问题一，根据题目给出的异常数据类型对文件中给出的原始数据进行筛选，运用统计学分析方法得出原始数据的分布规律，将一些由于 GPS 信号丢失，车辆长期停车等原因造成的异常数据进行剔除，保证数据的有效性并给出各个文件经过处理后最终的有效记录数。

针对问题二，根据运动学片段的定义并结合原始数据的分布规律，将预处理后的有效数据划分为一系列运动学片段，选取可以描述和评价这些片段的特征参数并进行相应的计算。这些特征值可以表征车辆在这一段时间内的行驶特点，是对这些运动学片段进行分类的依据。对所有的运动学片段进行有效筛选并给出各数据文件最终得到的运动学片段数量。

针对问题三，先对运动学片段的特征参数进行标准化得到无量纲的特征矩阵，再利用 PCA 主要成分分析做降维处理提取出五个主要成分。将降维后得到的五个主要成分作为研究变量，运用 K-means 聚类算法对所有运动学片段进行分类，得到四类代表不同车辆行驶特征的运动学片段库。然后采用基于总体特征参数偏差最小的片段选取方法对备选工况片段进行遴选，最终构建出行驶工况曲线，并对最终构建出来的汽车行驶工况进行有效性的验证。

关键词：污染物排放；汽车行驶工况；运动学片段；PCA 主成分分析；K-means 聚类

目　录

一、问题的重述

随着我国经济水平的快速发展，汽车的数量得到了快速地增长，但是汽车带来的一系列问题也随之而来，比如汽车污染物的排放。所以研究汽车在实际道路上行驶时的燃油量以及污染物排放量变得越来越重要。

汽车行驶工况是描述汽车行驶的速度-时间曲线，体现了汽车在道路上行驶的运动学特征，它可以用来评估燃油消耗量、汽车污染物排放和新型车的开发等。但由于我国的各个城市的发展程度、气候条件和交通状况的不同，所以基于城市自身的汽车行驶数据进行城市汽车行驶工况的构建研究变得越来越重要，这样才能获得更真实的燃油消耗和污染排放水平。运动学片段是指汽车从怠速状态开始至下一个怠速状态开始之间的车速区间，通常包含怠速、加速、减速和匀速四个状态。典型的运动学片段如下图 1 所示。

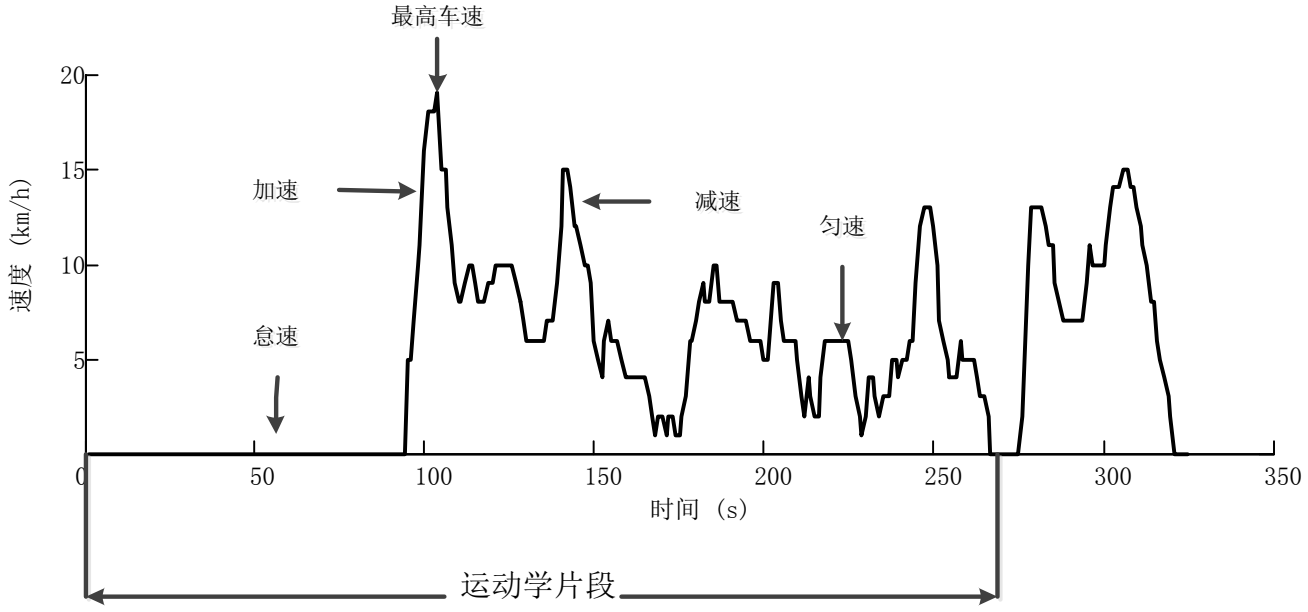

图 1 运动学片段的定义

在本文的研究中，构建汽车行驶工况是我们的研究重点，针对采集设备直接采集得到的汽车行驶的原始数据，我们需要解决以下问题：

（1）根据不良数据的类型，对原始数据进行预处理，剔除异常数据，保证数据的可靠性和有效性，并给出文件处理后的记录数；

（2）对原始数据划分为一系列运动学片段，并对运动学片段进行筛选，得到有效的运动学片段数量；

（3）对运动学片段的特征参数进行建模，利用 PCA 对特征参数做降维处理，得到主要成分，作为运动学片段分类的依据；

（4）运用 K-means 聚类对运动学片段进行分类，选取最优工况片段进行组合，得到最终工况曲线；

（5）分别计算出汽车行驶工况曲线与该城市所采集数据源的特征参数值，验证所构建的汽车行驶工况的合理性。

二、模型假设

为了使得问题更易于理解，我们做出以下合理假设：

- 假设 1，由于 GPS 信号丢失，造成所提供数据中的时间不连续，进而导致不连续时间段的末尾时刻无法计算瞬时加速度，故作为无效数据处理。
- 假设 2，长期停车包含停车不熄火等候人、停车熄火了但采集设备仍在运行等情况。若停车时间超过 1 小时，则视为长期停车。
- 假设 3，如果在一个运动学片段中存在超过 300 秒的不连续数据，则视为无效运动学片段。
- 假设 4，如果运动学片段的运行时间小于 20 秒，那么则认为这个运动学片段无效。

三、符号说明

本文建立模型的过程中主要涉及以下符号，符号及说明如下[1]：

表 1 运动学片段特征值及其说明

序号	符号	描述	单位
1	T	运行时间	s
2	T_a	加速时间	s
3	T_d	减速时间	s
4	T_e	匀速时间	s
5	T_i	怠速时间	s
6	$V_{\max}$	最大速度	km/h
7	V_m	平均速度	km/h

8	V_{mr}	平均行驶速度(除怠速外)	km/h
9	V_{sd}	速度标准差	km/h
10	a_a	平均加速度	m/s^2
11	a_d	平均减速度(绝对值)	m/s^2
12	a_{sd}	加速度标准差	m/s^2
13	$a_{\max\_a}$	最大加速度	m/s^2
14	$a_{\max\_d}$	最大减速度	m/s^2
15	S	运行距离	m
16	P_i	怠速时间比	%
17	P_a	加速时间比	%
18	P_d	减速时间比	%
19	P_e	匀速时间占比	%

四、模型的建立与求解

下图为汽车行驶工况构建的基本流程图，主要包含数据与处理，运动学片段的提取，构建汽车行驶工况曲线以及工况曲线有效性的说明等部分。数据预处理主要是由于采集设备采集的原始数据包含异常数据，为保证实验数据的有效性和实验结果的真实性，需要对异常数据进行处理，排除异常数据。

运动学片段是指汽车从怠速状态开始至下一个怠速状态开始之间的车速区间，通常包含怠速、加速、减速和匀速四个状态[2]。运动学片段是构成最终汽车行驶工况曲线的基本单位，针对运动学片段的特征参数进行建模，下文的分析都是基于此模型进行。汽车行驶工况曲线根据运动学片段的参数特征，对所有运动学片段进行聚类，每一类都是包含相同交通特征的运动学片段，然后从每一类中选出备选片段进行组合，得到最终的汽车行驶工况。最后将汽车行驶工况的特征和整体实验数据的对应特征进行比较，验证汽车行驶工况的有效性。

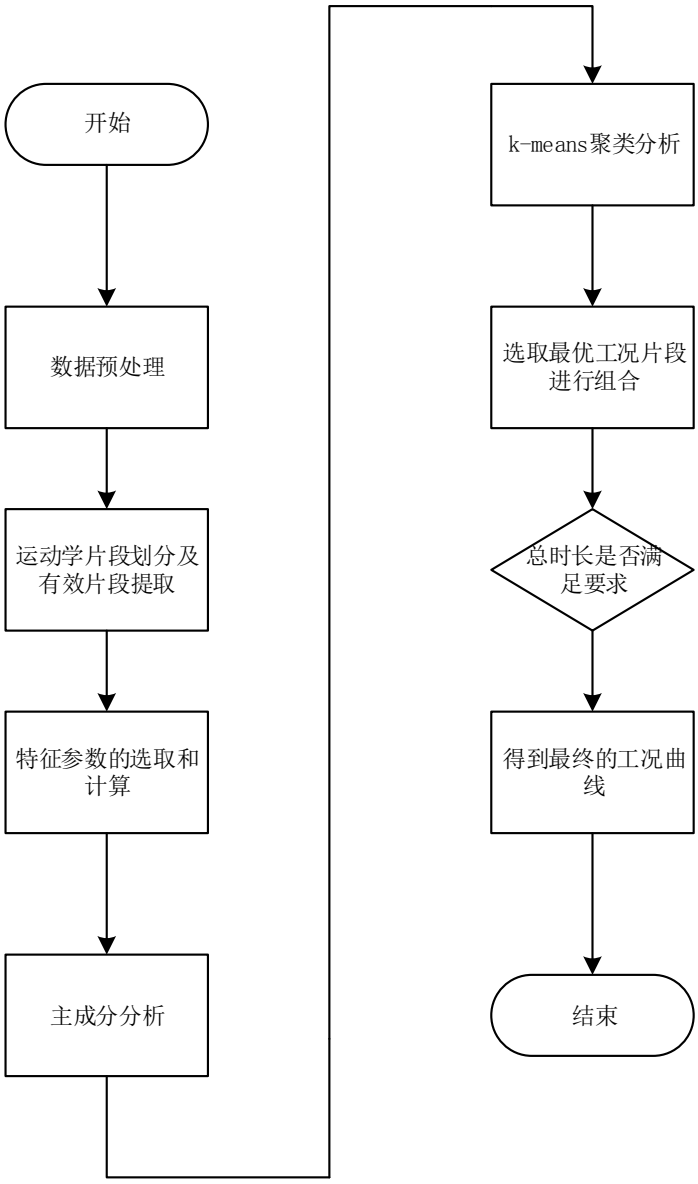

图 2 汽车行驶工况构建的基本流程图

4.1 数据预处理

题目提供了三个文件，每个数据文件为同一辆车在不同时间段内所采集的数据。文件中的数据是由汽车行驶数据的采集设备直接记录的原始采集数据，往往包含一些不良数据值，我们需要对数据做预处理：

（1）剔除加、减速度异常的数据

利用预处理后的数据，对问题进行求解分析，这样可以保证数据的有效性和结果的真实性，得出的结果才会更加接近于实际情况。

题目中给出信息，普通轿车一般情况下，从 0 加速到 100km/h 所需要的加速时间大于 7 秒。经过计算，也就是说汽车加速时的最大加速度不能大于 3.96825m/s^2。同时给出的还有紧急刹车的最大减速度在 7.5m/s^{2}和 8m/s^{2}之间。综上所述，对加速度的约束条

件就是加速度要大于-8m/s^2 并且小于 3.96825m/s^2。根据加速度的约束条件对原始数据进行第一步预处理，剔除的无效数据的记录数为 755 条。

（2）剔除长期停车所采集的异常数据[3]

题中给出长期停车的情况有停车不熄火等候人，停车熄火了但采集设备仍在进行等两种情况，将长期停车采集的数据当作异常数据，需要予以剔除。通过分析，本文将长期停车定义为汽车速度等于零但是汽车转速不为零的情况。通过 Python 编程对数据进行分析，得到一段很明显的属于长期停车情况的异常数据，如下图 3 所示。

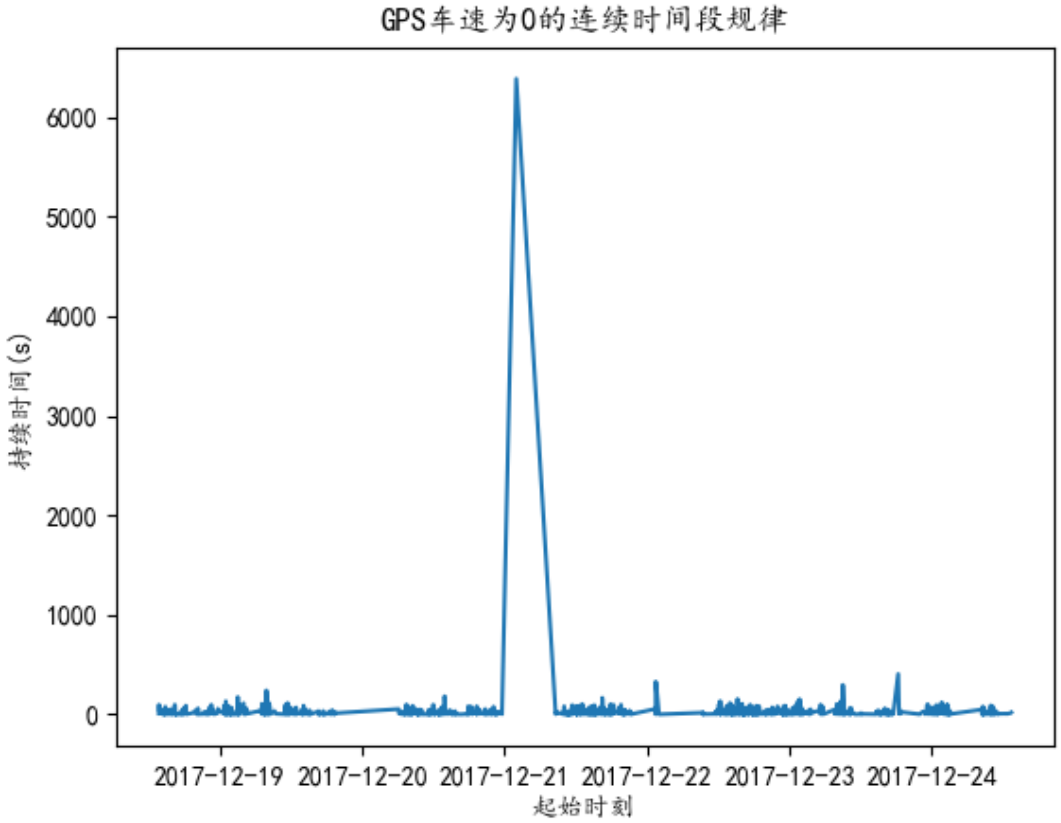

图 3 长期停车的异常数据

（3）一般认为怠速时间超过 180s 为异常情况

GPS 车速不大于 10km/h 的行驶数据视为怠速处理，对于连续怠速时间超过 180s 的异常情况，从前面部分截取掉超过 180 秒的数据。如从 A 点开始到 B 点结束，连续怠速时间为 300 秒，则只保留靠近 B 点的 180 秒数据。以数据文件 1 为例，对 GPS 车速大于等于 0 且小于 10km/h 的连续怠速时间规律如图 4 所示，剔除了 985 条无效记录。

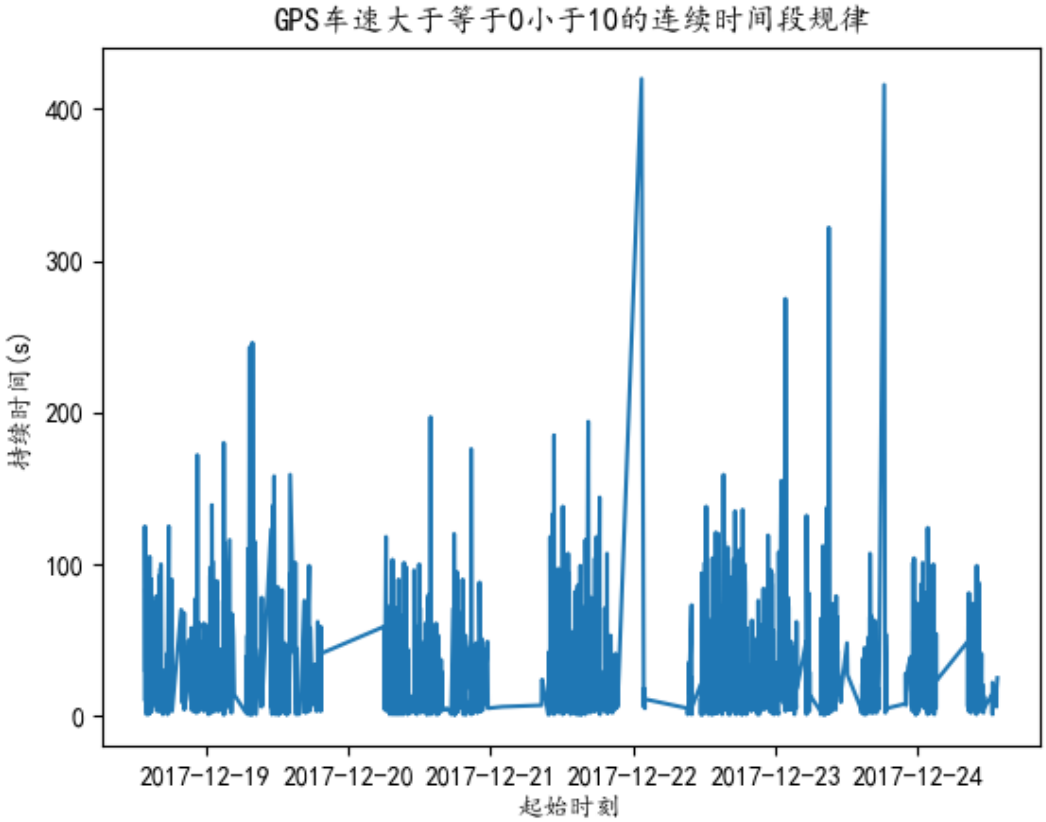

图 4 怠速时长大于 180 秒的异常数据

小结：经过数据预处理之后，文件 1 得到的最终记录数为 177603 条。同理，也可以得到文件 2、文件 3 的记录数。文件 2 得到的最终记录数为 138599 条，文件 3 得到的最终记录数为 157846 条，如下表 2 所示。

表 2 各文件数据预处理后的记录数

文件名	剩余记录数
文件 1	177603
文件 2	138599
文件 3	157846

4.2 运动学片段的提取

对预处理之后的数据按照运动学片段的原理进行划分之后会存在一些短时间片段和无效片段，会导致后续特征值分析的结果产生很大的偏差，所以按照下面的规则对其进行筛选：（以下规则介绍均以文件 1 为例）

（1）一个运动学片段的全部时长若小于 20 秒，则将该片段剔除[1]

基于运动学片段的定义将预处理之后的数据划分为一系列的运动学片段。经过数据处理之后发现文件 1 运动学片段的时长小于 20 秒的记录数有 309 条，将这些不满足条件的运动学片段当做无效片段予以剔除。

（2）如果一个运动学片段中相邻有效数据时间间隔超过 300 秒，认为该片段无效

如果一个运动学片段中相邻有效数据时间间隔超过 300 秒，我们认为这个运动学片段的数据不具有连续性，不能作为有效运动学片段处理。经过数据分析和处理，发现文件 1 存在这种现象的原始数据记录数有 116 条，对这些记录数同样予以排除。

（3）怠速时间占总运行时间的比例超过 80%，认为该片段无效

对每个运动学片段而言，当怠速时间占总运行时间的比例超过 80%时，认为这个运动学片段是无效的。例如，某个运动学片段的总运行时间为 100 秒，怠速时间为 81 秒，怠速时间占总运行时间的比例超过 80%，故认为该运动学片段无效。经过数据分析和处理，发现文件 1 存在这种现象的运动学片段有 89 条，对这些运动学片段予以排除。

小结：经过数据预处理以及有效运动学片段提取之后，文件 1 得到的最终的运动学片段的数量为 1103 条。同理，也可以得到文件 2 和文件 3 的运动学片段的个数。文件 2 得到的最终的运动学片段的数量为822 条，文件3 得到的最终的运动学片段的数量为786 条。

表 3 各文件有效的运动学片段数

文件名	有效运动学片段数
文件 1	1103
文件 2	822
文件 3	786

4.3 问题模型的建立与求解

4.3.1 建模思路

车辆行驶工况构建是从每一类运动学片段库中按照特定的选取规则挑选出符合要求的候选片段进行组合拼接形成最终工况的。在每一类运动学片段库中，每一个运动学片段都包含了若干个速度-时间点，每个运动学片段也都有各自的运动学特征[5]。要描述和评价一个运动学片段，需要选取相应的运动学特征参数，这些运动学特征参数能够尽可能全面地将运动学片段根据其运行状态进行描述并形成基于运动学片段的数学模型。

基于运动学片段构建的行驶工况需要通过与整体样本的特征参数进行比较来判定其有效性和合理性，所以特征参数的计算包括三个部分：单个运动学片段的特征参数计

算、最终构建出的行驶工况的特征参数计算以及整体样本的特征参数计算。

4.3.2 模型建立

1.单个运动学片段的特征参数计算[6]：

（1）加速度作为一个重要的评价参数，不在数据采集设备能够采集的范围内，只能通过以下公式计算得到：

$$a_{i,i+1}=\frac{V_{i+1}-V_i}{t_{i+1}-t_i}\cdot\frac{1000}{3600}=\frac{V_{i+1}-V_i}{3.6},i=1,2,3...k-1 \tag{4-1}$$

式中$a_{i,i+1}$为第i秒和第$i+1$秒的加速度，单位为m/s^2；和分别为第i秒和第$i+1$秒的速度，单位是km/h；t_i和t_{i+1}分别是第i秒和第$i+1$秒的时刻，单位是s；k表示该运动学片段所有数据点的个数。

（2）特征参数：T,T_i,T_a,T_d,T_e

T = 该运动学片段的总点数 (4-2)

T_i = 速度为 0 的数据点的总个数 (4-3)

T_a = 该运动学片段中加速度不小于 0.1m/s^2 的总点数 (4-4)

T_d = 该运动学片段中加速度不大于 0.1m/s^2 的总点数 (4-5)

$$T_e=T-T_i-T_a-T_d \tag{4-6}$$

（3）特征参数：$V_{\max},V_m,V_{mr},V_{sd}$

$$V_{\max}=\max\{V_i,i=1,2,...,k\} \tag{4-7}$$

$$V_m=S/T \tag{4-8}$$

$$V_{mr}=S/(T-T_i) \tag{4-9}$$

$$V_{sd}=\sqrt{\frac{1}{k-1}\sum_{i=1}^{k}(V_i-V_m)^2},i=1,2,...,k \tag{4-10}$$

（4）特征参数：S

由于数据采样频率为$1Hz$，运动学片段的运行距离等于该片段内所有点的速度之和：

$$S=\sum_{i=1}^{k}V_i,i=1,2,...,k \tag{4-11}$$

（5）特征参数：$a_a, a_d, a_{sd}, a_{\max\_a}, a_{\max\_d}$

$$a_{\max\_a} = \max\{a_i, i = 1,2,...k-1\} \tag{4-12}$$

$$a_a = \frac{sum\{a_i \mid a_i \geq 0.1, i = 1,2,...,k-1\}}{T_a} \tag{4-13}$$

$$a_{\max\_d} = \left|\min\{a_i, i = 1,2,...k-1\}\right| \tag{4-14}$$

$$a_d = \frac{\left|sum\{a_i \mid a_i \leq 0.1, i = 1,2,...,k-1\}\right|}{T_d} \tag{4-15}$$

$$a_{sd} = \sqrt{\frac{1}{k-1}\sum_{i=1}^{k} a_i^2}, i = 1,2,...,k \tag{4-16}$$

2.运动学片段综合特征参数计算：

运动学片段的综合特征值就是每一类片段库中的运动学片段的平均统计量，能综合反映该类片段库中运动学片段的行驶特征，主要计算方法如下：

（1）综合特征值T, T_i, T_a, T_d, T_e, S

这6个综合特征值都是由所有动学片段对应项的之和与运动学片段的个数相除得到的，下面以 T 为例说明计算的方法。

$$T = \frac{\sum_{i=1}^{n} T(i)}{n} \tag{4-17}$$

n 为该片段库中运动学片段的总个数。

（2）综合特征值$V_{\max}, a_{\max_a}, a_{\max_d}$

以上 3 个特征为该类片段库中所有运动学片段对应项的最大值，以$V_{\max}$为例说明计算方法。

$$V_{\max} = \max\{V_i, i = 1,2,...,k\} \tag{4-18}$$

4.3.3 求解方法

1.主成分分析方法

在我们选取的 15 个运动学片段特征参数中，各个特征参数之间存在着显性或者隐性的相关关系，如果采用所有的参数作为聚类的依据，往往还会带来错误的结果，因而需要采取一定的措施将原本重复的变量进行剔除并形成能够尽可能全面反映原始变量信息的新变量。主成分分析的步骤：

假设样本数为n，特征参数个数为p，则可以得到$n \times p$维的矩阵Z：

$$Z=\begin{bmatrix} z_{11} & z_{12} & \cdots & z_{1p} \\ z_{21} & z_{22} & \cdots & z_{2p} \\ \vdots & \vdots & \vdots & \vdots \\ z_{n1} & z_{n2} & \cdots & z_{np} \end{bmatrix}$$

（1）原始数据标准化

由于 p 个特征参数的量纲各不相同，所以需要对样本数据进行标准化，使得每个变量标准化后的平均值为 0，标准差为 1。则标准化后可得到矩阵 X ：

$$X=\begin{bmatrix} x_{11} & x_{12} & \cdots & x_{1p} \\ x_{21} & x_{22} & \cdots & x_{2p} \\ \vdots & \vdots & \vdots & \vdots \\ x_{n1} & x_{n2} & \cdots & x_{np} \end{bmatrix}$$

其中：

$$x_{ij}=(z_{ij}-\bar{z}_j)\Big/s_j, i=1,2,...,n; j=1,2,...,p \tag{4-19}$$

$$\bar{z}_j=\frac{1}{n}\sum_{i=1}^{n}z_{ij} \tag{4-20}$$

$$s_j{}^2=\frac{1}{m-1}\sum_{i=1}^{m}(z_{ij}-\bar{z}_j)^2 \tag{4-21}$$

（2）求相关系数矩阵 R

$$R=\begin{bmatrix} r_{11} & r_{12} & \cdots & r_{1p} \\ r_{21} & r_{22} & \cdots & r_{2p} \\ \vdots & \vdots & \vdots & \vdots \\ r_{n1} & r_{n2} & \cdots & r_{np} \end{bmatrix}$$

其中，r_{ij} 的计算公式为：

$$r_{ij}=\frac{\sum_{k=1}^{n}((x_{ki}-\bar{x}_i)(x_{kj}-\bar{x}_j))}{\sqrt{\sum_{k=1}^{n}(x_{ki}-\bar{x}_i)^2\sum_{k=1}^{n}(x_{kj}-\bar{x}_j)^2}}, i=1,2,...,n; j=1,2,...,p \tag{4-22}$$

（3）计算矩阵 R 的特征值和特征向量

首先对 $|\lambda I-R|=0$ 特征方程求解，得到 p 个特征值并对它们进行降序排列，然后分别求出其对应的正交化特征向量 $e_i(i=1,2,...,p)$，其中 $\|e_{ij}\|=1$ ，e_{ij} 表示 e_i 的第 j 个分量。

（4）计算主成分方差贡献率及累积方差贡献率

各主成分在总方差中所占的比重称为方差贡献率，当累积贡献率达到一定百分比时，则认为这些主成分可以综合表示所有指标所要表达的信息，从而达到降维的目的。

第i个主成分Y_i的方差贡献率为：

$$\frac{\lambda_i}{\sum_{k=1}^{p}\lambda_k}(i=1,2,...,p) \tag{4-23}$$

累积方差贡献率为：

$$\frac{\sum_{k=1}^{i}\lambda_k}{\sum_{k=1}^{p}\lambda_k}(i=1,2,...,p) \tag{4-24}$$

一般取累积方差贡献率达到 85%以上的特征值$\lambda_1,\lambda_2,...,\lambda_m$所对应的主成分作为最终的特征参数指标。

2.k-means 聚类方法[6]

对特征参数进行PCA分析之后得到5个主成分，然后对所有的运动学片段进行聚类，每一类都是由具有相同交通特征的运动学片段构成的，分别选取每类片段库中最优备选片段，进而构建出具有代表性的工况曲线。

k-means 聚类的具体步骤如下：

（1）选择初始凝聚点和初始分类，比如取k个初始凝聚点

（2）对所有样品逐一计算它到k个凝聚点的距离（通常采用欧氏距离），根据距离的大小将n个样品分成k类，计算公式如下：

$$d_{xy}=\sqrt{\sum_{i=1}^{n}(x_i-y_i)^2} \tag{4-25}$$

其中，x_i为样本x的第i个变量的变量值，y_i为样本y的第i个变量的变量值。若某样品到它原来所在类的距离最近，则它仍在原类，否则将它移动到和它距离最近的那一类。

（3）计算 k 类中每一类数据的重心，若重心与初始凝聚点不重合则以重心为新的凝聚点并重复步骤 b 直到所有的样品都不能移动为止，或者说如果某一步所有的分类的重心与凝聚点重合，则计算过程终止。

3.片段选取方法[1]

聚类分析是将特征参数值相近、交通特征相似度高的运动学片段聚合到同一类中，而最终的工况曲线包含了每一类片段库中的运动学片段，所以这些片段应尽可能全面的反映每一类片段库的典型特征，这样构建出来的工况曲线才能够客观地反映车辆的实际行驶情况。本文提出了一种基于总体特征参数偏差最小的片段选取方法，具体步骤如下：

（1）假设 k-means 聚类之后将运动学片分为n个片段库，每一个运动学片段的特征参数个数为p，m_k是第 k 个片段库中运动学片段的个数。由于每个特征参数之间的数量级不一样，需要对特征参数进行标准化，具体公式如下：

$$z_{ij,k}=(x_{ij,k}-\min x_{j,k})/(\max x_{j,k}-\min x_{j,k}) \tag{4-26}$$

$$i=1,2,...,m_k;\ j=1,2,...,p;k=1,2,...,n$$

其中，$x_{ij,k}$是第k个片段库中第i个片段的第j个参数的数值，$\max x_{j,k}$和$\min x_{j,k}$分别是第k个片段库中所有片段的第j个参数的最大值和最小值。 $z_{ij,k}$是标准化之后的特征参数值。

（2）在上一步的基础上，计算第k个片段库中所有片段对应的特征参数之和，公式如下所示：

$$z_{i,k}=\sum_{j=1}^{p}z_{ij,k},i=1,2,...,m_k \tag{4-27}$$

（3）计算第个片段库中所有片段的每个参数的平均值，对其进行标准化之后再求和，公式如下所示：

$$Y_k=\sum_{j=1}^{p}(\overline{x}_{j,k}-\min x_{j,k})/(\max x_{j,k}-\min x_{j,k}) \tag{4-28}$$

其中$\overline{x}_{j,k}$是第k个片段库中所有片段第j个参数的平均值，Y_k是指第k个片段库中所含片段的每个特征参数的平均值经标准化之后相加的和。

（4）计算各个片段库在最终工况曲线中的时间占比

$$T_k=\frac{\sum_{i=1}^{m_k}T_{i,k}}{\sum_{k=1}^{n}\sum_{i=1}^{m_k}T_{i,k}}\times T_f \tag{4-29}$$

其中，T_k为第k类运动学片段在最终工况曲线中的时间占比，$T_{i,k}$是第k 类中第i个片段的持续时间，T_f是设定的最终循环工况的运行时间（题目要求为 1200-1300s，本文取 1200s）

（5）计算第k类运动学片段的特征参数之和$z_{ij,k}$与该类片段库中所有片段的平均特征参数之和Y_k的差值绝对值，按照绝对值的大小从小到大排列并依次选取为备选工况片段，直至选取的片段总运行时间大于等于 d 中计算的结果T_k。将从每一类片段库中选出的片段进行组合，构成最终的汽车行驶工况曲线。

4.3.4 求解结果

1.求解特征参数

根据上面的模型对文件 1 进行处理，可以得到 1103 个运动学片段的车辆行驶特征参数，将矩阵记为F， F 如表 4 所示。

表 4 文件 1 运动学片段的特征参数

	$T(s)$	$T_a(s)$	$T_d(s)$	$T_i(s)$	$S(m)$	$V_{\max}(km/h)$	…	$a_{sd}(m/s^2)$
1	69	26	19	8	131.8741	17.4	…	0.242519767
2	370	111	86	97	2729.836	56.5	…	0.227593626
3	120	48	35	24	699.4778	45.2	…	0.273211089
4	219	61	41	73	1677.325	55	…	0.270416178
5	171	51	36	34	1889.725	68.7	…	0.397249324
6	221	92	54	40	1833.522	65.5	…	0.274270078
7	435	186	104	53	6127.389	81.5	…	0.233791922
8	290	130	80	39	3427.973	65.9	…	0.225184347
9	196	80	44	46	1616.201	60.9	…	0.415417656
…	…	…	…	…	…	…	…	…
1103	487	143	106	14	7014.026	67.2		0.234858655

同理，经过计算可以得到文件 2 和文件 3 的运动学片段分别为 822 个和 786 个，如下表 5 和 6 所示。

表 5 文件 2 运动学片段的特征参数

	$T(s)$	$T_a(s)$	$T_d(s)$	$T_i(s)$	$S(m)$	$V_{\max}(km/h)$	…	$a_{sd}(m/s^2)$
1	150	17	16	111	134.3208	11.8	…	0.2851753
2	23	9	7	4	49.8636	13.2	…	0.35944182
3	230	60	66	58	1217.4147	36.7	…	0.28292501
4	1142	329	280	180	21110.2788	84.7	…	0.31608808
5	39	8	4	24	35.6532	12.2	…	0.21708385
6	140	47	68	1	10719.5722	61.8	…	0.81474307
7	46	20	12	7	302.9686	35.6	…	0.41379785
8	26	2	2	18	21.6188	2.2	…	0.21609183
9	403	132	116	59	6456.7085	71.9	…	0.44510935
…	…	…	…	…	…	…	…	…
822	29	6	8	7	72.9631	8.6		0.55295484

表 6 文件 3 运动学片段的特征参数

	$T(s)$	$T_a(s)$	$T_d(s)$	$T_i(s)$	$S(m)$	$V_{\max}(km/h)$	…	$a_{sd}(m/s^2)$
1	108	42	34	18	765.9982	56.5	…	0.32737671
2	191	44	44	81	636.7453	42	…	0.49459098
3	177	21	12	128	418.0459	40.1	…	0.37379936
4	104	28	24	43	334.9536	43	…	0.43890118
5	25	8	8	2	91.6509	17	…	0.4678413
6	73	21	20	17	951.0209	50.8	…	0.23824384
7	75	25	28	7	368.1055	30.1	…	0.62725551
8	112	47	41	2	897.1827	43	…	0.34957359
9	51	22	10	4	648.5848	37.4	…	0.43202894
…	…	…	…	…	…	…	…	…
786	732	242	179	47	14242.16	87.4		0.25156571

2.PCA 降维

因为是对所有文件中的运动学片段进行聚类，所以将文件 1，2，3 中所有的运动学片段合并到一起，一共有 2711 个有效的运动学片段，计算所有运动学片段的特征参数值，下表为文件中所有运动学片段的特征参数。由于选取的特征参数过多，各个特征参数之间存在着显性或者隐性的相关关系，如果采用所有的参数作为聚类的依据，往往还会带来错误的结果，所以需要采取一定的措施将原本重复的变量进行剔除并形成能够尽可能全面反映原始变量信息的新变量。通过对特征参数矩阵进行 *PCA* 分析，得到特征参数矩阵的主成分，降维之后得到的主成分能反映出原来的大部分特征参数。这种方法降低计算量的同时还增加了试验结果的正确性以及可靠性。

表 7 文件中所有运动学片段的特征参数

	$T(s)$	$T_a(s)$	$T_d(s)$	$T_i(s)$	$S(m)$	$V_{\max}(km/h)$	…	$a_{sd}(m/s^2)$
1	69	26	19	8	131.8741	17.4	…	0.24251977
2	370	111	86	97	2729.8356	56.5	…	0.22759363
3	120	48	35	24	699.4778	45.2	…	0.27321109
4	219	61	41	73	1677.3246	55	…	0.27041618
5	171	51	36	34	1889.7254	68.7	…	0.39724932
6	221	92	54	40	1833.5215	65.5	…	0.27427008
7	435	186	104	53	6127.3886	81.5	…	0.23379192
8	290	130	80	39	3427.9725	65.9	…	0.22518435
9	196	80	44	46	1616.2013	60.9	…	0.41541766
…	…	…	…	…	…	…	…	…
2711	732	242	179	47	14242.16	87.4		0.25156571

综上可知，文件 1，文件 2，文件 3 合并在一起共有 2711 条运动学片段，根据 4.3.3

节所描述的主成分分析步骤，对标准化后的特征值矩阵进行 PCA 主要成分分析，得到 19 个成分，分别将他们记为 M_1 ，M_2 ，...，M_{19}，计算其方差贡献率和累积方差贡献率等值，结果见下表 8 所示。

表 8 2711 条运动学片段的成分分析

成分	主成分方差值	方差贡献率%	累积方差贡献率%
M_1	7.361	38.743	38.743
M_2	2.707	14.246	52.989
M_3	2.559	13.469	66.459
M_4	1.601	8.424	74.883
M_5	1.054	5.550	80.432
M_6	0.999	5.256	85.688
M_7	0.810	4.261	89.950
M_8	0.544	2.864	92.813
M_9	0.446	2.346	95.159
M_{10}	0.303	1.592	96.752
M_{11}	0.213	1.119	97.871
M_{12}	0.161	0.847	98.718
M_{13}	0.093	0.489	99.208
M_{14}	0.073	0.383	99.591
M_{15}	0.054	0.283	99.874
M_{16}	0.016	0.083	99.958
M_{17}	0.008	0.042	100.000
M_{18}	-7.548E-17	-3.972E-16	100.000
M_{19}	-1.407E-16	-7.407E-16	100.000

相关系数矩阵的特征值代表了所对应的主成分所涵盖原始变量（即 19 个特征参数）的信息量的多少，特征值越大，表示该主成分能较大程度地取代原始变量，相反，如果特征值越小，尤其是小于 1 时，则表示该主成分没有原来的 1 个变量反映的信息多。贡献率是另一个衡量主成分信息涵盖能力的指标，它表明主成分对原始指标的反映程度，贡献率越高则表明主成分对原始指标的表征更全面准确，一般认为累积贡献率达到 85% 时能够较为真实而全面地反映原始指标的信息[8]。由上表 8 可以看出，前五个成分的累积贡献率已达到 80.432%，它们已经几乎可以代表所有原始变量反映的信息，且这五个

成分的特征值均大于 1，所以前五个成分作为主成分足以代表原始特征参数。此外，主成分载荷是指该类主成分中每个指标对应的线性组合系数，它代表了该主成分中与某个指标的相关程度以及包含该指标的信息量大小。某个指标在某个主成分上的载荷系数的绝对值越大，则说明该指标与这个主成分的相关系数越高，所包含的信息量也就越多。前五个主成分的载荷矩阵，即主成分系数矩阵，如表 9 所示。

表 9 主成分的成份矩阵

特征参数	1	2	3	4	5
T (运行时间)	0.834	-0.166	0.401	0.285	0.006
T_a(加速时间)	0.883	-0.170	0.206	0.225	0.026
T_d (减速时间)	0.854	-0.185	0.187	0.335	0.074
T_i(怠速时间)	-0.028	0.150	0.860	-0.103	0.164
T_e (匀速时间)	0.797	-0.247	0.241	0.379	-0.181
S(运行距离)	0.000	-0.019	-0.027	-0.015	-0.190
V_{max}(最大速度)	0.887	0.240	0.116	-0.247	0.049
V_m (平均速度)	0.931	0.061	-0.086	-0.158	0.013
V_{mr} (行驶速度)	0.874	0.193	0.154	-0.282	0.064
V_{sd} (速度标准偏差)	0.738	0.336	0.182	-0.428	0.120
V_{max_a} (最大加速度)	0.352	0.689	-0.024	0.374	0.225
a_a (加速度段平均加速度)	-0.324	0.671	-0.042	0.384	-0.106
a_{min_d}(最小减速度)	-0.543	-0.465	0.046	0.054	0.389
a_d(减速段平均减速度)	-0.012	-0.693	0.077	0.283	0.491
a_{sd} (加速度标准偏差)	-0.193	0.730	-0.088	0.506	0.131
P_i(怠速时间比)	-0.636	0.139	0.726	-0.051	0.028
P_a(加速时间比)	0.561	0.085	-0.583	-0.330	0.151
P_e (匀速时间比)	0.398	-0.374	-0.250	0.287	-0.538
P_d (减速时间比)	0.346	0.011	-0.704	0.182	0.384

由上表 9 可知，第一主成分主要反映了运行时间、加速时间、减速时间、匀速时间、最大速度、平均速度、行驶速度和速度标准偏差；第二主成分主要反映了最大加速度、加速度段平均加速度和加速度段标准偏差；第三主成分主要反映了怠速时间；第四主成分主要反映速度标准偏差和加速度标准偏差；第五主成分主要减速段平均减速度、最小减速度和减速时间比。

以上五个主成分涵盖了上文提出的 19 个主要特征，能够全面的表征运动学片段的

特征。规定将标准化后的样本数据矩阵与主成分成份矩阵相乘即可得到运动学片段的因子得分系数矩阵。本文将选取前五个主成分的因子得分系数矩阵作为研究 K-means 聚类的输入变量。

3.聚类的结果

用 4.3.3 节 K-means 聚类方法对得到的五个主成分得分进行聚类，根据事先设定初始凝聚点的个数 k 为 4，通过 Python 语言编程进行数据分析，将原来的 2711 个运动学片段划分成 4 类：第一类包括 1075 个运动学片段，第二类包括 1046 个运动学片段，第三类包括 389 个运动学片段，第四类包括 201 个运动学片段。

为了能够进一步分析聚类后每一类片段库中的运动学片段所代表的车辆行驶特征，根据上述的 19 个特征值计算公式进行相应地计算，得到各个类片段库中所有运动学片段及整体试验数据的综合特征值，详见下表 10。

表 10 各类片段库综合特征参数值

特征参数	第一类	第二类	第三类	第四类
总运行时间(s)	59765	105665	72305	144515
平均运行时间(s)	15989	37662	16114	54014
总路程(m)	193645.864	1716719	423198.2	27618507
总怠速时间(s)	17586	5558	34303	17573
总加速时间(s)	15989	37662	16114	54014
总减速时间(s)	15375	30278	12893	42048
总匀速时间(s)	10815	32167	8995	30880
平均加速时间(s)	14.87	187.37	41.42	51.64
最大速度(km/h)	43.2	261.4	93.6	105.8
平均速度(km/h)	7.55	43.64	13.81	26.00
行驶速度(km/h)	10.69	46.17	26.91	30.43
速度标准偏差(km/h)	6.00	18.08	15.02	15.14
平均怠速(km/h)	16.36	27.65	88.18	16.80
平均怠速时间比(%)	0.32	0.05	0.51	0.14
平均匀速时间比(%)	0.17	0.30	0.11	0.20
平均加速时间比(%)	0.25	0.36	0.21	0.38
平均减速时间比(%)	0.25	0.29	0.17	0.29
加速度标准偏差(m/s^2)	0.42	0.31	0.42	0.38

由表 10 可以看出，各个类中所有运动学片段的综合特征参数呈现一定的分布规律，根据每一类运动学片段的整体统计分布特点可以将所有运动学片段分为超低速、低速、中速、高速这四大类，对于这四类中所有片段的 19 种特征参数的平均值或总和进行比较，可以得到每一类运动学片段表现出的车辆行驶特征，总结如下：

（1）第一类运动学片段运行时间最短，运行里程最短。平均速度和行驶中的平均速度最小，行驶中平均速度仅为 10.69km/h。怠速时间相对较高，说明车辆在加减速频繁、交通堵塞情况较为严重的低速次干道上行驶。

（2）第二类运动学片段运行里程为 1717km，平均加速时间最大，总怠速时间比例较小，最大加速度和行驶中平均速度都是四类中的最大值。说明该路段非常畅通，不拥挤，车辆在该路段保持匀速运行的时间最长和匀速时间比最大，代表了在主干道上高速运行的模式。

（3）第三类运动学片段怠速时间比例最高，约为第一、第二和第四类的怠速时间的总和。平均速度只有 13.81km/h，运行中的平均时速也不过 26.91km/h，但比超低速平均速度高，平均怠速时间比占比 51%，这代表了车辆在红绿灯较频繁的路段行驶，交通堵塞情况不严重的低速干道上断断续续运行的模式。

（4）第四类运动学片段运行时间最长，运行里程为 27619km，总加、减速时间最长，平均加减速时间比最大，怠速时间比例较小，平均速度处于中等水平，说明该路段比较畅通，车辆在该路段行驶时经常加减速，这代表了在主干道上中速运行的模式。

4.最终的结果

根据文中要求，本文采用 1200s 作为构建工况的预定持续时间。聚类将所有的运动学片段划分为四类，各个类包含的运动学片段数分别为 1075，1046，389 和 201 个。根据聚类分析结果计算出每一类片段库中应选出的运动学片段时长，以及 4.3.3 节所述的基于总体特征参数偏差最小的片段选取方法，共选出 10 个运动学片段，总时长为 1200s，如下表 11 所示。

表 11 构成工况曲线

运动学片段序号	所属类号	运行时间	$\left\|z_{ij,k}-Y_k\right\|$
0	0	43	0.019343
1	0	136	0.044453
2	0	30	0.050800
3	2	162	0.001707
4	2	85	0.330195
5	3	60	0.002823
6	3	131	0.014244

7	3	96	0.020637
8	3	170	0.026531
9	1	287	6.557490

最终的汽车行驶工况曲线如下图 5 所示。

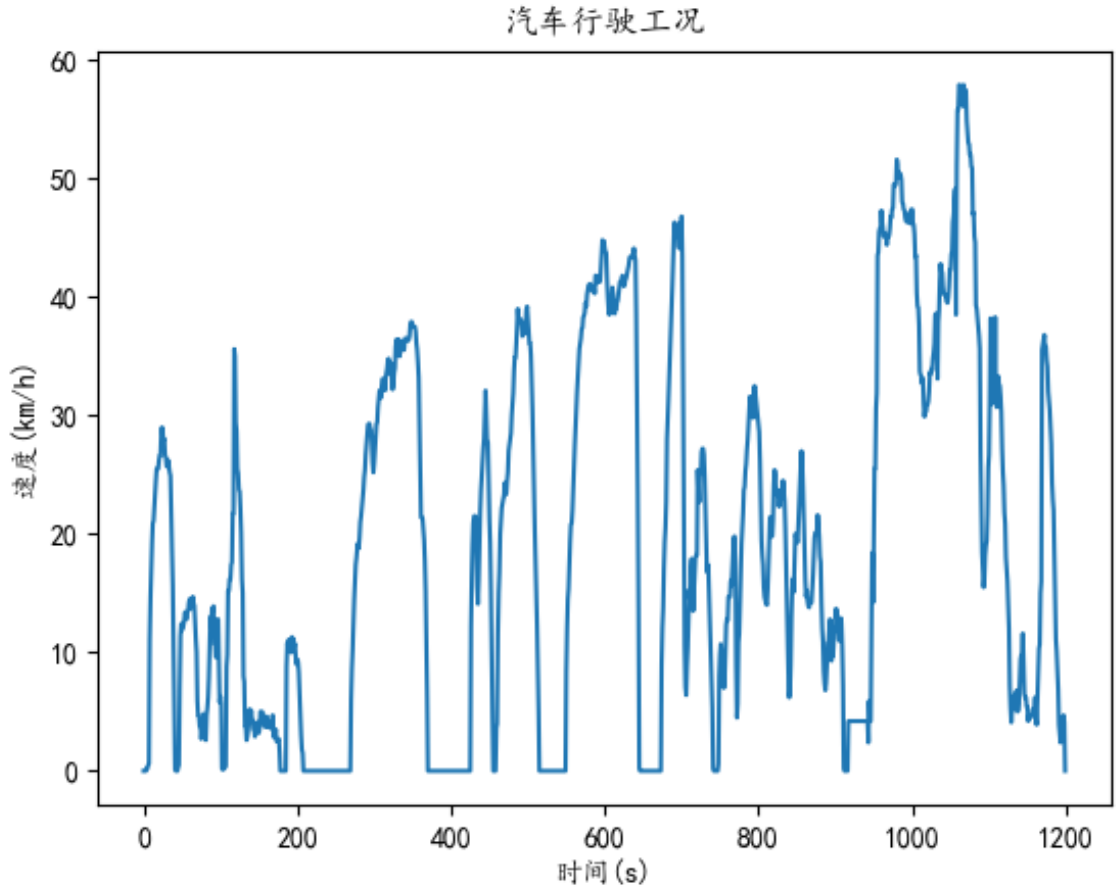

图 5 汽车行驶工况

4.3.5 模型的分析与检验

通常情况下，构建出的行驶工况曲线需要对该曲线进行验证，才能判断该曲线是否具有代表性。本文通过选取特征参数法进行有效性的判定，即选定最大速度、平均速度、行驶速度、速度标准偏差、平均加速度、平均减速度、加速度标准偏差、怠速时间比、加速时间比、匀速时间比、减速时间比等具有代表性的特征参数，将划分运动学片段后特征参数与构建行驶工况曲线的特征参数进行比较，根据其误差概率的近似程度来评判汽车构建行驶工况曲线的代表性优劣，如下表 12 所示，构建出的行驶工况曲线特征参数与所采集数据源的部分特征参数对比状况。

表 12 工况特征评价参数

	源数据特征	工况的特征	误差率
最大速度	37.171081	36.860000	0.008369
平均速度	18.24187	17.739288	0.027551
行驶速度	23.261045	22.674312	0.025224
速度标准偏差	11.71523	12.309383	0.050716
平均加速度	0.495923	0.493151	0.005588
平均减速度	-0.59241	-0.587878	0.007650
加速度标准偏差	0.396769	0.405681	0.022670
怠速时间比	0.259096	0.231116	0.108102
加速时间比	0.302607	0.311254	0.028575
匀速时间比	0.182927	0.208653	0.140634
减速时间比	0.25537	0.288978	0.131605

从表 12 中可以看出，最终的行驶工况曲线和整体实验数据的相应特征参数的误差大部分位于 10%以内。该结果表明构建出来的汽车行驶工况能够反映汽车真实的道路行驶特征，验证了其有效性。

六、参考文献

[1] 李洋. 基于聚类算法的汽车行驶工况研究[D].北京理工大学,2016.

[2] 吴岩.电动汽车行驶工况构建研究[J].汽车实用技术,2018(23):14-16.

[3] 高建平,任德轩,郁建国.基于全局 K-means 聚类算法的汽车行驶工况构建[J].河南理工大学学报(自然科学版),2019,38(01):112-118.

[4] Zhou T, Liu Y, Wang Y, et al. Model Study and Analysis Based on Student Performance Data under SPSS Statistics—Take a College from Hangzhou Normal University as an Example[J]. DEStech Transactions on Social Science, Education and Human Science, 2018 (ichae).

[5] 彭育辉,杨辉宝,李孟良,乔学齐.基于 K-均值聚类分析的城市道路汽车行驶工况构建方法研究[J].汽车技术,2017(11):13-18.

[6] 姜平. 城市混合道路行驶工况的构建研究[D].合肥工业大学,2011.

[7] Lin J, Niemeier D A. An exploratory analysis comparing a stochastic driving cycle to California's regulatory cycle[J]. Atmospheric Environment, 2002, 36(38): 5759-5770.

[8] Ergeneman M, Sorusbay C, Goktan A. Development of a driving cycle for the prediction of pollutant emissions and fuel consumption[J]. International Journal of Vehicle Design, 1997, 18(3): 391-399.

[9] Knez M, Muneer T, Jereb B, et al. The estimation of a driving cycle for Celje and a comparison to other European cities[J]. Sustainable cities and society, 2014, 11: 56-60.

七、附录

1.K-means 聚类

```
'''聚类分析'''
def JL_FenXi():
    from sklearn import preprocessing
    # 正则化
    min_max_scaler = preprocessing.MinMaxScaler()
    X_norm = min_max_scaler.fit_transform(JW_result)

    # kmeans 聚类
    # inertia 样本到最近的聚类中心的距离总和
    # 肘部法则
    import matplotlib.pyplot as plt
    from sklearn.cluster import KMeans
    distortions = []
    for i in range(1, 40):
        km = KMeans(n_clusters=i,
                    init='k-means++',
                    n_init=10,
                    max_iter=300,
                    random_state=0)
        km.fit(X_norm)
        distortions.append(km.inertia_)
        print('分成',i,'类，对应的距离总和（越小越好）：',km.inertia_)
    #画图
    plt.plot(range(1, 40), distortions, marker='o')
    plt.xlabel('Number of clusters')
    plt.ylabel('Distortion')
    plt.show()#inertia 样本到最近的聚类中心的距离总和。越小越好


    print('----------------------------------')
    #轮廓系数
    from sklearn import metrics
    import matplotlib.pyplot as plt
    from sklearn.cluster import KMeans
    scores = []
    for i in range(2, 100):
        km = KMeans(        n_clusters=i,
                            init='k-means++',
                            n_init=10,
                            max_iter=300,
```

```
                                random_state=0       )
        km.fit(X_norm)
        scores.append(metrics.silhouette_score(X_norm, km.labels_ ,
metric='euclidean'))
        print('分成', i, '类，轮廓系数（越大越好）：',
metrics.silhouette_score(X_norm, km.labels_ , metric='euclidean'))
    plt.plot(range(2,100), scores, marker='o')
    plt.xlabel('Number of clusters')
    plt.ylabel('silhouette_score')
    plt.show()  #越大越好。[-1,1]


    print('-----------------------------------')
    #Calinski-Harabaz Index。这种评估也同时考虑了族内族外的因素。类别内部数
据的协方差越小越好，类别之间的协方差越大越好
    from sklearn import metrics
    import matplotlib.pyplot as plt
    from sklearn.cluster import KMeans
    ch_scores = []
    for i in range(2, 100):
        km = KMeans(n_clusters=i,
                    init='k-means++',
                    n_init=10,
                    max_iter=300,
                    random_state=0)
        km.fit(X_norm)
        ch_scores.append(metrics.calinski_harabaz_score(X_norm, km.labels_))
        print('分成', i, '类，Calinski-Harabaz Index：',
metrics.calinski_harabaz_score(X_norm, km.labels_))
    plt.plot(range(2, 100), ch_scores, marker='o')
    plt.xlabel('Number of clusters')
    plt.ylabel('calinski_harabaz_score')
    plt.show()
```

2. 基于总体特征参数偏差最小的片段选取方法

```
# -*- coding:utf-8 -*-
import datetime
import numpy as np
import pandas as pd
pd.set_option('display.max_columns', 1000)#pd的输出最大多少列
pd.set_option('display.max_rows', 1000)#pd的输出最大多少行
pd.set_option('display.width', 1000)#pd的输出宽度
pd.set_option('display.max_colwidth', 1000)#pd的输出最大列宽
import matplotlib.pyplot as plt
```

```
from sklearn.decomposition import PCA
from sklearn.cluster import KMeans
from sklearn import metrics
from sklearn import preprocessing
df1=''
'''1.读入文件'''
def ReadFile():
    df1 =
pd.read_csv('HeBing3GeWenJianDeYunDongXuePianDuanTeZheng(BaoHanCuHao).csv',
encoding='utf-8')
    f = lambda x: datetime.datetime.strptime(x, "%Y-%m-%d %H:%M:%S")  # 把时
间字符串变为时间对象
    df1["起始时刻"] = df1["起始时刻"].apply(f)
    df1["结束时刻"] = df1["结束时刻"].apply(f)
    return df1

'''2.标准化'''
def BiaoZhunHua(data):
    global df1
    data_standardized = preprocessing.scale(data)  # 标准化     #<class
'numpy.ndarray'>
    data_standardized=pd.DataFrame(data_standardized,columns=['S 运行时间
','S 加速时间','S 减速时间','S 怠速时间','S 匀速时间','S 运行距离','S 最大速度
','S 平均速度','S 行驶速度','S 速度标准偏差','S 最大加速度','S 加速度段平均加
速度','S 最小减速度','S 减速段平均减速度','S 加速度标准偏差','S 怠速时间比','S
加速时间比','S 匀速时间比','S 减速时间比'])
    data_standardized = pd.concat((df1,data_standardized), axis=1)  #横着串
联起来
    return data_standardized


'''3.求（所有簇）每个运动学片段的特征之和，放在 DF 后面。'''
def EveryMotionSeries_FeatureSum(temp):
    # print(temp)
    FeatureSum=temp.sum(axis=1) #求每行的和。返回 series
    return FeatureSum

'''4.分别对每个簇内部，求每个特征的均值，无量纲化后，加起来'''
def EveryCu_Feature_AVG_Standard_Sum(temp):
    result = list(range(len(temp['簇号'].value_counts())))#每个簇最终的结果
[]。先占位。
    for i in range(len(temp['簇号'].value_counts())):
        EveryFeatureAVG = temp[temp['簇号']==i].mean()#返回 Series。每列的均
值
```

```
        data_standardized = preprocessing.scale(EveryFeatureAVG)#无量纲化，即标准化
        result[i]=data_standardized.sum()
    return result

'''5.计算每个簇 在曲线中 所占时间的比例，乘以 1200s'''
def EveryCu_Time(temp):
    result_Time = list(range(len(temp['簇号'].value_counts())))  # 每个簇在曲线中占的时间[]。先占位。
    for i in range(len(temp['簇号'].value_counts())):
        result_Time[i]=round(1200*(temp[temp['簇号']==i]['运行时间'].sum()/temp['运行时间'].sum()),0)#时间就不要小数了。
    return result_Time

'''6.每个运动学片段的特征之和 - 它自己所在簇的特征计算值          再求绝对值，依然放在 DF 后面。然后从小到大排序作为备选工况， 直到时间大于等于 5 中计算的该簇在曲线中的时长'''
#每个运动学片段的特征之和 - 它自己所在簇的特征计算值          再求绝对值,放在 DF 后面
def JueDuiZhi(temp,cuFeature):
    # print(temp)
    # print(cuFeature)
    result_JDZ=pd.Series()#装绝对值
    for i in range(len(temp)):
        result_JDZ.loc[i]=abs(temp['每个 YDX 片段的特征和'][i]-cuFeature[temp['簇号'][i]])
    return result_JDZ
#曲线由哪些运动学片段组成，返回(簇号,起始时刻,结束时刻,时间,片段特征减去簇特征的绝对值)
def Choose_YDXSeries(temp,EveryCu_Time):
    # print(temp)
    # print(EveryCu_Time) #每个簇在曲线中的时间
    QuXianZuChengIndex = [] #[(簇号,起始时刻,结束时刻,时间,片段特征减去簇特征的绝对值),(),()...]
    # for i in range(len(EveryCu_Time)):#遍历每个簇，挑它们的课代表。结束条件是大于等于当前簇要求的时间。按照 0、2、3、1 的顺序
    for i in (0,2,3,1):
        times=0
        kuochongTimes=100#弹性的 100 秒
        current_Cu=temp[temp['簇号']==i].sort_values(by='片段特征减去簇特征的绝对值',ascending=True)#当前簇
        for j, value in current_Cu.iterrows():
            if times>=EveryCu_Time[i]:  #时间已经够了
                break
```

```
            # if (times+current_Cu['运行时间'][j]-EveryCu_Time[i])>25: continue #不让这个簇超过太多时间
            if(times+current_Cu['运行时间'][j]-EveryCu_Time[i])>kuochongTimes: continue #当前加入的 片段 超时时 必须要在弹性 100 秒内
            if(times+current_Cu['运行时间'][j]-EveryCu_Time[i])<=kuochongTimes and (times+current_Cu['运行时间'][j]-EveryCu_Time[i])>0: kuochongTimes-=(times+current_Cu['运行时间'][j]-EveryCu_Time[i])
            times+=current_Cu['运行时间'][j]
            QuXianZuChengIndex.append([current_Cu['簇号'][j],current_Cu['起始时刻'][j],current_Cu['结束时刻'][j],current_Cu['运行时间'][j],current_Cu['片段特征减去簇特征的绝对值'][j]])
    QuXianZuChengIndex=pd.DataFrame(QuXianZuChengIndex,columns=['簇号','起始时刻','结束时刻','运行时间','片段特征减去簇特征的绝对值'])
    return QuXianZuChengIndex

'''7.画速度曲线图'''
def DrawVT(QuXianZuCheng_index):
    # print(QuXianZuCheng_index)#簇号  起始时刻  结束时刻  运行时间  片段特征减去簇特征的绝对值
    #读 预处理最后一步 三个文件的合集 data123_(DropGPSV10)_474048.csv
    data123 = pd.read_csv('data123_(DropGPSV10)_474048.csv', encoding='utf-8')
    f = lambda x: datetime.datetime.strptime(x, "%Y-%m-%d %H:%M:%S")  # 把时间字符串变为时间对象
    data123["时间"] = data123["时间"].apply(f)

    tempDf=pd.DataFrame()#把曲线 里的运动学片段 对应时间的数据 都保存起来。
    for i, value in QuXianZuCheng_index.iterrows():#遍历曲线 里的运动学片段

        startTime=value['起始时刻']
        endTime=value['结束时刻']
        tempDf = pd.concat((tempDf, data123[(data123['时间'] >= startTime) & (data123['时间'] <= endTime)]), axis=0)  # 竖着串联起来
        # y.extend(data123[(data123['时间']>=startTime)&(data123['时间']<=endTime)]['GPS 车速'])
    # print(tempDf)
    # tempDf=tempDf.sort_values(by='时间')
    #图
    plt.plot(list(tempDf['GPS 车速']))
    plt.xlabel('时间(s)')
    plt.ylabel('速度(km/h)')
    plt.title('汽车行驶工况')
```

```
    plt.savefig('汽车行驶工况.png')
    plt.show()


if __name__ == '__main__':
    '''1.读入文件'''
    df1 = ReadFile()
    # print(df1)

    '''2.所有特征无量纲化， 即标准化'''
    df1_BZH = BiaoZhunHua(df1.loc[:,['运行时间','加速时间','减速时间','怠速时间','匀速时间','运行距离','最大速度','平均速度','行驶速度','速度标准偏差','最大加速度','加速度段平均加速度','最小减速度','减速段平均减速度','加速度标准偏差','怠速时间比','加速时间比','匀速时间比','减速时间比']])
    '''3.求（所有簇）每个运动学片段的特征之和，放在DF后面。'''
    df1_BZH['每个YDX片段的特征和']=EveryMotionSeries_FeatureSum(df1_BZH.loc[:,['S运行时间','S加速时间','S减速时间','S怠速时间','S匀速时间','S运行距离','S最大速度','S平均速度','S行驶速度','S速度标准偏差','S最大加速度','S加速度段平均加速度','S最小减速度','S减速段平均减速度','S加速度标准偏差','S怠速时间比','S加速时间比','S匀速时间比','S减速时间比']])
    # print(df1_BZH)
    '''4.分别对每个簇内部，求每个特征的均值，无量纲化后，加起来'''
    EveryCu_Feature_Sum = EveryCu_Feature_AVG_Standard_Sum(df1_BZH.loc[:,['S运行时间','S加速时间','S减速时间','S怠速时间','S匀速时间','S运行距离','S最大速度','S平均速度','S行驶速度','S速度标准偏差','S最大加速度','S加速度段平均加速度','S最小减速度','S减速段平均减速度','S加速度标准偏差','S怠速时间比','S加速时间比','S匀速时间比','S减速时间比','簇号']])
    # print(EveryCu_Feature_Sum)    #顺序与簇号 一样0、1、2、3。每个簇的 特征求和
    '''5.计算每个簇 在曲线中 所占时间的比例，乘以1200s'''
    EveryCuTime = EveryCu_Time(df1_BZH.loc[:,['簇号','运行时间']])
    # print(EveryCuTime)  #顺序与簇号 一样0、1、2、3。每个簇 在曲线中所占时间长度
    '''6.每个运动学片段的特征之和 - 它自己所在簇的特征计算值          再求绝对值，依然放在DF后面。然后从小到大排序作为备选工况， 直到时间大于等于5中计算的    该簇在曲线中的时长'''
    #计算 每个运动学片段的特征之和 - 它自己所在簇的特征计算值  的绝对值
    df1_BZH['片段特征减去簇特征的绝对值'] = JueDuiZhi(df1_BZH.loc[:,['簇号','每个YDX片段的特征和']],EveryCu_Feature_Sum)
    # print(df1_BZH)
    #曲线由哪些运动学片段组成，返回[(簇号,起始时刻,结束时刻,时间,片段特征减去簇特征的绝对值),(),()...]
    QuXianZuCheng_index = Choose_YDXSeries(df1_BZH.loc[:,['簇号','起始时刻','结束时刻','运行时间','片段特征减去簇特征的绝对值']],EveryCuTime)
```

```
print(EveryCuTime)
print(QuXianZuCheng_index)
print(QuXianZuCheng_index['运行时间'].sum())
'''7.画速度曲线图'''
DrawVT(QuXianZuCheng_index)
```