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	湖南大学
参赛队号	19105320034
队员姓名	1.张慧杰
	2.严露
	3.郭虎

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

题 目 汽车行驶工况构建

摘 要：

本文旨在对给定的汽车行驶数据构建汽车行驶工况。基于数据预处理和提取的运动学片段，选取最具代表性的运动学片段合成汽车行驶工况图。

针对问题一，首先以 NA 补齐给定数据中的缺失记录，然后对异常数据有选择性地进行缺失标记、删除和保留。计算各个连续缺失记录块的长度，对其从大到小进行删除，使全局数据缺失率不超过 20%。接着采用 PMM 方法（预测均值匹配法）加滑动窗口对数据进行选择性（滑动窗口内数据缺失率不超过 20%）的填补。然后采用 T4253H 非线性滤波器对数据中的白噪声和抖动做平滑处理。得出：经过上述数据预处理后，文件一、二和三保留的记录数分别是 178735，143970 和 159685。

针对问题二，“当前怠速状态开始至下一怠速状态开始”是提取运动学片段的首要规则，但为了保证提取的运动学片段包含足够的信息，本文进一步提出 4 个规则。因此，结合问题一的数据预处理结果和查阅资料，本文确定运动学片段提取的 5 个规则如下所示：（1）起止于两个相邻怠速状态开始的行程；（2）片段时间长度不少于 20s；（3）加速度在 0.54 到 14.4（$Km/(h\cdot s)$）范围内；（4）减速度在-14.4 到-0.54（$Km/(h\cdot s)$）范围内；（5）行驶距离不少于 10 米。对三个文件满足规则（1）的片段进行遍历，按照规则（2）-（5），得出：文件一、二和三满足规则的运动学片段数量分别为 895，641 和 698。三个文件提取的运动学片段总数量是 2234，其中时间最短的运动学片段是 21 秒，时间最长的运动学片段是 3179 秒。

针对问题三，首先对选取的运动特征评估指标进行主成分降维，选取特征值最大的前 4 个主成分（累计贡献率为 88.636%）。基于选取的 4 个主成分，采用高斯混合聚类和 K-均值聚类对问题二选取的运动学片段聚类。利用 Calinski-Harabasz(CH)指标对聚类性能进行评价，得到高斯混合聚类和 K-均值聚类最佳的聚类数量分别是 3 和 2。设定构建的工况图总时间为 1200s，以每个簇（类）中片段持续时间的总和占所有簇中片段持续时间的总和的比例作为该簇运动学片段在工况图中时间的占比。然后在每个簇中，依据其中的运动学片段与簇中心的距离，从小到大选取代表性片段直到组合的片段时长满足要求，最后把所有簇的代表性片段合成工况图。分别计算所采集的数据源与本文基于滑动窗口 PMM 填补的数据构建的汽车行驶工况的各指标值，结果如下表所示。

基于滑动窗口 PMM 填补的数据构建的工况	平均速度	平均行驶速度	怠速时间比	平均加速度	平均减速度	加速时间比	减速时间比	速度标准差	加速度标准差
采集数据源	29.021	37.027	0.194	1.443	1.781	0.279	0.228	27.037	1.275
混合高斯聚类	17.286	24.185	0.270	1.805	2.055	0.294	0.246	15.253	2.974
K-means 聚类	24.459	34.790	0.274	2.204	2.353	0.260	0.237	22.443	5.537

为了评价构建的汽车行驶工况是否合理，本文提出另外 4 种不同的数据预处理方法，对它们处理后的数据结合两种聚类方法，重复上述操作得到 8 个工况图。基于构建的汽车工况和采集的数据源之间 9 个运动特征评估指标的相对误差值，通过熵权赋值法对各指标赋权，建立基于综合评价的汽车工况选择体系。基于上述综合评价体系对得到的 10 个汽车工况进行评价，得到：上表中基于滑动窗口 PMM 填补的 K-means 聚类构建的汽车工况最具代表性，综合评价得分为 0.104，即综合的相对误差最小。

关键词：PMM，高斯混合聚类，K-均值聚类，工况验证，T4253H

第一章 问题重述

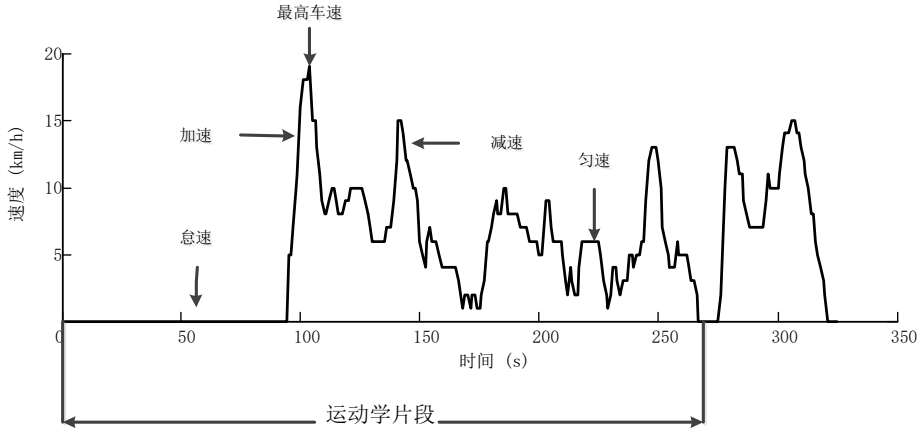
汽车行驶工况是描述汽车行驶的速度-时间曲线图，体现了汽车道路行驶的运动学特征，既是汽车能耗测试方法的限值标准，也是汽车各项性能指标标定优化时的主要基础。随着我国经济的发展，欧洲的 NEDE 工况标准和世界轻型车测试循环 WLTC 标准渐渐不能适应我国的国情。为了更好地理解汽车行驶工况曲线的重要性，本文基于给定的某型号汽车行驶的数据信息构建一条反映其行驶特征的工况曲线，该曲线所体现的汽车运动特征能代表所采集数据源的相应特征，两者间误差越小，说明构建的汽车行驶工况代表性越好。要解决的问题如下

（1） 题目给出的数据是汽车上的数据采集设备直接记录的原始数据，其中包含了一些不良数据，要求设计一些合理的方法对数据进行预处理。不良数据包括的主要类型如下：

- 由于高层建筑覆盖或过隧道等，GPS 信号丢失，造成所提供数据中的时间不连续；
- 汽车加、减速度异常的数据（普通轿车一般情况下：0 至 100km/h 的加速时间大于 7 秒，紧急刹车最大减速度在 7.5~8 m/s2）；
- 长期停车（如停车不熄火等候人、停车熄火了但采集设备仍在运行等）所采集的异常数据。
- 长时间堵车、断断续续低速行驶情况（最高车速小于 10km/h）,通常可按怠速情况处理。
- 一般认为怠速时间超过 180 秒为异常情况，怠速最长时间可按 180 秒处理。

（2） 提取运动学片段

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，如下图所示：



设计合理的方法，将经过（1）处理后的数据划分为多个运动学片段，并求出各数据文件最终得到的运动学片段数量。

（3） 根据（2）处理后的数据，构建一条能体现参与数据采集汽车行驶特征的汽车行驶工况曲线（1200-1300 秒），该曲线的汽车运动特征能代表所采集数据源（经

过（1）处理后的数据）的相应特征，两者间的误差越小，说明所构建的汽车行驶工况的代表性越好。要求如下：

- 科学、有效的构建方法；
- 合理的汽车运动特征评估体系（至少包含但不限于以下指标：平均速度（km/h）、平均行驶速度（km/h）、平均加速度（m/ s^2）、平均减速度（m/ s^2）、怠速时间比（%）、加速时间比（%）、减速时间比（%）、速度标准差（km/h）、加速度标准差（m/ s^2）等）；
- 按照构建好的汽车行驶工况及汽车运动特征评估体系，分别计算出汽车行驶工况与该城市所采集数据源（经（1）（2）处理后的数据）的各指标（运动特征）值，并说明构建的汽车行驶工况的合理性。

第二章 模型假设

1. 由于“停车不熄火等人”和“怠速”都具有速度为 0，转速大于 0 的特征，为简化处理，把两种情况统一描述为速度等于 0，转速大于 0。
2. 当缺失的速度记录跨天时，插值填补缺失值会造成一定误差，假设不存在该误差。
3. 假设题目给出的三个数据的记录自同一辆汽车的不同的时间段的记录。

第三章 符号说明

符号	解释
L	运行距离/m
V_m	平均速度/ $km \cdot h^{-1}$
$V_{\max}$	最大速度/ $km \cdot h^{-1}$
T	运行时间/ s
T_i	怠速时间/ s
T_a	加速时间/ s
T_d	减速时间/ s
T_c	匀速时间/ s
$a_{\max}$	最大加速度/ $Km/(h \cdot s)$
A_a	加速段的平均加速度/ $Km/(h \cdot s)$
A_d	减速段的平均减速度/ $Km/(h \cdot s)$
D_{ij}	片段距离簇中心的大小/ i 表示簇， j 表示簇中的片段
A_i	簇序列
T_{slice}	构建的工况图的时间/ s
T_{all-i}	每个簇中时间片段得和/ s

第四章 问题一

4.1 问题分析

汽车的行驶数据记录反映了汽车的运动过程，行驶中的各个参数反映了汽车当下时刻的状态。将表一文件中的数据中的经纬度，在高德地图上显示出来，如图 4.1 所示，从图可以看出从，采集该数据的车经过桥梁、隧道，同时位于市区，建筑物较密集，这些因素都会导致 GPS 信号丢失，造成数据中没有该段数据。同时由于仪器本身的原因，在一些特定情况下，仪器采集的加速度、减速度不能很好的反映实际车辆的加速度和减速度，造成

数据的异常。由于处于市区，交通较为拥堵，车辆长时间堵车、断断续续低速行驶的情况较多，长期停车等采集到较为异常的数据，同时采集数据的终端常常存在一定的零点漂移，在定位成功的情况下也会存在一定的车速偏差。本文对这些异常的数据，建立了一系列模型对其识别与剔除。

图 4.1 文件一汽车 12 月 18 号的行程图

针对高层建筑覆盖或过隧道灯导致的 GPS 信号丢失，造成所提供数据中的时间不连续现象，本文建立了 PMM 模型，而对采集设备存在的零点漂移现象，本文建立了 T4253H 模型。针对长时间堵车、怠速和长期停车所采集的异常的数据，本文建立了初步预处理模型。

4.2 模型建立

4.2.1 初步预处理模型建立

汽车行驶数据的不良信息主要分为五种，首先是要对异常值进行处理，给出数据的异常值有以下几点：

（1） GPS 信号丢失导致时间不连续：

以天为单位，一天记录的开始时间到结束时间之间若时间不连续，则暂时判定为缺失，以 NA 填充当数据缺失处理。

（2） 汽车加，减速异常:

为数据处理方便，把题目给定的条件换算成加速度（正/负），并统一单位为 Km/(h·s), 因此记录中正的加速度不能超过 14.286 Km/(h·s)（100/7），负的最大加速度的绝对值不能超过 28.8 Km/(h·s)（8×3.6）。这里只考虑采样数据前后连续两秒速度变化造成的加速度，若某一记录前一秒数据缺失，则不进行加，减速异常的判断。加，减速异常的判断数学表达式如公式（1）所示：

$$\frac{\Delta v}{\Delta t} - 14.268 > 0 \quad or \quad \frac{\Delta v}{\Delta t} + 28.8 < 0 \tag{1}$$

此种情况下的异常值为采集设备记录错误导致的，删去原记录并以 NA 填充当数据缺失处理。

（3） GPS 速度超出限速范围：

根据我国道路交通法规定，高速路的最大时速不超过 Vmax=120km/h， 给出的 GPS 速度数据中存在超过 120km/h 的情况，将此视为采集设备记录错误。删去

原记录并以 NA 填充当数据缺失处理。

（4） 停车并熄火：

停车表示速度为 0，熄火表示发动机不运转，即发动机转速 r 为 0。数学表达式如公式（2）所示：

$$(v-0)+(r-0)=0,\text{即}\ v+r=0 \tag{2}$$

对停车熄火但是采集器仍在进行的异常数据直接删除。

（5） 怠速：

怠速的定义为 v=0 且转速 r>0。由假设 1，“停车不熄火等人”和怠速相同，最高车速<10Km/h 的情况也按照怠速处理，因此可以把上述三种情况统一表示为公式（3）所示：

$$(v-10)r<0 \tag{3}$$

汽车怠速时长过长，不是正常的行驶数据，需要进行处理，因此对怠速时长超过 180s 的情况按照异常处理，从怠速开始，保留前 180s 的记录，之后的怠速记录删除。

步骤（1）-（5）处理完毕后，可把留下的记录按时间先后顺序连接起来，为第二步的缺失数据填补做准备。

4.2.2 PMM 模型建立

在对数据进行初步预处理模型处理后，便可以对数据不连续和上一个步骤被删除的数据进行填补，数据的填补常用的方法有以下几种[1]。

表 4.1 常用的数据填补方法

缺失数据处理方法	信息利用度	难度	适用范围	稳定性	偏差
预测均值匹配（PMM）	充分利用信息，考虑了缺失值具有不确定性	大	广	稳定	低
成对删除	信息利用度低，且删除了很多有用信息	小	缺失率<5%	敏感	高
均值插补	局限于回答信息	小	局限	敏感	高
回归插补	局限于回答信息	小	局限	敏感	低估方差、抽样误差
随机回归插补	局限于回答信息	小	广	稳定	方差估计较好，抽样误差不易控制
极大似然估计多重插补	充分利用信息，考虑了缺失值具有不确定性	大	广	稳定	误差不易控制
期望值最大化法	充分利用信息，考虑了缺失值具有不确定性	小	广	稳定	低

徐韬[1]（2018 年）通过实证发现，相比回归填补法和期望值最大化法（EM 法），PMM 方法在填补缺失率较大的数据时表现最好。因此本文采用 PMM 方法（预测均值匹配法）

对数据进行填补。结合文献[2]：多重填补方法 MI（包括 PMM 方法）在数据缺失率达到 20% 时，填补效果较其他方法最优，以及综合考虑，本文先对缺失的数据进行处理（剔除），使得数据的全局缺失率和滑动窗宽缺失率均不超过 20%，再使用 PMM 方法填补缺失数据。本文设计的缺失数据剔除方法如下：

（1）基于第一步处理后的数据，以天为单位，从第一个记录时刻开始，到最后一个记录结束。对数据连续缺失的情况进行记录，用 N 表示连续缺失时间的长度，比如 N= 5 表示存在一个连续 5 秒的记录缺失，然后对 N 的集合 $NN=\{N_1,N_2,\cdots\}$ 进行排序，从最小的 N 开始保留缺失数据的片段，直到留下的数据缺失值个数 x 满足全局缺失率不超过 20%，数学表达式如公式（4）所示：

$$\frac{x}{x+X}\le 20\% \tag{4}$$

其中，X 为不缺失的记录个数。

（2） 给定滑动窗宽为 30，对缺失值进行遍历，计算经过 1 处理后的序列的滑动窗宽缺失率，如不等式（5）左边所示，当公式（5）成立时，对该缺失值进行 PMM 填补，否则剔除。

$$\frac{\sum_{j=i-15}^{i+15} I\left(v_j\text{为空缺}\right)}{30}\le 20\% \tag{5}$$

PMM 方法的模型如下：

对一个含有缺失值的定量序列 X_{NA} ，基于拟合的回归模型，由后验预测参数模拟后的新的回归模型对其进行填补，回归模型如公式（6）所示：

$$X_{NA}=\alpha_0+\alpha_1X_1+\alpha_2X_2+\ldots+\alpha_kX_q \tag{6}$$

基于公式（6），利用观测到的记录估计 X_{NA} 与自变量 $X_1,X_2,\ldots,X_q$ 之间的回归参数 $\hat{\alpha}=\left(\hat{\alpha}_0,\hat{\alpha}_1,\hat{\alpha}_2,\ldots,\hat{\alpha}_q\right)$ ，协方差矩阵为 $\hat{\sigma}_{\mathrm{i}}^2V_{\mathrm{i}}$ 。

PMM 填补方法主要分为 3 步[3]：

（1）基于回归参数 $\hat{\alpha}$ 的后验分布对其进行模拟，得到新的回归参数 $\alpha_n=\left(\alpha_{\mathrm{n}0},\alpha_{\mathrm{nl}},\ldots,\alpha_{\mathrm{nq}}\right)$ 和新的方差 σ_{ni}^2 。

$$\sigma_{ni}^2=\hat{\sigma}_i^2\left(n_i-q-1\right)/g \tag{7}$$

其中，n_i 表示 X_{NA} 指标中非空缺的数量，q 为自变量的个数，g 是服从于 $\chi^2_{n_i-q-1}$ 分布的一个随机变量。

（2） 假设 X_{NA_i} 为序列中的第 i 个缺失记录，用新的回归模型对其进行预测，数学表达式如公式（8）所示：

$$\hat{X}_{NA\_\mathrm{i}}=\alpha_{n0}+\alpha_{n1}X_1+\alpha_{n2}X_2+\ldots+\alpha_{nq}X_q \tag{8}$$

（3） 以与 $\hat{X}_{NA\_i}$ 最接近的 X_i 的响应值替代 X_{NA_i}。

4.2.3 T4253H 模型建立

由于汽车数据采集器自身存在的一些缺陷和汽车的一些意外原因，数据的采集和真实值之间存在一定的误差，包括白噪声在内的数据抖动情况，有必要对数据进行滤波平滑操作。T4253H 是一种非线性滤波器，1980 年由 Velleman, P. F. 首次提出，低通转移性优良，吉布斯反弹低，对非高斯干扰有良好的抵制性[4]。通过归纳总结相关文献和 SPSS 的核心系统用户指南可知[5]，SPSS- T4253H 平滑首先对原始数据做 4 次平滑处理，平滑窗宽依次为 4，2，5，3。然后把 4 次平滑后得到的序列和原序列做比较，得到残差序列，接着对残差序列重复上述 4 次平滑过程得到二次残差序列，从原始序列的平滑序列中减去二次残差序列即得到最后的 T4253H 滤波结果。对具体步骤如下[4][5]：

（1）对原始数据 $\{P_j\}$ 进行窗宽为 4 的平滑，为避免异常值的干扰，这里平滑方式选择中位数，数学表达式如公式（9）所示：

$$\begin{cases} Q_{j+1/2}=median\left(P_{j-1},P_j,P_{j+1},P_{j+2}\right) \quad j=2,3,\ldots,n-2 \\ Q_{0.5}=P_1,Q_{1.5}=\dfrac{1}{2}\left(P_1+P_2\right),Q_{n-1/2}=\dfrac{1}{2}\left(P_{n-1}+P_n\right),Q_{n+1/2}=P_n \end{cases} \tag{9}$$

（2）对（1）的结果序列 $\{Q_{j+1/2}\}$ 进行窗宽为 2 的简单移动平均处理，数学表达式如公式（10）所示。

$$\begin{cases} Q_j^{(1)}=median\left(Q_{j-1/2},Q_{j+1/2}\right) \quad j=2,3,\ldots,n-1 \\ Q_1^{(1)}=Q_{0.5}, \quad Q_n^{(1)}=Q_{n+1/2} \end{cases} \tag{10}$$

（3）对（2）的结果序列 $\{Q_j^{(1)}\}$ 进行窗宽为 5 的平滑，平滑方式为取中位数，数学表达式如公式（11）所示。

$$\begin{cases} Q_j^{(2)}=median\left(Q_{j-2}^{(1)},Q_{j-1}^{(1)},Q_j^{(1)},Q_{j+1}^{(1)},Q_{j+2}^{(1)}\right) \quad j=3,4,\ldots,n-2 \\ Q_1^{(2)}=Q_1^{(1)},Q_2^{(2)}=median\left(Q_1^{(1)},Q_2^{(1)},Q_3^{(1)}\right) \\ Q_{n-1}^{(2)}=median\left(Q_{n-2}^{(1)},Q_{n-1}^{(1)},Q_n^{(1)}\right),Q_n^{(2)}=Q_n^{(1)} \end{cases} \tag{11}$$

（4）对（3）的结果序列 $\{Q_j^{(2)}\}$ 进行窗宽为 3 的平滑，平滑方式为取中位数，数学表达式如公式（12）所示。

$$\begin{cases} Q_j^{(3)}=median\left(Q_{j-1}^{(2)},Q_j^{(2)},Q_{j+1}^{(2)}\right) \quad j=2,3,\ldots,n-1 \\ Q_1^{(3)}=median\left(3Q_2^{(3)}-2Q_3^{(3)},Q_1^{(2)},Q_2^{(3)}\right) \\ Q_n^{(3)}=median\left(3Q_{n-1}^{(3)}-2Q_{n-2}^{(3)},Q_n^{(2)},Q_{n-1}^{(3)}\right) \end{cases} \tag{12}$$

对（4）的结果序列 $\{Q_j^{(3)}\}$ 进行窗宽为 3 的加权移动平均处理，权重依次为 1/4,1/2,1/4（也称作 Hanning 加权平均），得到平滑序列 $Q_j^{(4)}$，数学表达式如公式（13）所示。

$$\begin{cases} Q_j^{(4)}=\dfrac{1}{4}Q_{j-1}^{(3)}+\dfrac{1}{2}Q_j^{(3)}+\dfrac{1}{4}Q_{j+1}^{(3)} \quad j=2,3,\ldots,n-1 \\ Q_1^{(4)}=Q_1^{(3)},Q_n^{(4)}=Q_n^{(3)} \end{cases} \tag{13}$$

从原序列中减去公式（13）得到的$\{Q_j^{(4)}\}$序列，得到初始残差序列$\{D_j^{(1)}\}$，对初始残差序列$\{D_j^{(1)}\}$再进行公示（9）-（13）的操作，得到二次残差序列$\{D_j^{(2)}\}$。

$$D_j^{(1)} = P_j - Q_j^{(4)} \quad j = 1,2,\ldots,n \tag{14}$$

从平滑序列$Q_j^{(4)}$中减去二次残差序列$\{D_j^{(2)}\}$，得到最后的 T4253H 平滑序列$\{Y_j\}$。

$$Y_j = Q_j^{(4)} - D_j^{(2)} \quad j = 1,2,\ldots,n \tag{15}$$

4.3 模型求解与结果分析

4.3.1 预处理模型的求解

对于文件一、文件二和文件三，进行处理后，得到表 4.2 的不良数据统计结果。

表 4.2 不良数据统计结果

	文件 1	文件 2	文件 3
缺失值数目	270139	199690	221898
速度异常数目	0	0	298
加速度异常数目	1929	1877	1609
长期停车、怠速异常	7192	4098	5629

从表中可以看出，表一与表二中速度异常个数为零，说明在采集表一与表二的车辆最大速度低于 120km/h。同时缺失值的个数是很多的，已经超过所给数据本身大小。这可能是车辆处于市区，且从图 4.1 可以看出周围存在许多河道，因此车辆会经常出入隧道导致 GPS 信号长时间缺失。图 4.2，图 4.3 和图 4.4 是缺失值在整个时间轴线上的分布

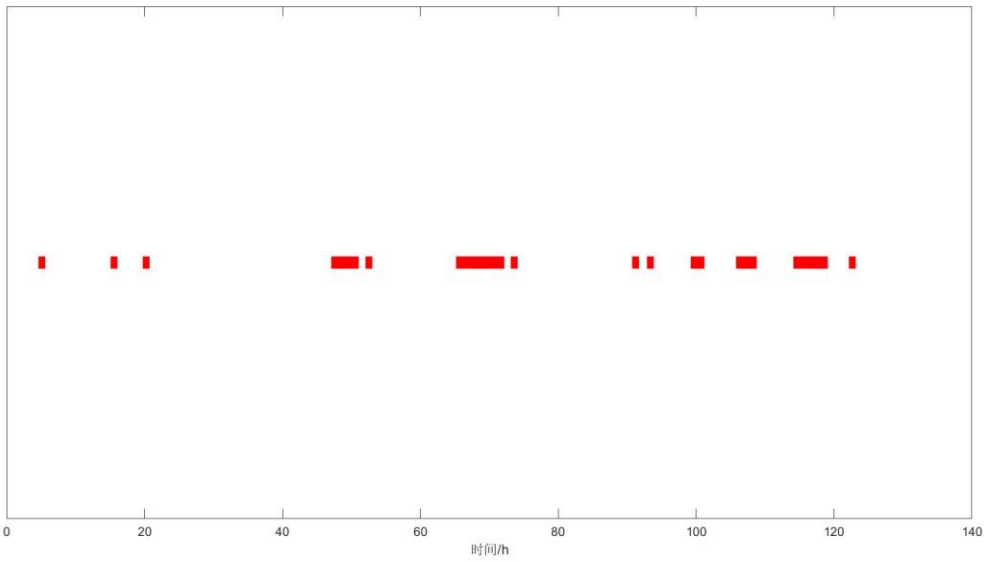


图 4.2　文件一的缺失时间统计图

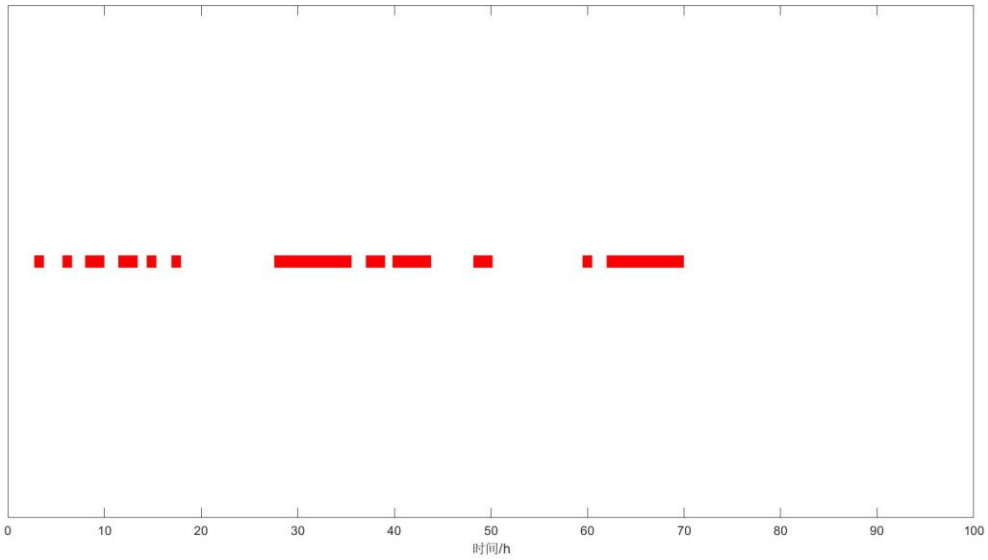


图 4.3　文件二的缺失时间统计图

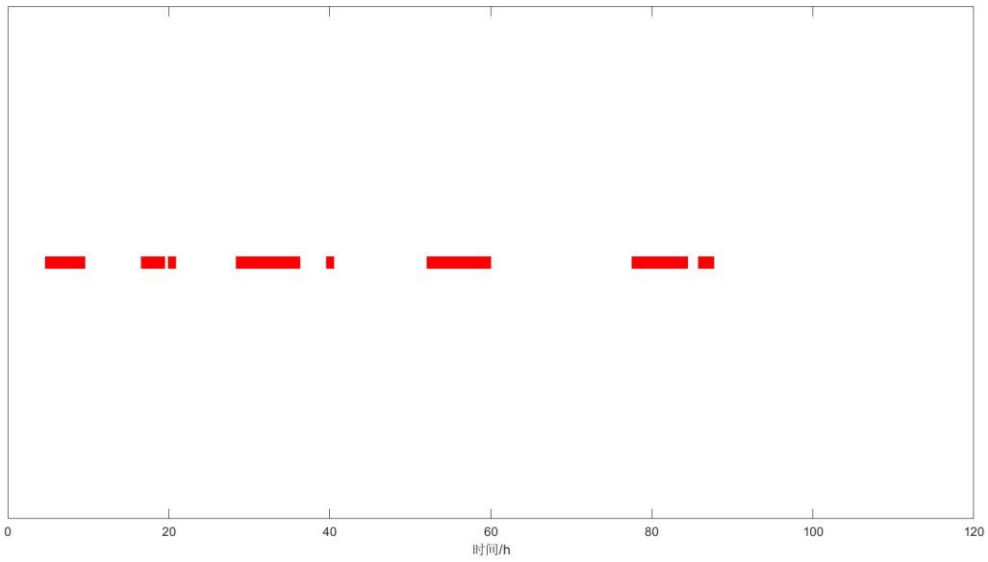


图 4.4　文件三的缺失时间统计图

从三个图中可以看出，部分时间段缺失时间较短，部分时间段缺失时间较长，可以认为在较长缺失时间段汽车不工作。

4.3.2 PMM 模型的求解

通过 PMM 数据填补方法对三个表给出的数据进行填补，填补记录如下表：

表 4.3 三个文件的填补个数记录表

文件	填补个数记录
文件一	42014
文件二	30774
文件三	36462

4.3.3 T4253H 模型的求解

采用 T4253H 滤波器对数据中的白噪声和抖动进行滤波。截取片段的滤波情况如下图

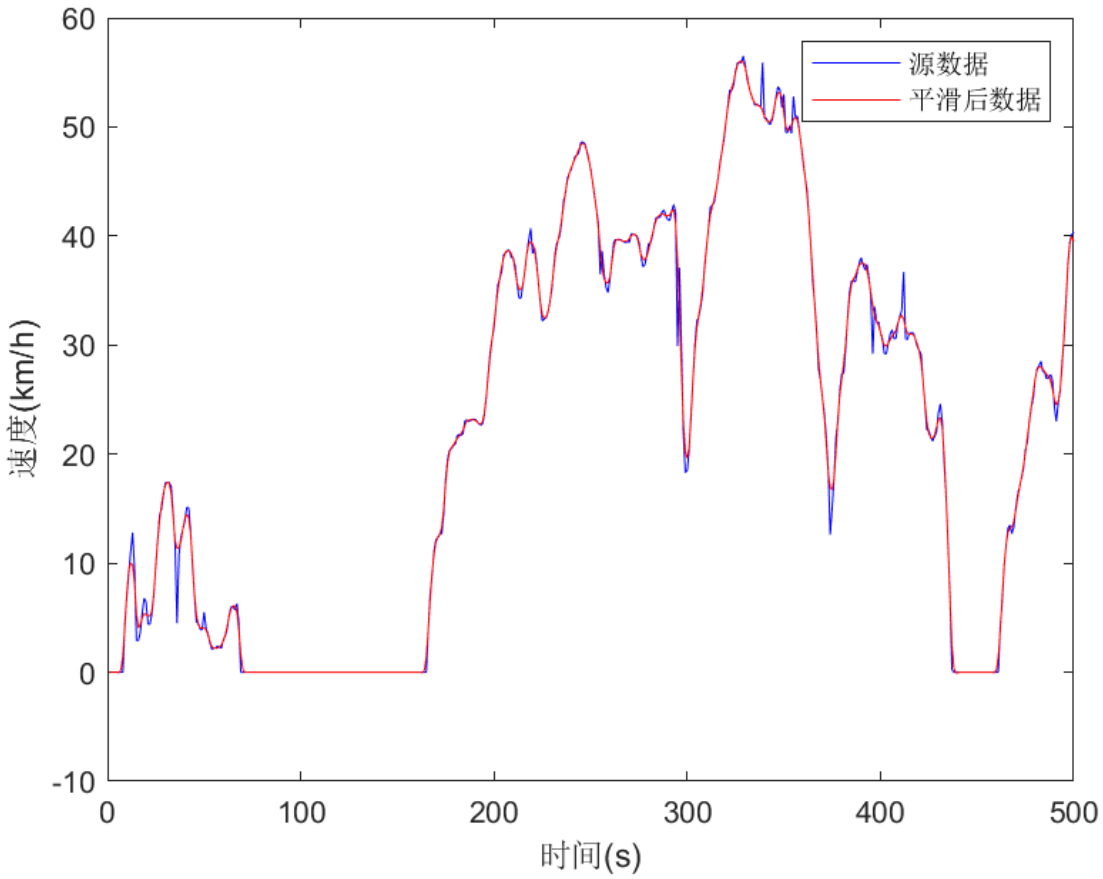


图 4.5 滤波前后示意图

从滤波的情况可以看出，相比较于滤波前，曲线毛刺现象被削弱，曲线相对比较平滑，数据的抖动现象明显降低。

经过数据的预处理之后，三个文件各存在的记录数为：

表 4.4 经过数据预处理完后各个文件的记录数

文件	记录个数
文件一	178735
文件二	143970
文件三	159685

第五章 问题二

5.1 问题分析

基于运动学片段构建汽车行驶工况曲线是目前常用的方法之一。运动学片段构建的好坏直接影响后续对汽车行驶工况的构建，一个好的运动学片段，应该要在很大程度上符合该地区大多数道路的情况，选出的运动学片段要有一定的代表性，这就要求运动学片段的持续时间应该够长，汽车行驶距离也不应大小，同时，极端情况，如加速度或减速度也应该位于合理的区间。

5.2 模型建立

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，在第一问中本文假设了怠速时发动机的最低转速大于 0，提取汽车运动学片段的依据是两个怠速之间的时间段。为了保证截取出来的运动学片段具有代表性且包含足够的信息，本文对运动学片段提出如下要求，只有下述要求被同时满足时，才会被截取出来为第三问构建汽车行驶工况做准备。

（1） 起止于怠速的行程：
运动学片段首先需满足条件：从怠速状态开始至下一个怠速状态开始，即起止于怠速的行程。起止于怠速的行程起始的速度V_{start}和终止速度V_{start}都为 0。数学表达如公式（16）所示：

$$V_{start}+V_{end}=0 \tag{16}$$

（2） 运动学片段时间长度不少于 20s：
所选取的片段时长过去，不具有代表性，不能反映大部分道路的情况，国内大多数学者[6]建立的运动学片段时间长度都不少于 20s。数学表达如公式（17）所示：

$$T_{trip}\geq 20s \tag{17}$$

（3） 加、减速度在合理的范围：
T4325H 滤波是对 PMM 填补后的 GPS 速度数据做进一步的操作（其中包括 PMM 填补时造成的偏差），一个优良的运动学片段应该是有合理的加减速过程，但加减速不能太急促。为了反映一个短程（从怠速状态开始至下一个怠速状态开

始）中整体的加减速情况，本文分别对短程中所有加/减速情况下的加速度做平均，得到加速段的平均加速度 $A_a = mean(\text{所有加速情况下的瞬时加速度})$，减速段的平均减速度 $A_d = mean(\text{所有减速情况下的瞬时减速度})$。对所有短程的 A_a, A_d 做频率统计，如图 5.1，图 5.2 所示：

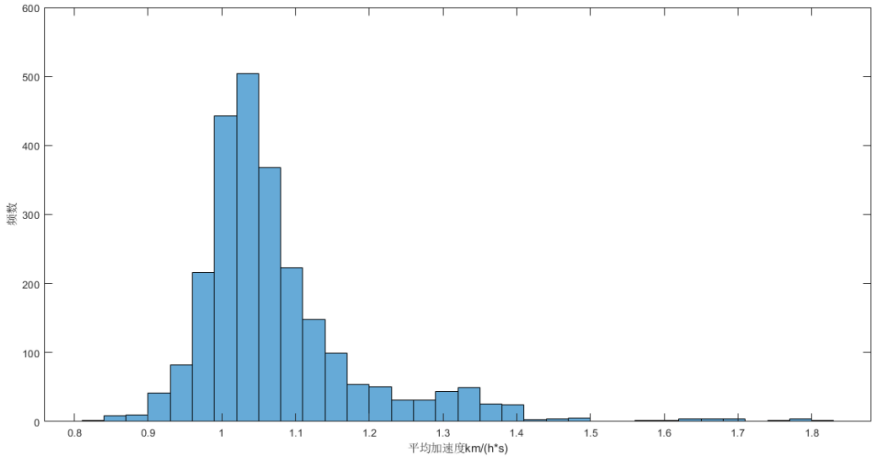


图 5.1　平均加速度分布直方图

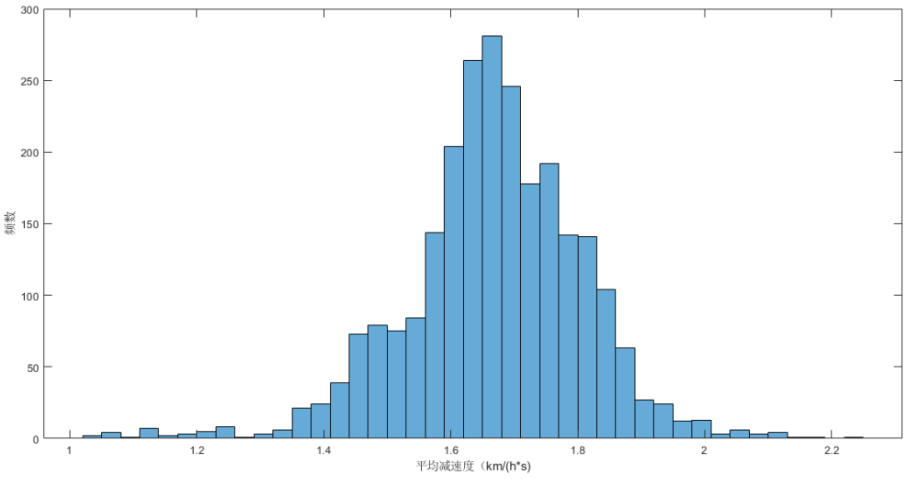


图 5.2　平均减速度分布直方图

由图 5.1，图 5.2 可知车辆加速段的平均加速度总体分布在 0.54 到 14.4（单位是 $Km/(h\cdot s)$）之间所以在筛选合适的运动学片段时选择加速度在 0.54 到 14.4（单位是 $Km/(h\cdot s)$）范围内，减速度在-14.4 到-0.54（单位是 $Km/(h\cdot s)$）范围内。数学表达如公式（18）所示：

$$\begin{cases}(A_a-0.54)(A_a-14.4)\le 0\\(A_d+0.54)(A_d+14.4)\le 0\end{cases} \tag{18}$$

（4） 行驶距离不少于 10 米：

在一个运动学片段内，汽车至少要行驶 10m，否则不能反映实际的行驶过程。数学表达如公式（19）所示：

$$d_s \geq 10m \tag{19}$$

基于运动学片段构建汽车行驶工况曲线是日前最常用的方法之一，通过运动学片段的提取，可以看出汽车在两个怠速之间的行驶状况。不同的运动学片段反映了汽车行驶过程中的状况，同时也侧面反映了当地的道路状况。

5.3 模型求解与结果分析

本题采用 matlab 作为编程工具，根据上一步建立的模型，统计出满足约束的汽车运动学片段如下图分别是 895，641，698 个，三个文件截取的运动学片段如下图 5.3：

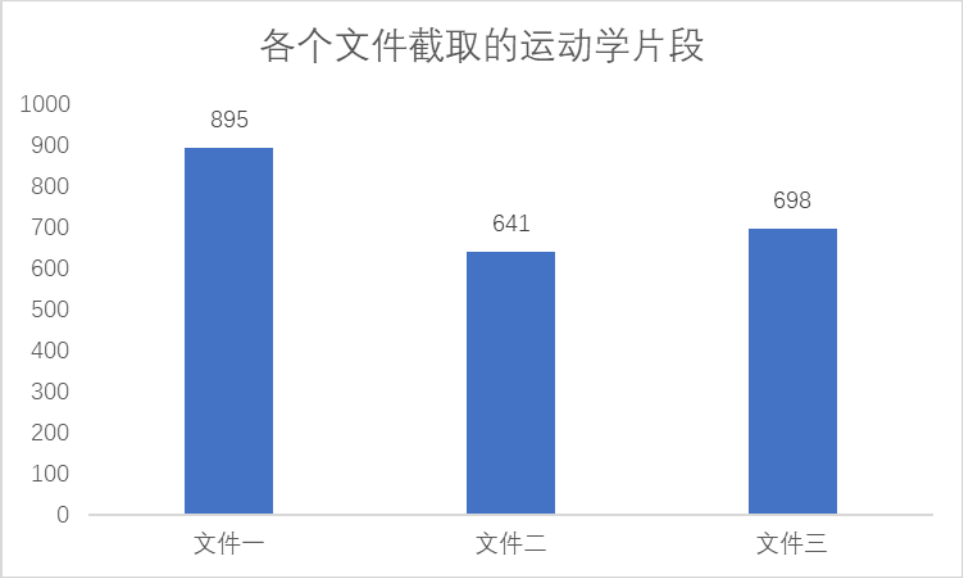


图 5.3 各文件截取的运动学片段

其中时间最短的片段是 21 秒，时间最长的片段由 3179 秒。最长的运动学片段，最短的运动学片段，比例最多的运动学片段选取一个如下图 5.4，图 5.5：

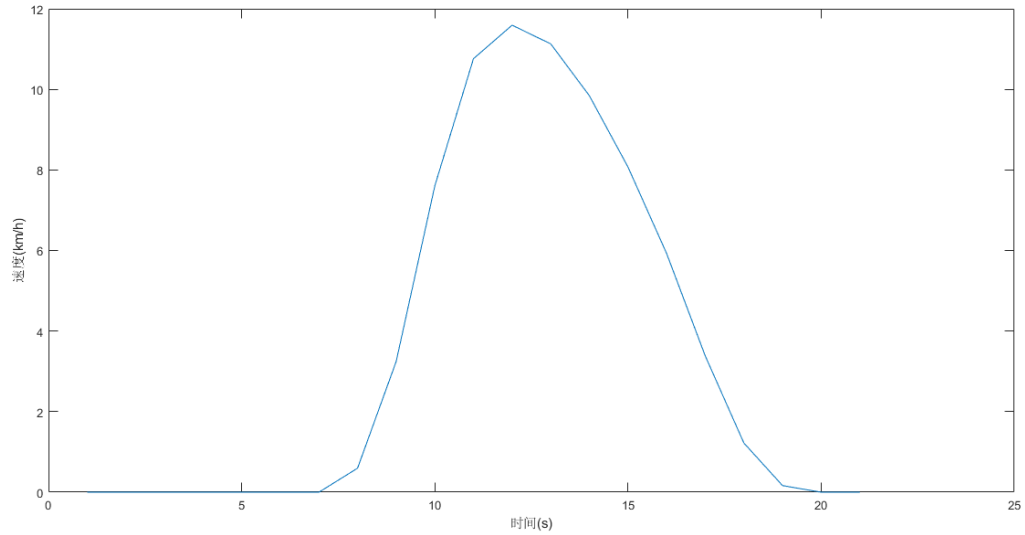


图 5.4 最短运动学片段

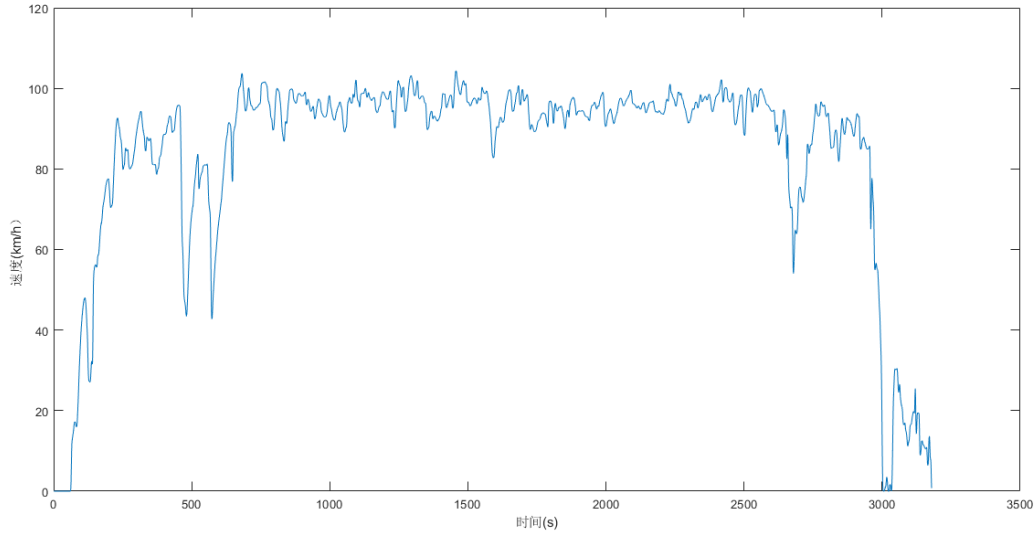


图 5.5 最长运动学片段

两个运动学片段的统计信息如下：

表 5.1 两个运动学片段的统计

指标	平均速度	平均行驶速度	怠速时间比	平均加速度	平均减速度	加速时间比	减速时间比	速度标准差	加速度标准差
最长的运动学片段	83.272	85.037	0.019	0.880	-1.126	0.229	0.185	25.459	0.709
最短的运动学片段	3.499	6.128	0.333	1.236	-1.590	39.333	38.714	4.448	0.942

可以看出，每个运动学片段包含了加速，减速，怠速，等状态信息。

第六章 问题三

6.1 问题分析

在本题中，运动学片段的提取是构建汽车行驶工况图的关键一步，有了良好的运动学片段的数据，通过合理的建模和求解可以构建出良好的汽车行驶工况图。通过第一问的数据的预处理，得到较为完整数据，通过第二问的运动学片段的提取得到足够数量的合理的运动学片段。构建汽车的行驶工况需要依据提取出来的汽车的运动学片段。如利用运动学片段的分布状态和相应的特征参数找出代表性的数据进行汽车工况图的构建。

汽车行驶工况的构建需要根据相应的指标来反应出构建出来的工况的代表性，而大量的运动学相关的指标常常存在极强的相关性，直接用这些指标进行对运动学片段进行挑选和合成具也是一大难点。

6.2 模型建立

针对汽车工况的代表性，本文建立指标模型；而对于大量运动学指标之间常常存在的极强相关性，本文建立了主成分模型；对于不同运动学片段所反映的道路的差异性，本文提出了 kmeans 和高斯混合聚类模型。针对多种方法下建立的工况图的选取问题，本文建立综合熵权法综合评价模型。

通过对数据进行主成分分析，选取主要成分之后，将其用于聚类，以反映不同的道路情况，再在各类中选择有代表性的运动选片段，将其合成最终的工况图，同时对于不同方法不同情况下合成的多个工况图，建立综合评价模型。

6.2.1 指标模型

构建汽车工况图需要找出那些运动学片段是合适的，指标评价的是短行程的整体概况而不是针对于某一个短行程评价，如果单纯的按照每个运动学片段的记录计算会使得计算时间开销很大，且不同的运动学片段数据的维度不同也是计算的难点。所以需要对每个运动学片段刻画，在运动学片段的评估中有很多的参数可以作为刻画指标。通过查阅资料和题目提示选取和汽车行驶工况图比较相关的指标有：平均速度、平均行驶速度、怠速时间比、平均加速度、平均减速度、加速时间比、减速时间比、速度标准差、加速度标准差九个指标作为分析。各自的计算公式如下：

（1） 平均速度V_m：

假设在[1,T]时间周期内，有 T 个速度记录（时间连续，间隔为 1s），则平均速度V_m的数学表达式如公式（20）所示：

$$V_m = \frac{1}{T}\sum_{i=1}^{T} v_i \tag{20}$$

（2） 平均行驶速度V_{on_m}：

假设在[1,T]时间周期内，有 T 个速度记录（时间连续，间隔为 1s），$I(v_i > 0)$

为示性函数，在 $v_i>0$ 时取 1，否则取 0。平均行驶速度 V_{on_m} 的数学表达式如公式（21）所示：

$$V_{on\_m}=\frac{\sum_{i=1}^{T}v_i I(v_i>0)}{\sum_{i=1}^{T}I(v_i>0)} \tag{21}$$

（3） 怠速时间比 T_i^p ：

假设在[1,T]时间周期内，有 T 个速度记录（时间连续，间隔为 1s），$I(v_i=0)$ 为示性函数，在 $v_i=0$ 时取 1，否则取 0。怠速时间比 T_i^p 的数学表达式如公式（22）所示：

$$T_i^p=\frac{1}{T}\sum_{i=1}^{T}I(v_i=0) \tag{22}$$

（4） 平均加速度 A_a：

假设在[1,T]时间周期内，有 T 个速度记录（时间连续，间隔为 1s），$I(a_i>0)$ 为示性函数，在 $a_i>0$ 时取 1，否则取 0。平均加速度 a_{m} 的数学表达式如公式（23）所示：

$$A_a=\frac{\sum_{i=1}^{T}a_i I(a_i>0)}{\sum_{i=1}^{T}I(a_i>0)} \tag{23}$$

（5） 平均减速度 A_d：

假设在[1,T]时间周期内，有 T 个速度记录（时间连续，间隔为 1s），$I(a_i<0)$ 为示性函数，在 $a_i<0$ 时取 1，否则取 0。平均减速度 A_d 的数学表达式如公式（24）所示：

$$A_d=\frac{\sum_{i=1}^{T}\left[-a_i I(a_i<0)\right]}{\sum_{i=1}^{T}I(a_i<0)} \tag{24}$$

（6） 加速时间比 T_a^p：

假设在[1,T]时间周期内，有 T 个速度记录（时间连续，间隔为 1s），$I(a_i<0)$ 为示性函数，在 $a_i<0$ 时取 1，否则取 0。加速时间比 T_a^p 的数学表达式如公式（25）所示：

$$T_a^p=\frac{1}{T}\sum_{i=1}^{T}I(a_i>0) \tag{25}$$

（7） 减速时间比 T_d^p：

假设在[1,T]时间周期内，有 T 个速度记录（时间连续，间隔为 1s），$I(a_i<0)$ 为示性函数，在 $a_i<0$ 时取 1，否则取 0。减速时间比 T_d^p 的数学表达式如公式（26）所示：

$$T_d^p=\frac{1}{T}\sum_{i=1}^{T}I(a_i<0) \tag{26}$$

（8） 速度标准差v_{std}：

假设在[1,T]时间周期内，有 T 个速度记录（时间连续，间隔为 1s），速度标准差v_{std}的数学表达式如公式（27）所示：

$$v_{std}=\left[\frac{1}{T-1}\sum_{i=1}^{T}\left(v_i-V_m\right)^2\right]^{1/2} \tag{27}$$

其中，V_m表示[1,T]时间内所有速度的算术平均值。

（9） 加速度标准差a_{std}：

假设在[1,T]时间周期内，有 T 个速度记录（时间连续，间隔为 1s），加速度标准差a_{std}的数学表达式如公式（28）所示：

$$a_{std}=\left\{\frac{1}{TT-1}\sum_{i=1}^{TT}\left[a_i I\left(a_i>0\right)-A_a\right]^2\right\}^{1/2} \tag{28}$$

其中，$TT=\sum_{i=1}^{T}I\left(a_i>0\right)$ 表示[1,T]时间周期内处于加速状态的记录数目，A_a表示[1,T]时间周期内汽车处于加速状态时加速度的算术平均值。

6.2.2 主成分模型

运动学片段的运动学指标之间常常会有很强的相关性，将其直接用于运动选片段的筛选与合成当中会造成信息冗余，掩盖了数据的主要特征。以下是几种常用的成分分析方法的比较：

表 6.1 常用的降维方法

算法	优点	缺点
主成分分析方法	侧重于信息贡献影响力综合评价，对客观经济现象进行科学评价。在应用上侧重于信息贡献影响力综合评价[7]。	当主成分的因子负荷的符号有正有负时，综合评价函数意义就不明确。命名清晰性低[7]。
线性判别 LDA	可以使用先验知识，依赖于均值而不依赖于方差[8]。	降维最多降到类别数 K-1 的维度。可能过度拟合数据。
LLE 局部线性嵌入	可以学习任意维度的局部线性的低维流形，计算复杂度相对较小容易实现[9]。	算法学习的流形状只能是不闭合的，且样本集是稠密均匀的。

通过对比各个降维方法本文选择主成分分析（PCA）方法作为主要的降维方法。计算流程如下：

（1） 将原始数据标准化，标准化后的数据记为X 。

（2） 计算相关系数矩阵，数学表达式如公式（29）所示：

$$R=\begin{bmatrix} r_{11} & r_{12} & \cdots & r_{1p} \\ r_{21} & r_{22} & \cdots & r_{2p} \\ \cdots & \cdots & \cdots & \cdots \\ r_{p1} & r_{p2} & \cdots & r_{pp} \end{bmatrix} \tag{29}$$

其中$r_{ij}(i,\ j=1,2,\cdots,\ p)$是原始变量的 x_i 与 x_j 之间的相关系数,数学表达式如公式(30)所示:

$$r_{ij}=\frac{\sum_{k=1}^{n}(x_{ki}-\overline{x})(x_{kj}-\overline{x})}{\sqrt{\sum_{k=1}^{n}(x_{ki}-\overline{x})^2\sum_{k=1}^{n}(x_{kj}-\overline{x})^2}} \tag{30}$$

(3) 计算特征与特征向量:

基于特征方程$|\lambda I-R|=0$，求得特征值$\lambda_i(i=1,2,3...p)$，并对其逆序排列即$\lambda_1\geq\lambda_2\geq\lambda_3.....\geq\lambda_p\geq 0$，分别求出与特征值$\lambda_i$相对应的特征向量$e_i(i=1,2,3.....p)$，$e_{ij}$表示向量$e_i$的第 j 个分量。

(4) 计算主要成分的贡献率和累积贡献率:

根据公式（31）计算主成分Z_i 的贡献率:

$$\frac{\lambda_i}{\sum_{k=1}^{p}\lambda_k}(i=1,2.....p) \tag{31}$$

根据公式（32）计算累积贡献率:

$$\frac{\sum_{k=1}^{i}\lambda_k}{\sum_{k=1}^{p}\lambda_k}(i=1,2.....p) \tag{32}$$

一般情况下，选取累计贡献率不少于85%的特征值$\lambda_1,\lambda_2.....\lambda_m$对应的第一，二，$m$（$m\leq p$）个主要成分。

（5） 根据公式（33）计算m 个相应的单位特征向量：

$$\begin{cases} e_1=\left(e_{11},e_{12},\cdots,e_{1p}\right)^T \\ e_2=\left(e_{21},e_{22},\cdots,e_{2p}\right)^T \\ \cdots \\ e_m=\left(e_{m1},e_{m2},\cdots,e_{mp}\right)^T \end{cases} \tag{33}$$

（6） 根据公式（34）计算主成分：

$$Z_i=e_{1i}X_1+e_{2i}X_2+\cdots+e_{pi}X_p,i\in\{1,2,\cdots,p\} \tag{34}$$

6.2.3 高斯混合聚类模型

本文首先采用混合高斯聚类进行数据分类。查阅文献[10]并对高斯混合 EM 聚类的模型进行归纳总结，具体细节如下所示。首先确定高斯混合分布的数目 k（也是聚类的目标数量），高斯混合分布函数如公式（35）所示：

$$f_G(x)=\sum_{i=1}^{k} w_i f\left(x;\mu_i,\sigma_i\right) \tag{35}$$

其中，$w_i\in[0,1]$为混合系数，满足$\sum_i^k w_i=1$ 。$f\left(x;\mu_i,\sigma_i\right)$为高斯概率密度函数，$\mu_i$和$\sigma_i$ 分别表示高斯概率密度函数的均值向量和协方差矩阵。

E 步：

根据公式（36），可计算初始数据集$P=\{P_1,P_2,\cdots,P_m\}$ 中任一元素P_j的基于第 i 个高斯混合成分的后验概率。

$$f_G\left(i\mid P_j\right)=\frac{w_i f\left(P_j\mid\mu_i,\sigma_i\right)}{\sum_{l=1}^{k} w_l f\left(P_j\mid\mu_l,\sigma_l\right)} \tag{36}$$

M 步：

对μ_i， σ_i和w_i做极大似然估计，数学表达式如公式（37）-（39）所示：

$$\mu_i=\frac{\sum_{j=1}^{m} f_G\left(i\mid P_j\right)P_j}{\sum_{j=1}^{m} f_G\left(i\mid P_j\right)} \tag{37}$$

$$\sigma_i=\frac{\sum_{j=1}^{m} f_G\left(i\mid P_j\right)\left(P_j-\mu_i\right)\left(P_j-\mu_i\right)^{\mathrm{T}}}{\sum_{j=1}^{m} f_G\left(i\mid P_j\right)} \tag{38}$$

$$w_i=\frac{1}{m}\sum_{j=1}^{m} f_G\left(i\mid P_j\right) \tag{39}$$

对 E 步和 M 步进行重复操作，直到终止条件满足，以迭代模型参数。将迭代最后得到的模型参数带入公式（35）得到高斯混合分布，然后将数据集P聚成 k 个类$P=\{P_1,P_2,\cdots,P_k\}$。

在经过聚类以后，可以得到D个运动学片段簇。每个簇中的运动学片段距离簇中心的距离大小不一，对距离D_{ij} 进行排序得到序列A_i 。

假设构建的工况总时长为T_{slice}，计算每个簇中片段的时间总量T_{all-i} ,并按照每个簇的时间总值在所有运动学片段中之间总和的比值作为该簇所代表的运动学情况在总的工况图中的比值，数学表达如公式（40）所示：

$$T_i=\frac{T_{all-i}}{\sum_{i=1}^{D} T_{all-i}}T_{slice} \tag{40}$$

从每个簇中按照距离簇中心距离从小大选取相应的片段合成工况图。从簇中选取多个片段合并会存在某个片段刚好跨越临界值T_i的情况，对此做切割处理。

6.2.4 K-means 聚类

由于高斯混合聚类计算量较大。如果其中一个聚类的数据并不服从正态分布、偏态分布，聚类算法会出现偏差[11]。所以本文选取 K-均值聚类作为对比，成分分析选取了贡献值较大的四个成分作为数据分类的指标，使用 K-均值聚类的步骤方法如下：

K-均值聚类（又称 K-Means 聚类），基于事先给定的目标聚类数 K 和 K 个初始聚类中心点，迭代计算，当 K 个类中的元素不再变化时停止迭代。此时，类内样本相似度高，类间样本相似度低。假设待聚类的样本为 $P=\{P_1,P_2,\cdots,P_m\}$，$P_i,i\in\{1,2,\cdots,m\}$ 为 n 维向量，给定目标聚类数 K 和初始聚类中心点 $O_1,O_2,\cdots,O_K$，$O_i,i\in\{1,2,\cdots,K\}$ 为 n 维向量。对下面 2 步进行迭代，直到类中心点不变或变化少于给定数值[12]。

（1） 根据公式（41）计算，得到样本 $P_i,i\in\{1,2,\cdots,m\}$ 应属于的类 s^i。

$$s^i:=\min_{j\in\{1,2,\cdots,K\}}\left\|P_i-O_j\right\|^2 \tag{41}$$

(2) 重复（1），得到一个聚类结果 $P=\{L_1,L_2,\cdots,L_K\}$，根据公式（42）计算类 $L_j,j\in\{1,2,\cdots,K\}$ 的中心点 O_j。

$$O_j:=\frac{\sum_{i=1}^{m}I\{s^i=j\}P_i}{\sum_{i=1}^{m}I\{s^i=j\}} \tag{42}$$

6.2.5 熵权法综合评价模型

对于数据集的不同的处理操作会产生不同的数据集，比如填补方法的不同会使得填补结果接近真实值的情况不同，不同的滤波操作会对结果的分析产生影响，也就是不同的数据的处理与不同的聚类方法组合成多个汽车行驶工况，为了说明本实验选取的工况最好所以需要建立一个工况图的选取方式（本文总体基于标签 8 进行建模求解）。对多种组合情况下的工况图进行比对得出最好的工况图。多种数据操作处理方法和多种聚类方式下的工况图结合如下：

- 异常值处理后数据(第一题第一步)
  - 缺失数据直接删除
    - 高斯混合聚类（标签1）
    - $K-$means聚类（标签2）
  - 缺失数据PMM填补
    - 全局填补
      - $T4253H$直接滤波
        - 高斯混合聚类（标签3）
        - $K-$means聚类（标签4）
      - $T4253H$对0不滤波
        - 高斯混合聚类（标签5）
        - $K-$means聚类（标签6）
    - 滑动窗口填补
      - $T4253H$直接滤波
        - 高斯混合聚类（标签7）
        - $K-$means聚类（标签8）
      - $T4253H$对0不滤波
        - 高斯混合聚类（标签9）
        - $K-$means聚类（标签10）

图 6.1 多种数据和聚类的组合

评价方法如下：

对五种经过预处理的数据提取运动学片段（缺失数据直接删除、T5453 直接滤波（全局填补）、T5453H 对 0 的不滤波（全局填补）、T5453H 直接滤波（滑动窗口填补）、T5453H 对 0 不滤波（滑动窗口填补））通过熵权赋值法对所有运动学片段的特征值（第一步中的九个指标）进行熵权赋值。

熵权赋值法是一种客观赋值方法，根据各个指标的变异程度，利用信息熵计算出各个指标的熵权，再通过熵权对各个指标的权重进行修正，从而得出较为客观的指标权重[13]。

假设每种状态的概率为p_i(i=1,2,3.....m)，该系统的熵 e，数学表达如公式（43）所示：

$$e=-\sum_{i}^{m} p_i \times \ln p_i \tag{43}$$

现有 m 个待评价的工况图，n 个评价指标，原始评价矩阵 $R=(r_{ij})_{m\times n}$ 对于某个指标 r_{ij} 有信息熵 e_j，数学表达如公式（44）所示：

$$e_j=-\sum_{i}^{m} p_{ij} \times \ln p_{ij} \tag{44}$$

可以看出，如果某个指标的熵值 e_{ij} 越小，说明这个指标的变化程度较大，则该指标在评价中的作用越大，其权重应该越大。

6.3 模型求解与结果分析

6.3.1 主成分模型的求解

这里本文采用 matlab 作为求解模型的语言工具，使用主成分分析方法（PCA）进行数据的维度选择工作，九个指标对数据的贡献分布如表 6.2：

表 6.2 主成分贡献率和累积贡献率

	特征值	贡献率（%）	累计贡献（%）
成分 1	3.696	41.071	41.071
成分 2	2.049	22.761	63.832
成分 3	1.408	15.647	79.479
成分 4	0.824	9.158	88.636
成分 5	0.593	6.593	95.229
成分 6	0.241	2.673	97.902
成分 7	0.174	1.937	99.838
成分 8	0.010	0.113	99.952
成分 9	0.004	0.048	100.000

主成分系数矩阵如下表 6.3

表 6.3 主成系数分矩阵表

成份 指标	成分 1	成分 2	成分 3	成分 4	成分 5	成分 6	成分 7	成分 8	成分 9
平均速度	0.467	-0.045	0.310	0.009	0.020	-0.089	-0.528	-0.629	-0.034
平均行驶速度	0.480	-0.066	0.188	0.254	0.148	0.021	-0.327	0.731	0.035
怠速时间比	-0.219	-0.037	-0.519	0.637	0.385	0.117	-0.296	-0.162	-0.006
平均加速度	0.041	0.631	-0.067	-0.232	0.014	0.708	-0.199	0.014	0.032
平均减速度	0.012	-0.502	-0.057	-0.512	0.644	0.258	-0.006	-0.002	0.019
加速时间比	-0.385	-0.002	0.537	0.183	0.142	0.089	-0.039	-0.032	0.706
减速时间比	-0.391	0.005	0.528	0.158	0.151	0.133	-0.054	0.042	-0.705
速度标准差	0.443	-0.024	0.150	0.379	0.162	0.327	0.680	-0.201	-0.016
加速度标准差	0.063	0.584	0.039	-0.105	0.587	-0.527	0.142	-0.002	-0.010

得分矩阵的部分数据如下表 6.4（全部数据见附录二）：

表 6.4 主成分分析得分矩阵

成份 观测	成分 1	成分 2	成分 3	成分 4	成分 5	成分 6	成分 7	成分 8	成分 9
观测 1	1.752	2.211	-0.818	-1.836	1.405	1.120	-0.336	0.078	0.103
观测 2	-1.068	2.536	-2.962	-1.173	1.562	0.781	-0.682	-0.133	0.099
观测 3	-0.766	2.507	-1.414	-2.198	0.765	0.751	-0.050	0.021	0.112
观测 4	1.674	1.709	-0.548	-2.557	1.254	1.180	-0.380	0.029	0.106
观测…	…	…	…	…	…	…	…	…	…
观测 N-1	0.038	-0.844	0.536	0.382	-0.486	-0.342	0.274	-0.021	0.046
观测 N-2	3.717	-1.384	0.866	0.644	-0.218	-0.141	-0.213	-0.127	-0.032

每个成分的贡献碎石图如图 6.2 所示

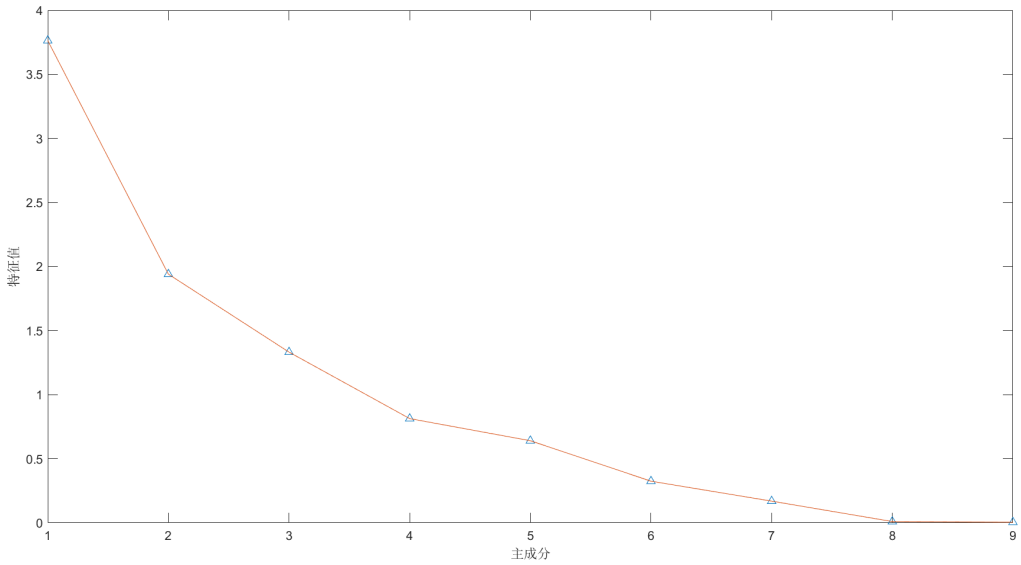


图 6.2 成分的碎石图

根据主成分贡献率的碎石图可以看出前面四个成分的累计贡献是 88.636%，成分累积贡献达到 85%[14]可以认为这几个成分可以代表原始数据的大部分信息。所以选取 4 个成分作聚类分析。

6.3.2 高斯混合聚类模型的求解

这里本文选择了高斯混合聚类作为比对。将经过主成分分析后的指标作为聚类的特征，对选取出来的汽车运动学片段进行高斯混合聚类。

对聚类结果进行分析，Calinski-Harabasz(CH)评价结果如下图 6.3

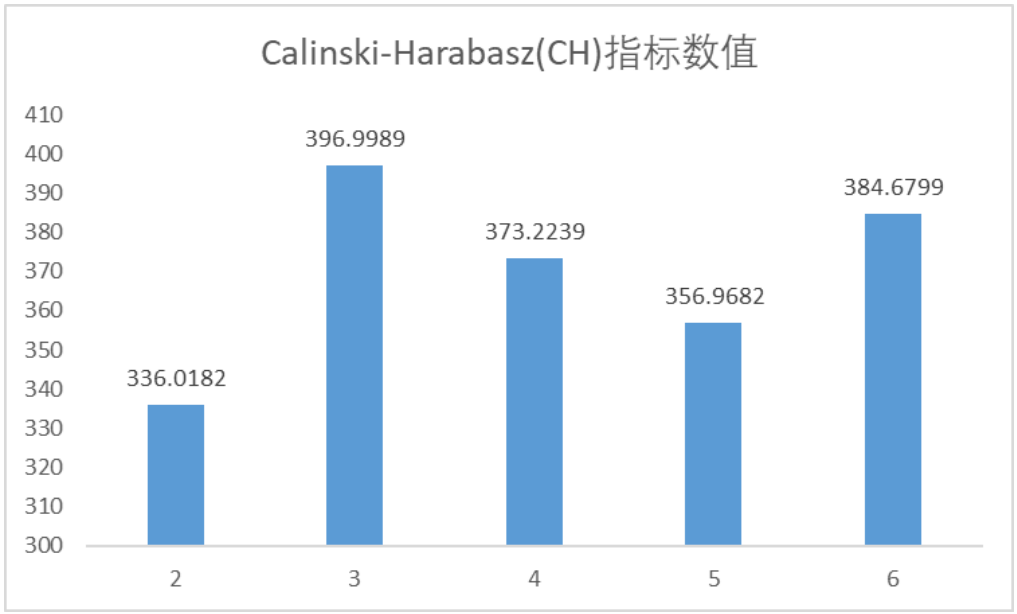


图 6.3 高斯混合聚类数量和 CH 指标变化图

通过聚类的指标判定表可以看出，当聚成 3 类的时候数 CH 数值最大，这个时候说明聚类效果最好，所以本文将运动学片段聚成 3 类，用来进行工况的构建。

6.3.3 k-均值模型的求解

将以主成分分析结果的指标作为聚类的特征对所有的运动学片段进行聚类，采用 Calinski-Harabasz(CH)指标作为聚类结果评价的指标。

对聚类结果进行分析，Calinski-Harabasz(CH)评价结果如下图 6.4

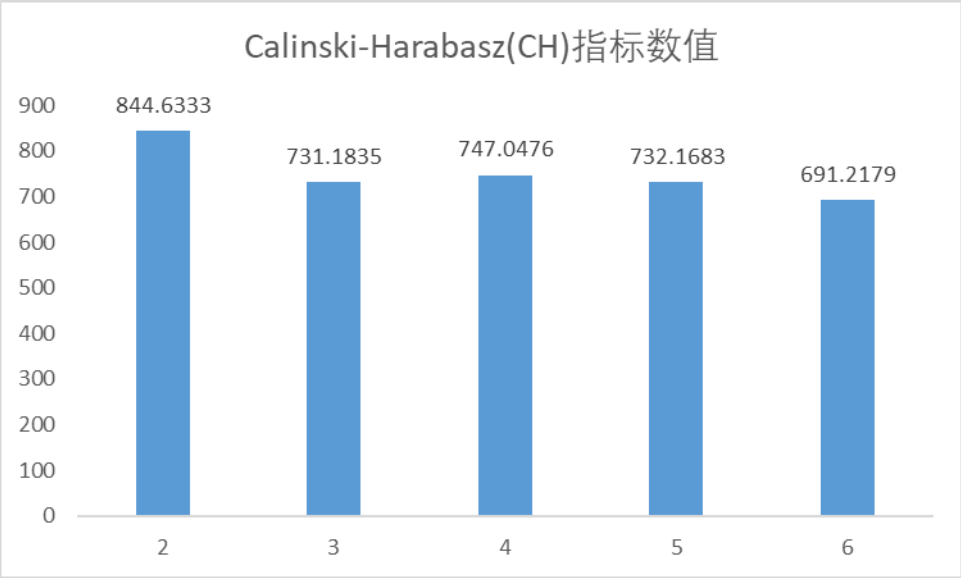


图 6.4　k-均值聚类聚类数量和 CH 指标变化图

通过聚类的指标判定表可以看出，当聚成 2 类的时候数 CH 数值最大，这个时候说明聚类效果最好，所以本文将运动学片段聚成 2 类，用来进行工况的构建。

6.3.4 工况的构建结果

对于 T5453H 直接滤波（滑动窗口填补）处理后的数据进行工况的构建，使用高斯混合聚类方法和 kmeans 方法两种聚类方法作为对比，得到对应的工况图如下图 6.5，图 6.6 所示

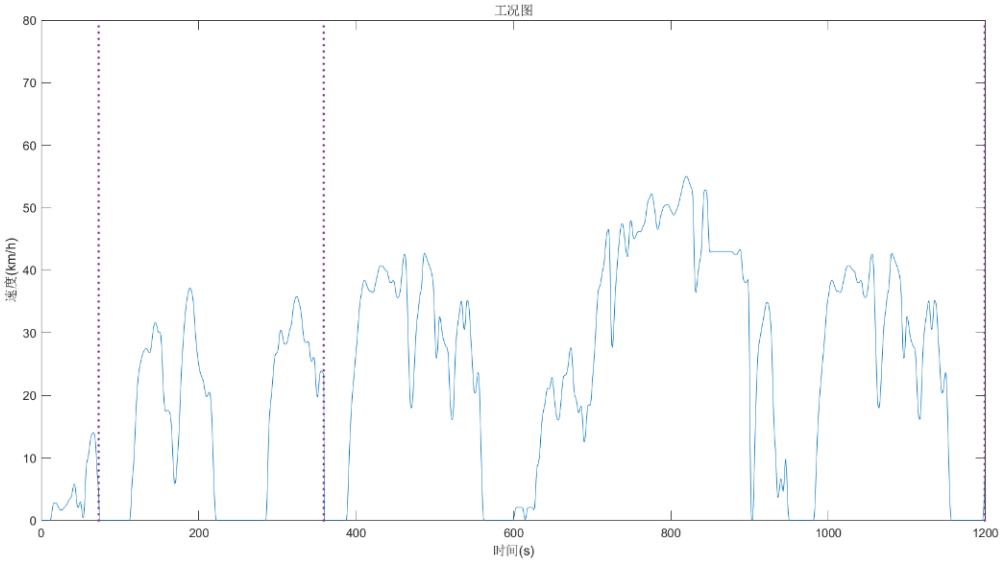


图 6.5 采用高斯混合聚类的工况图（标签 7 工况图）

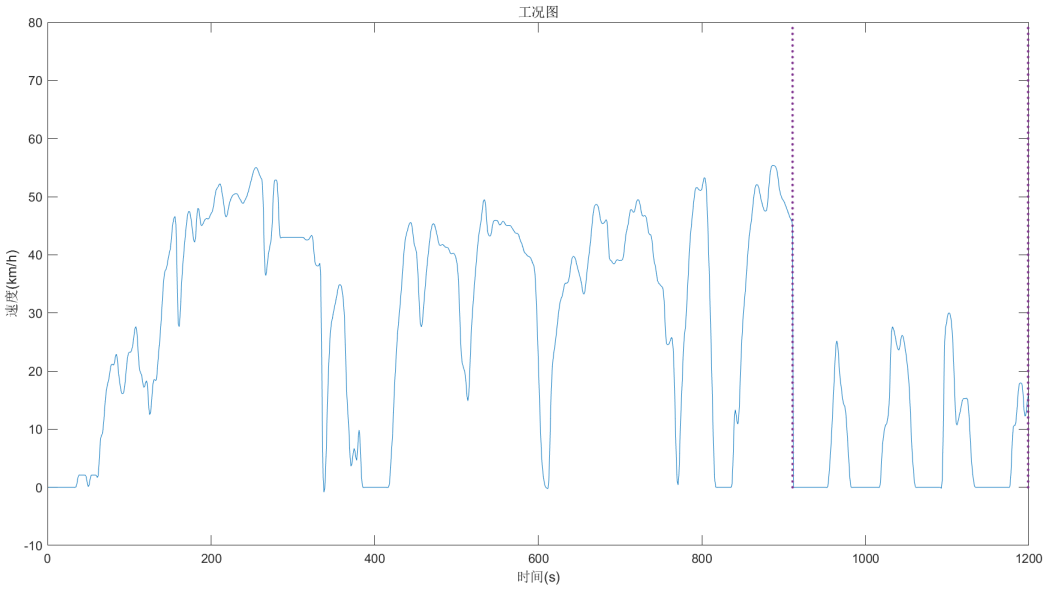


图 6.6 采用 K-means 聚类的工况图（标签 8 工况图）

从图 6.5，图 6.6 中可以，当聚类方法使用高斯混合聚类是，得到的工况图被分成三类，大致代表着低速、中速和高速路段，而采用 K-means 聚类得到的工况图被分成了两类，明显地可以看到左边部分速度较高，而右边部分速度较低，分别代表着低速路段和高速路段。

两种聚类方法的误差结果如表 6.5

表 6.5 误差结果分析表

指标 / 工况	平均速度	平均行驶速度	平均加速度	平均减速度	怠速时间比	加速时间比	减速时间比	速度标准差	加速度标准差
经过混合高斯聚类的工况	0.329	0.359	0.132	0.030	0.095	0.195	0.215	0.434	0.373
经过 K-means 聚类的工况	0.050	0.078	0.122	0.185	0.036	0.057	0.170	0.168	0.167

两种方法对应的构建出来的指标如下图 6.7

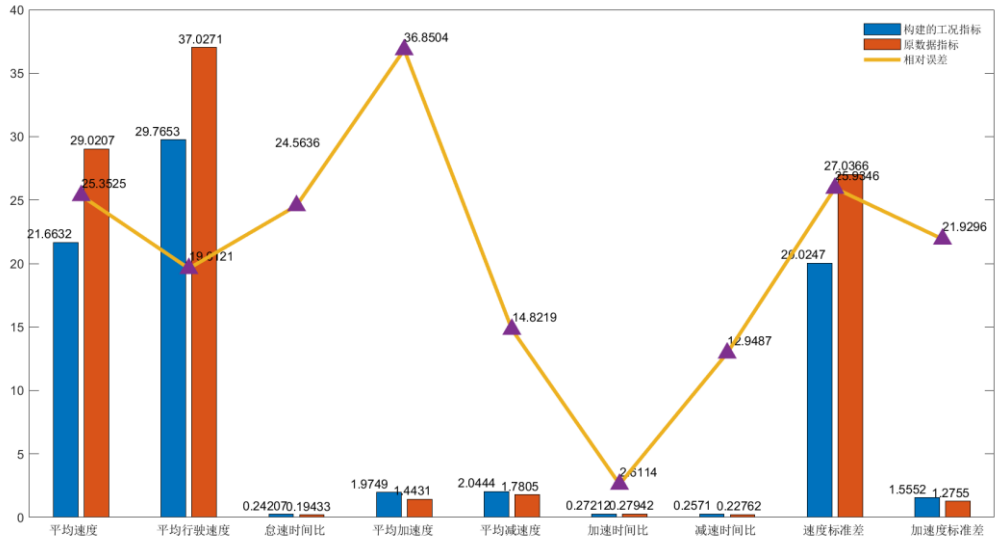


图 6.7 基于高斯模糊聚类的误差分析图

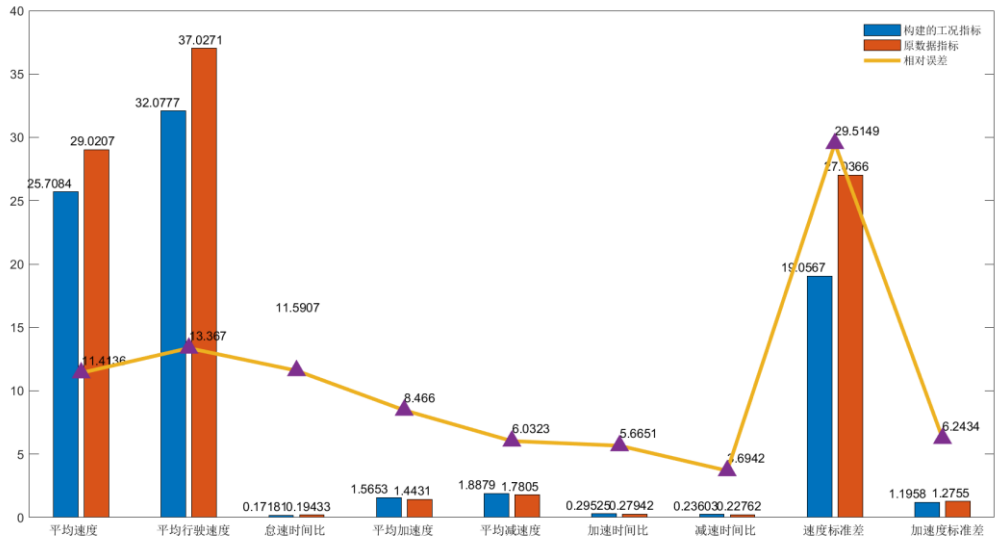


图 6.8 基于 K-均值聚类的误差分析图

从图 6.7,图 6.8 中可以看出，基于 K-均值聚类的工况的误差在除了加速时间比和速度标准差两个指标比高斯混合聚类低高，其它指标相对较低，说明 K-Means 聚类算法在构建工况的过程中效果优于高斯混合聚类。

6.3.5 熵权法综合评价模型的求解

对于图 6.1 的 5 个数据处：缺失数据直接删除、T5453 直接滤波（全局填补）、T5453H 对 0 的不滤波（全局填补）、T5453H 直接滤波（滑动窗口填补）、T5453H 对 0 不滤波（滑动窗口填补）），将其形成的 5 个数据集合成一个数据集，样本是来自 5 个数据集的运动学片段，指标是 9 个运动学指标，对其进行熵权法赋权，得到各指标权重为表 6.6

表 6.6 各个指标权重

指标	平均速度	平均行驶速度	怠速时间比	平均加速度	平均减速度	加速时间比	减速时间比	速度标准差	加速度标准差
权重	0.137	0.106	0.223	0.037	0.007	0.195	0.192	0.087	0.017

各自合成的工况图如附录一（标签 7 工况图和标签 8 工况图已经在上面给出，所以不再重复给出）。

综合评价的结果如下表 6.7

表 6.7 综合评价的结果

标签	1	2	3	4	5	6	7	8	9	10
综合评价得分	0.142	0.167	0.154	0.159	0.172	0.139	0.181	0.104	0.238	0.110

通过对比可以得出 T5453H 直接滤波（滑动窗口填补），也就是标签 8，是效果最好的，其对应的误差分值为 0.104，比较符合实际的情况。以上模型的程序部分见附录三。

第八章 总结

基于给定的汽车行驶数据，通过数据的预处理以及提取运动学片段，构建汽车行驶工况图。在数据的预处理阶段，依次采用不良数据处理、滑动窗口的 PMM 填补和 T4253H 非线性滤波对数据进行处理。接着依据本文提出的 5 个规则对预处理后的数据进行运动学片段的提取，三个文件一共提取出 2234 个运动学片段。然后通过主成分分析和两种聚类方法（K-均值和高斯混合聚类）对运动学片段进行分类。构建两种聚类下汽车行驶工况图并和所采集的数据源的特征进行，对比得到最优工况图。

为了评价构建的汽车行驶工况是否合理，基于构建的汽车工况和采集的数据源之间 9 个运动特征评估指标的相对误差值，通过熵权赋值法对各指标赋权，建立基于综合评价的汽车工况选择体系。综合评价得到：基于滑动窗口 PMM 填补的 K-means 聚类构建的汽车工况最具代表性，综合评价得分为 0.104，即综合的相对误差最小。

参考文献

[1]徐韬. 基于浮动车数据的道路运行车速动态预测研究[D].重庆交通大学,2017.
[2]茅群霞. 缺失值处理统计方法的模拟比较研究及应用[D].四川大学,2005.
[3]Landrum M B , Becker M P . A multiple imputation strategy for incomplete longitudinal data[J]. Statistics in medicine, 2001, 20(17-18):2741-2760.
[4]石敏. 轻型汽车行驶工况构建的研究[D].天津理工大学,2014.
[5] IBM 官网, IBM SPSS Statistics 25 Documentation, https://www.ibm.com/support/pages/ibm-spss-statistics-25-documentation#en, 2019.9.22.
[6]王楠楠. 城市道路行驶工况构建及油耗研究[D].合肥工业大学,2012.
[7] 主成分分析、聚类分析、因子分析的基本思想及优缺点 http://blog.sina.com.cn/s/blog_67fcf49e0101g1lt.html 2019.9.23
[8] 线性判别分析 LDA 原理总结 https://www.cnblogs.com/pinard/p/6244265.html 2019.9.23
[9] 局部线性嵌入(LLE)原理总结 https://www.cnblogs.com/pinard/p/6266408.html?utm_source=itdadao&utm_medium=referral 2019.9.23
[10] 周志华. 机器学习[M]. 北京: 清华大学出版社, 2016.
[11] 聚类算法比较：K-means 和高斯混合模型 https://segmentfault.com/a/1190000009294693?utm_source=tag-newest 2019.9.23
[12]吴文静,景鹏,贾洪飞,张铭航.基于 K 均值聚类与随机森林算法的居民低碳出行意向数据挖掘[J].华南理工大学学报(自然科学版),2019,47(07):105-111.
[13]林丽钦,孙福明.基于熵权赋值法的企业并购内部控制评价体系研究[J].科技和产业,2013,13(12):119-123+136.
[14]何晓群.多元统计分析[M].北京:中国人民大学出版社,2004:142-153.

附 录

附录清单
附录 1 图片
附录 2 表
附录 3 代码

附录一

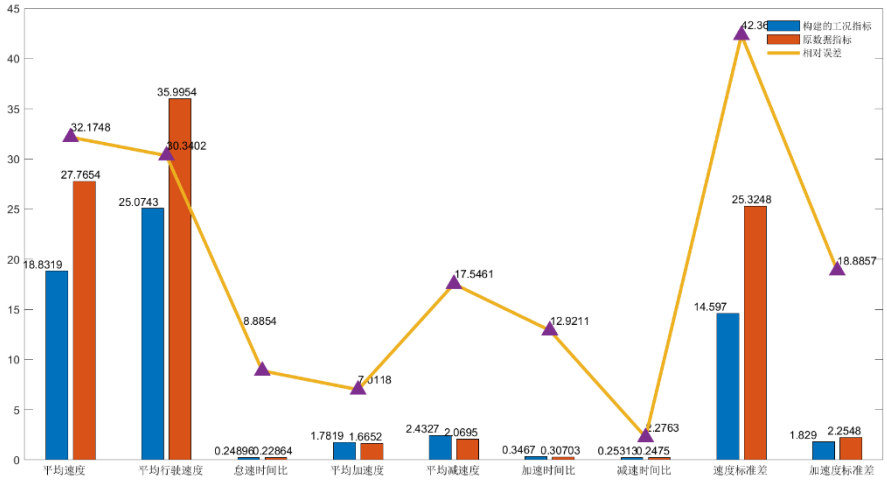


图 1 无对缺失值进行填 GMM 补构建的工况精度图

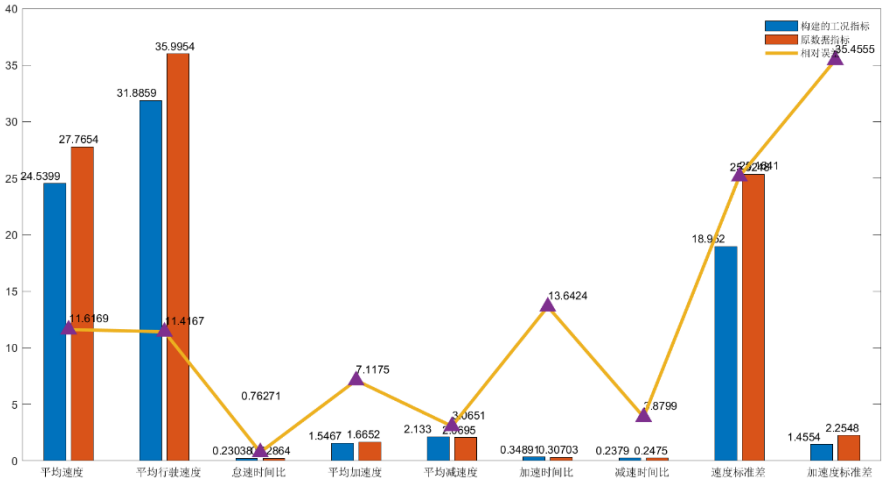


图 2 无对缺失值进行填补 Kmeans 补构建的工况精度图

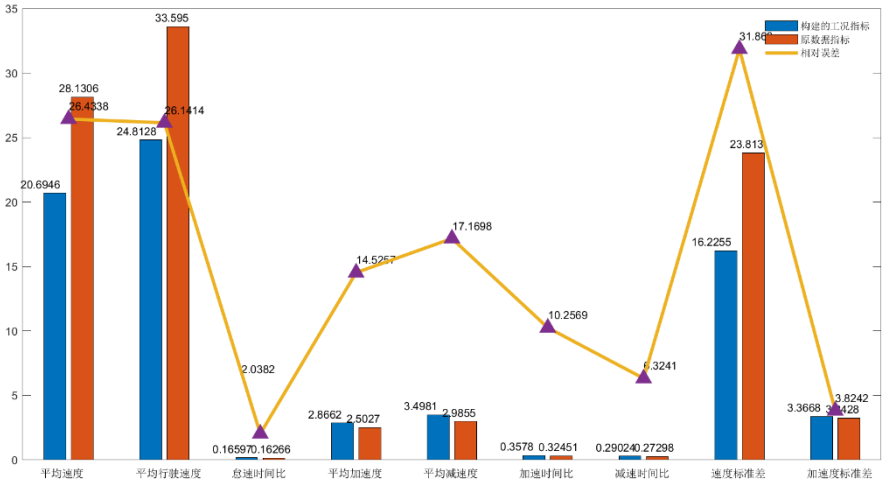


图 3 整体 PMM 填补 GMM 补构建的工况精度图

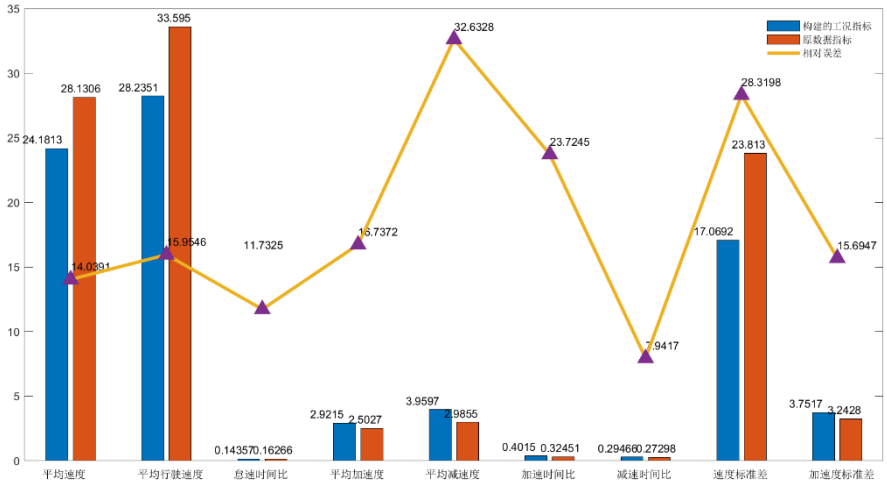


图 4 整体 PMM 填补 Kmeans 补构建的工况精度图

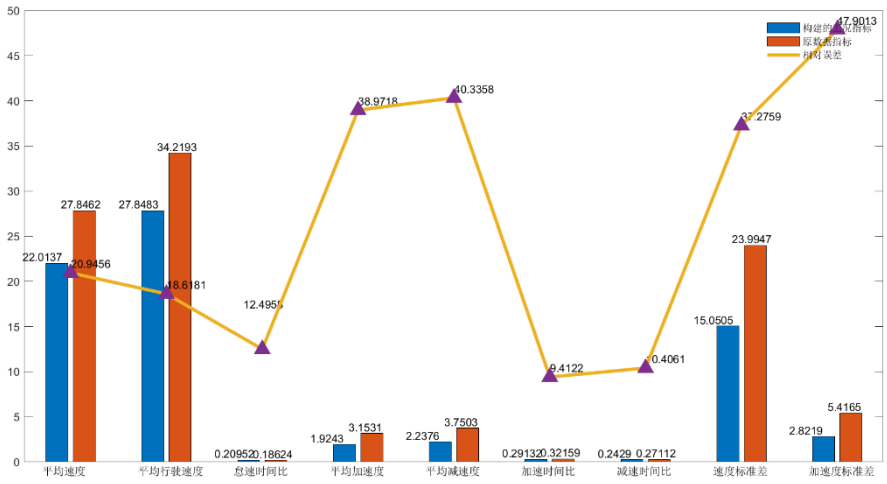


图 5 整体 PMM 对 0 不滤波 GMM 构建的工况精度图

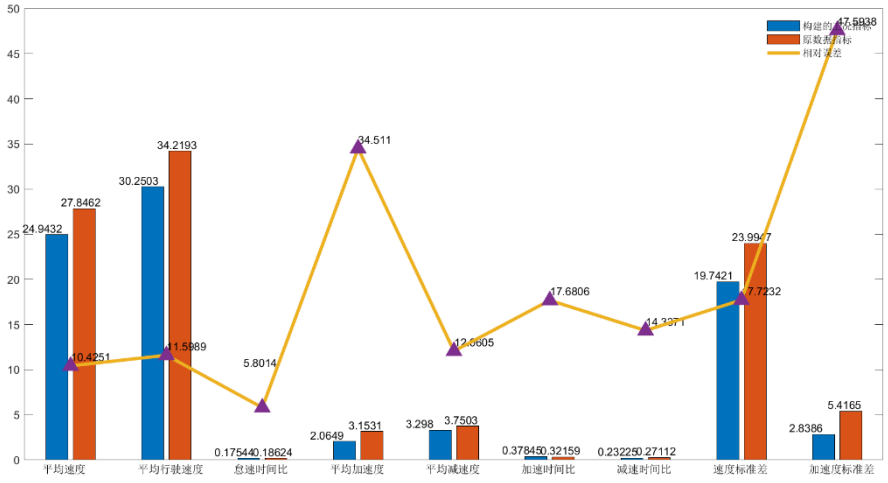


图 6 整体 PMM 对 0 不滤波 Kmeans 构建的工况精度图

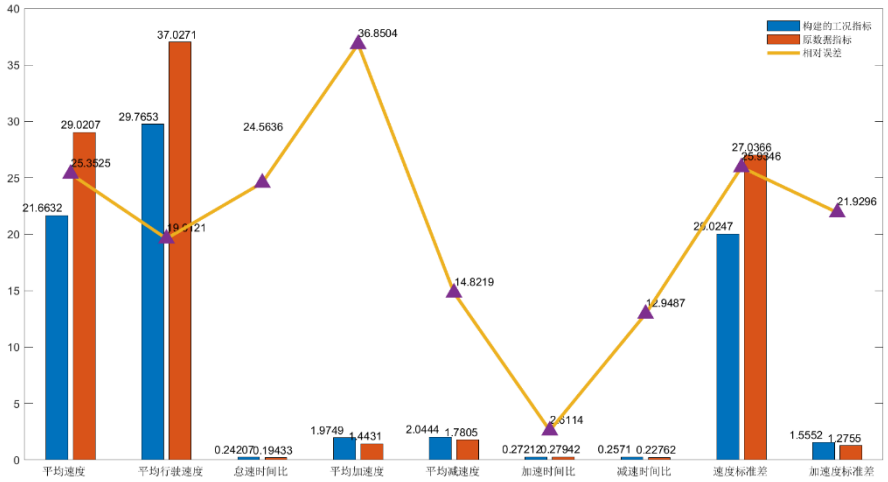


图 7 滑动窗口 PMM 填补 GMM 构建的工况精度图

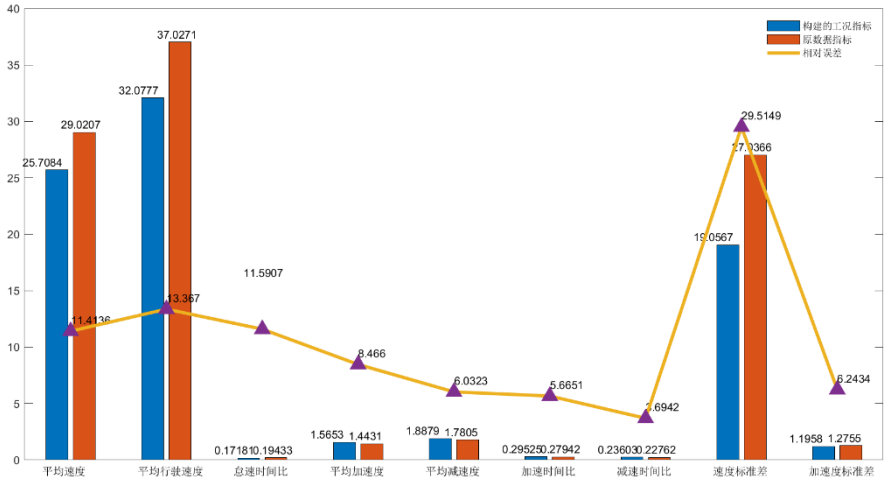


图 8 滑动窗口 PMM 填补 Kmeans 构建的工况精度图

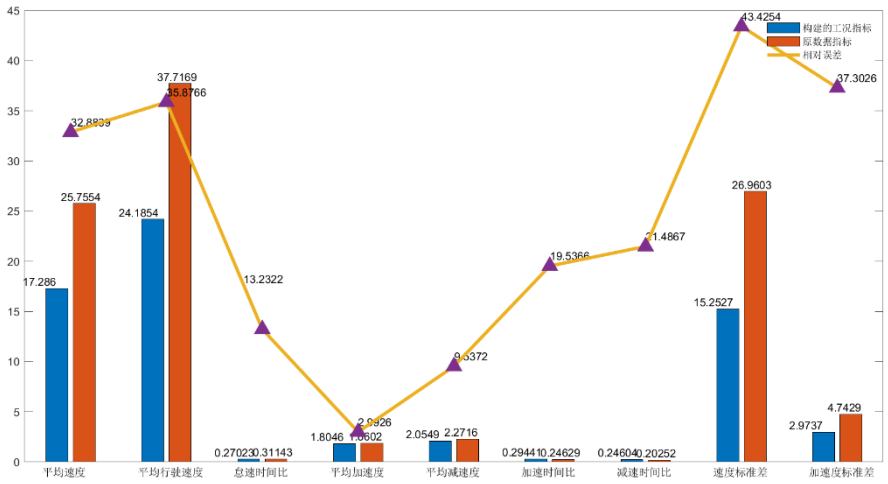


图 9 滑动窗口 PMM 对 0 不滤波 GMM 构建的工况精度图

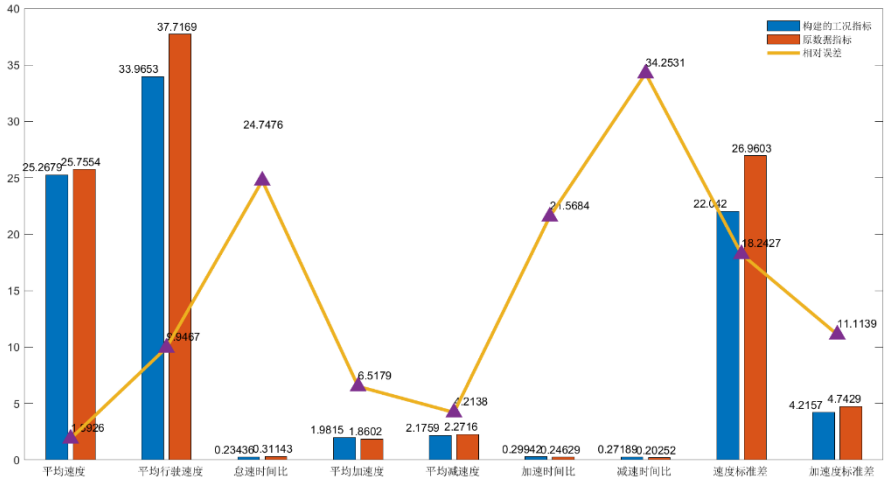


图 10 滑动窗口 PMM 对 0 不滤波 Kmeans 构建的工况精度图

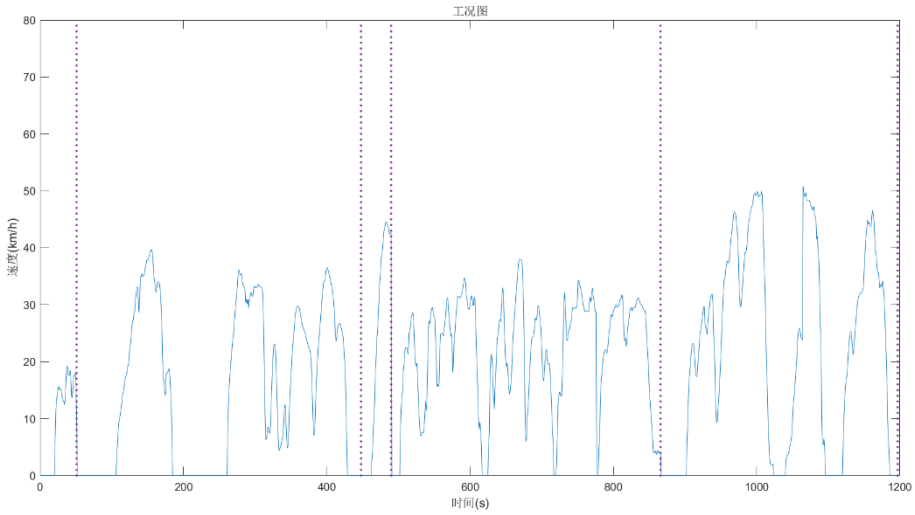


图 11 无对缺失值进行填 GMM 补构建的工况

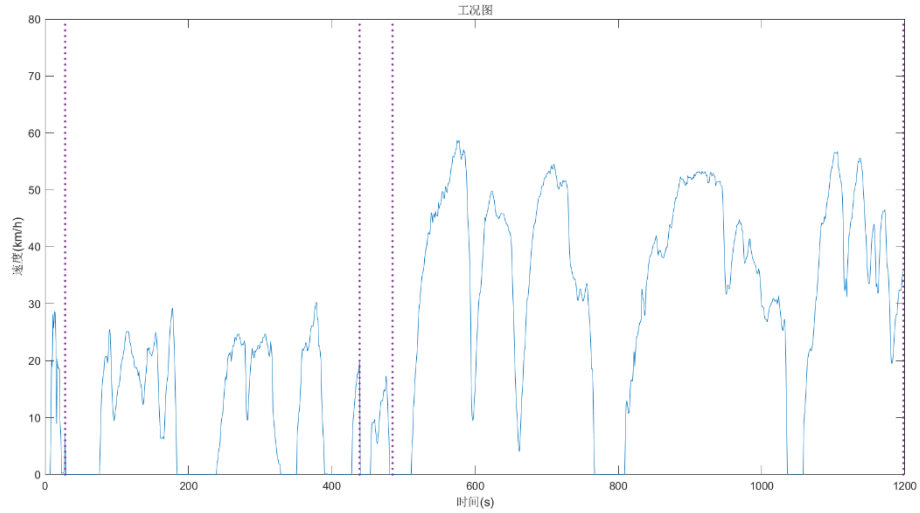


图 12 无对缺失值进行填补 Kmeans 补构建的工况

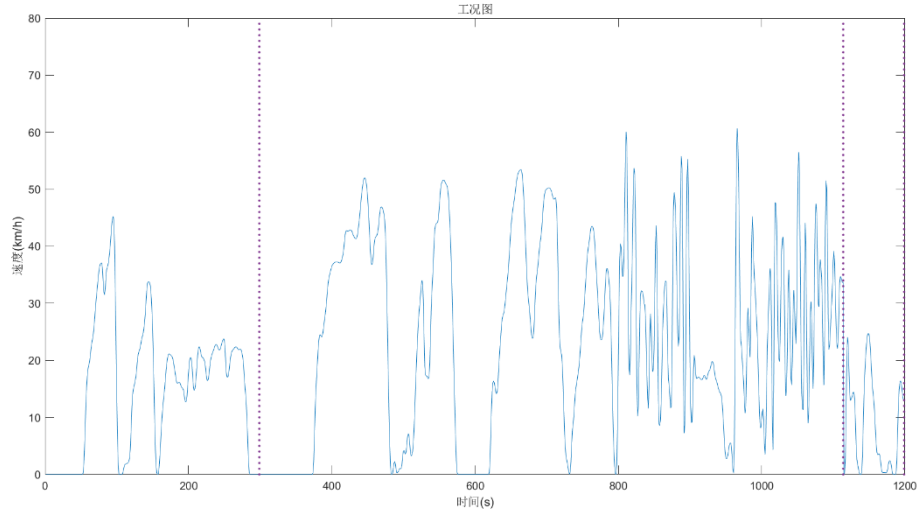


图 13 整体 PMM 填补 GMM 补构建的工况

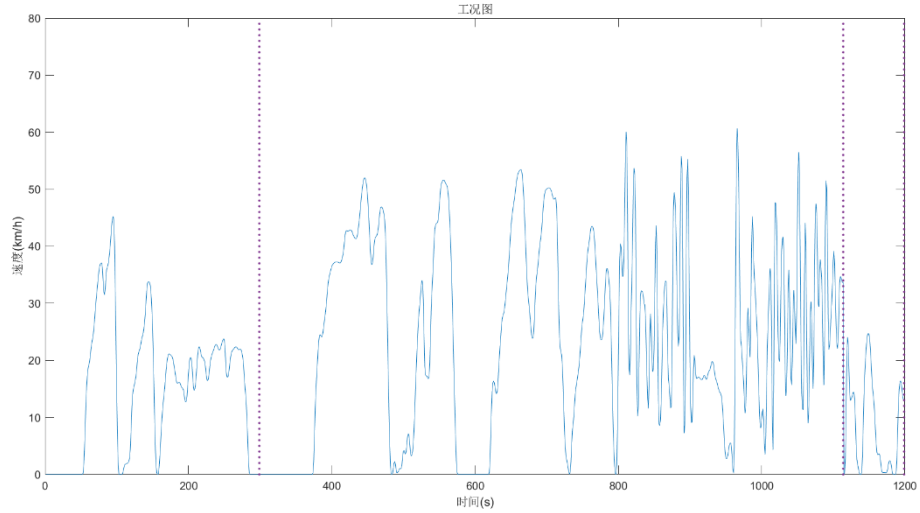


图 14 整体 PMM 填补 Kmeans 补构建的工况

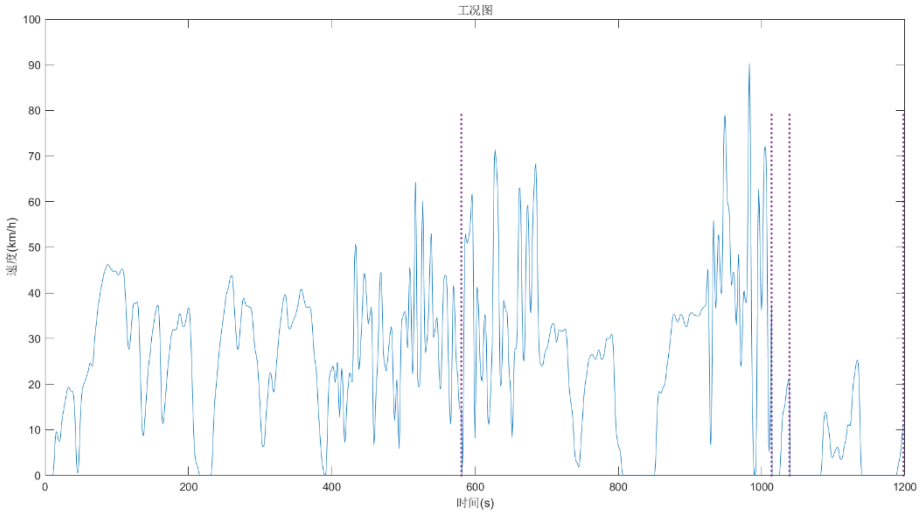


图 15 整体 PMM 对 0 不滤波 GMM 构建的工况

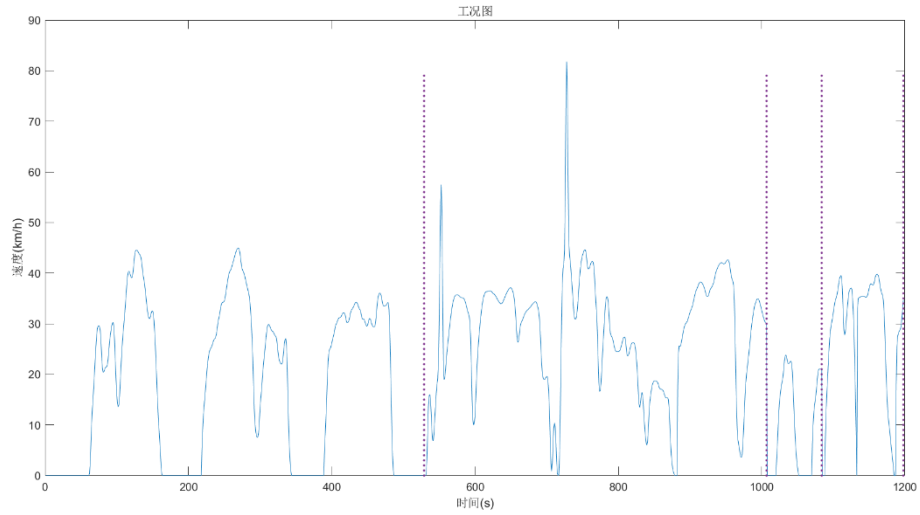


图 16 整体 PMM 对 0 不滤波 Kmeans 构建的工况

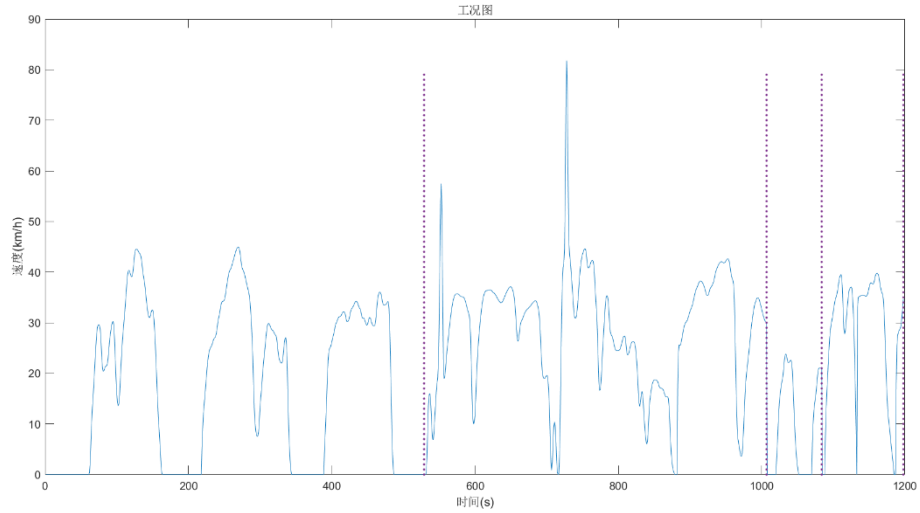


图 17 滑动窗口 PMM 填补 GMM 构建的工况

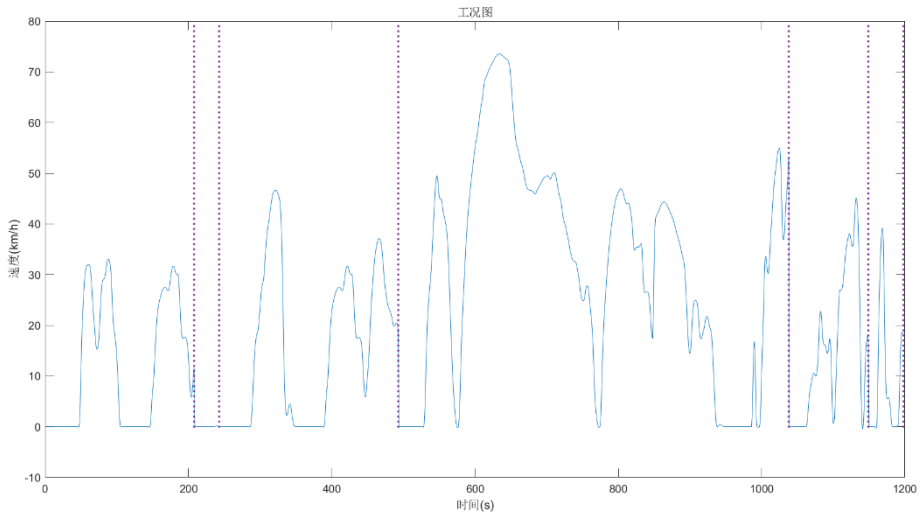


图 18 滑动窗口 PMM 填补 Kmeans 构建的工况

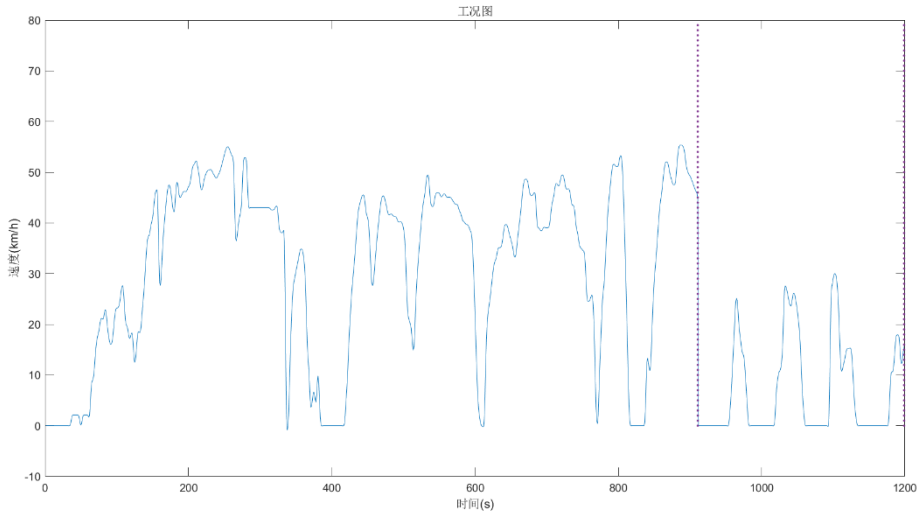
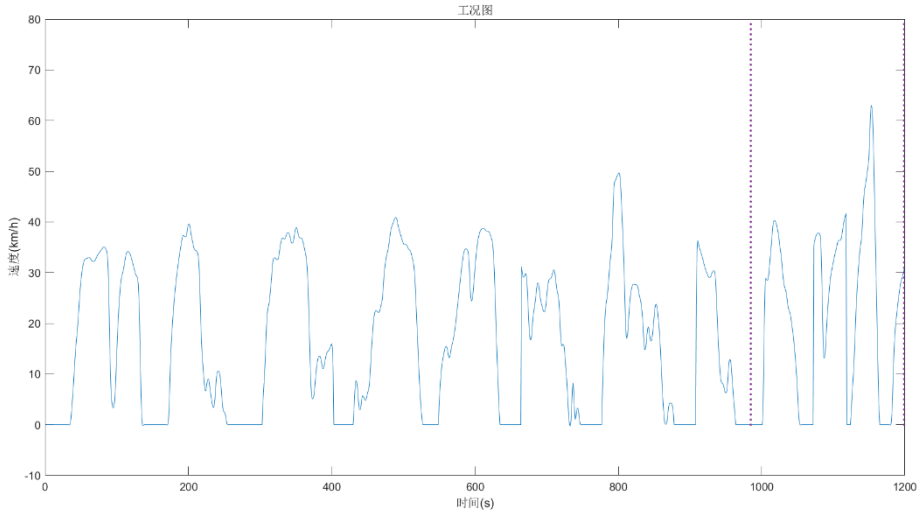


图 19 滑动窗口 PMM 对 0 不滤波 GMM 构建的工况



20 滑动窗口 PMM 对 0 不滤波 Kmeans 构建的工况

附录二

平均速度	平均行驶速度	怠速时间比	平均加速度	平均减速度	加速时间比	减速时间比	速度标准差	加速度标准差
29.020666097657	37.027082283535	0.194334613890448	1.443100116947	1.780538570339	0.279416842923123	0.227621185306686	27.036556379394	1.275474134613
17.285989300980	24.185351790402	0.270225187656380	1.804555816218	2.054935116525	0.294412010008340	0.246038365304420	15.252698054567	2.973652343167
24.459000000000	34.790271908363	0.273561301084237	2.204182942708	2.352631216989	0.260216847372811	0.236864053377815	22.442994634524	5.537025510595

附录三

1 FirsterProblem	第一个问题主程序
2 SecondProblem	第二个问题主程序
3 ThirdProblem	第三个问题主程序
4 Shang	熵权赋值法计算函数
5 Guiyi	归一化计算函数
6 computing_driver_cycle_index	计算工程指标函数
7 pmm	Pmm 计算程序

1 FirsterProblem

```
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
%%%%%%%   This code is for   16th mathmatical modeling and this function       %%
%%%%%%%   includes:                                                           %%
%%%%%%%         1、  read data from microsoft spredsheet and transform these   %%
%%%%%%%         files to table T1 T2 and T3                                   %%
%%%%%%%         2、identify   error piont                                      %%
%%%%%%%                   1）  missing value of time                          %%
%%%%%%%                   2）  speed which more than 120km/h a                %%
%%%%%%%                   3)   speedup which more than reasonable interval     %%
%%%%%%%                   4)   long park or idling                            %%
%%%%%%%                                                                        %
%
%%%%%%%
Email:1378917721@qq.com                                                  %%
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%

%%  %%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
%    set the value of pmm_window_smooth that represents which way of
%    process to choose
%    pmm_window_smooth=
%    1              pmm_window_smmoth
%    2              pmm_window_smmoth_without_zeros
%    3              pmm_smmoth
%    4                pmm_smooth_without_zeros
%    5              doesn't fill and doesn't smooth

%% %%%%%%%%%%%%%%%%%%%%%%%%%%%%%%% %%%%%%%%%%%%%%%%%%%%%
```

```
%%%%%%%%%
%      ----------------------- read file-------------------------------------
%      read three files which include the GPS data.In this case, the data is
%      named '文件 1.xlsx','文件 2.xlsx'and '文件 3.xlsx'.
%      input arguments:
%                              1.filename1:this value is '文件 1.xlsx' in this case
%                              2.filename2:this value is '文件 1.xlsx' in this case
%                              3.filename1:this value is '文件 1.xlsx' in this case
%      output arguments:
%                              1.T1:this is a table of type formalized form '文件 1.xlsx'
%                              2.T2:this is a table of type formalized form '文件 2.xlsx'
%                              3.T3:this is a table of type formalized form '文件 3.xlsx'
%  %%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
%%%%%%%%%%%%%%%%%%%%%
T1=readtable('文件 1.xlsx');
T2=readtable('文件 2.xlsx');
T3=readtable('文件 3.xlsx');
T=[T2;T3;T1];
T.Properties.VariableNames = ...
       {'time',                                                                                                'V'
'X_speedup','Y_speedup','Z_speedup','longitude','latitude','V_engine','percentage_torque'...
       'instant_fuel_consuption' 'opening_of_accelerator_pedal' 'air_fuel_ratio' ...
       'Percentage_of_engine_load' 'Air_intake_flow'};

time=datetime(T.time,'InputFormat','yyyy/MM/dd HH:mm:ss''.000.');
T.time=time;
h1=height(T1);
h2=height(T2);
h3=height(T3);
T2=T(1:h2,:);
T3=T(h2+1:h3+h2,:);
T1=T(h3+h2+1:h1+h2+h3,:);
time1=T1.time;
b1=time1-time1(1);
timestamp1=seconds(b1);
T1.timestamp=timestamp1;

T1.longitude=[];
T1.latitude=[];
T1.X_speedup=[];
T1.Y_speedup=[];
T1.Z_speedup=[];
%T1.V_engine=[];
% T1.percentage_torque=[];
```

```
T1.instant_fuel_consuption=[];
% T1.opening_of_accelerator_pedal=[];
% T1.air_fuel_ratio=[];
% T1.Percentage_of_engine_load=[];
% T1.Air_intake_flow=[];

time2=T2.time;
b2=time2-time2(1);
timestamp2=seconds(b2);
T2.timestamp=timestamp2;

T2.longitude=[];
T2.latitude=[];
T2.X_speedup=[];
T2.Y_speedup=[];
T2.Z_speedup=[];
%T2.V_engine=[];
% T2.percentage_torque=[];
T2.instant_fuel_consuption=[];
% T2.opening_of_accelerator_pedal=[];
% T2.air_fuel_ratio=[];
% T2.Percentage_of_engine_load=[];
% T2.Air_intake_flow=[];

time3=T3.time;
b3=time3-time3(1);
timestamp3=seconds(b3);
T3.timestamp=timestamp3;

T3.longitude=[];
T3.latitude=[];
T3.X_speedup=[];
T3.Y_speedup=[];
T3.Z_speedup=[];
%T3.V_engine=[];
% T3.percentage_torque=[];
T3.instant_fuel_consuption=[];
% T3.opening_of_accelerator_pedal=[];
% T3.air_fuel_ratio=[];
% T3.Percentage_of_engine_load=[];
% T3.Air_intake_flow=[];
T=[T2;T3;T1];
```

```
save original_data T1 T2 T3

%%  %%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
%%%%%%%%%%%
%   ------------- plot images of longitude and latitude -------------------
%   this step aim to plot dynamic images of car driveing based on logitude
%   and latitude which form '文件 1.xlsx','文件 2.xlsx'and '文件 3.xlsx'.
%   input arguments:
%                         1.T1:this is produced by last step
%                         2.T2:this is produced by last step
%                         3.T3:this is produced by last step
%    output arguments:
%                         a matlab dynamic figure
%    %%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
%%%%%%%%%%%%%%%%%%%%%%
%    spped time image of 13
%    index=find(hour(T1.time)==13);
%    x=T1.timestamp(index);
%    y=T1.V(index);
%    plot(x(1:500),y(1:500));

%    画动态轨迹图
%    index2=find(day(T1.time)==19);
%    scatter(T1.longitude(index2),T1.latitude(index2),1,'k')
%    x=T1.longitude(index2);
%    y=T1.latitude(index2);
%    h = animatedline('MaximumNumPoints',length(x));
%    for k = 1:length(x)
%          addpoints(h,x(k),y(k));
%            drawnow
%    end

%%  %%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
%%%%%%%%%%%%
%    -----identify speed and speedup exception------------------------------
%    speed any point shouldn't be more than 120km/h,this is well known
%    limited speed in china.speedup any point must lie in a reasonable
%    interval,this step is identify these error and delete them
%    input arguments:
%                         1.T1:this is produced by last step
%                         2.T2:this is produced by last step
%                         3.T3:this is produced by last step
%    output arguments:
%                         1.T1:T1 in which speed and speedup exception are
```

```
%                               deleted.
%                               2.T2:T2 in which speed and speedup exception are
%                               deleted.
%                               3.T3:T3 in which speed and speedup exception are
%                               deleted
%       %%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
%%%%%%%%%%%%%%%%%%%%%%%%%%%
%
%1 speed more than 120km/h
      t1=find(T1.V>120);
      T1(t1,:).timestamp(:)=-1;

        t2=find(T2.V>120);
      T2(t2,:).timestamp(:)=-1;

        t3=find(T3.V>120);
      T3(t3,:).timestamp(:)=-1;
      %statistic information
      disp('速度值异常值个数')
      disp([length(t1) length(t2) length(t3)])

% 2 speedup too big and only detect continue time point -1
% speedup -2 this a unknow speedup
%             -1 this a not satified speedup value
max_speedup=100/7;
max_reduce_speedup=-1*8/1000/(1/3600);
D=T1;
speedup=-2*ones(height(D),1);
for i=2:height(D)
      if D.timestamp(i)-D.timestamp(i-1)==1& D.V(i)<=120
            speedup(i)=D.V(i)-D.V(i-1);
            if speedup(i)>max_speedup |speedup(i)<max_reduce_speedup
                  speedup(i)=-1;
            end
      end
end
D.speedup=speedup;
t1=find(D.speedup==-1);
D(t1,:).timestamp(:)=-1; %mark missing value
T1=D;



D=T2;
speedup=-2*ones(height(D),1);
```

```
for i=2:height(D)
	if D.timestamp(i)-D.timestamp(i-1)==1
		speedup(i)=D.V(i)-D.V(i-1);
		if speedup(i)>max_speedup |speedup(i)<max_reduce_speedup
			speedup(i)=-1;
		end
	end
end
D.speedup=speedup;
t2=find(D.speedup==-1);
D(t2,:).timestamp(:)=-1; %mark missing value
T2=D;

D=T3;
speedup=-2*ones(height(D),1);
for i=2:height(D)
	if D.timestamp(i)-D.timestamp(i-1)==1
		speedup(i)=D.V(i)-D.V(i-1);
		if speedup(i)>max_speedup |speedup(i)<max_reduce_speedup
			speedup(i)=-1;
		end
	end
end
D.speedup=speedup;
t3=find(D.speedup==-1);
D(t3,:).timestamp(:)=-1; %mark missing value
T3=D;
  %statistic information
	disp('加速度值异常值个数')
	disp([length(t1) length(t2) length(t3)])
%%  %%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
%%%%%%%%%%%%
%-------------------find missing value of time-----------------------------
%      find missing value of time in T1,T2 and T3, and add a position in T1,T2
%      and T3 if identify them.
%      input arguments:
%                               1.T1:there is doesn't exsits speed and speedup exception
%                               2.T2:there is doesn't exsits speed and speedup exception
%                               3.T3:there is doesn't exsits speed and speedup exception
%      output arguments:
%                               1.T1. in which exsits missing position mark
%                               2.T2. in which exsits missing position mark
%                               3.T3. in which exsits missing position mark
%    %%%%%%%%%%%%%%%%%%%%%%%%%%%%a%%%%%%%%%%%%%%%%%%%%%%%
```

```
%%%%%%%%%%%%%%%%%%%%
t=[];
tt=[];
Data=[];
for j=18:24
    D=T1(day(T1.time)==j,:);
    N=height(D);
    t=[];
    for i=N-1:-1:1
        if D.timestamp(i)==-1
            continue;
        end
        n=D.timestamp(i+1)-D.timestamp(i);
        if n>1
            t=[t n-1];
            signal_not_GPS=T1(1:n-1,:);
              signal_not_GPS.V_engine(:)=nan;
              signal_not_GPS.percentage_torque(:)=nan;
                signal_not_GPS.opening_of_accelerator_pedal(:)=nan;
                signal_not_GPS.air_fuel_ratio(:)=nan;
                signal_not_GPS.Percentage_of_engine_load(:)=nan;
              signal_not_GPS.Air_intake_flow(:)=nan;
            signal_not_GPS.V(:)=-1;
            signal_not_GPS.timestamp(:)=-1;
            D=[D(1:i,:);signal_not_GPS;D(i+1:end,:)];
        end
    end
    tt=[tt sum(t)];
    Data=[Data;D];
end
D1=Data;
disp('表一不连续的个数为')
disp(sum(tt))
t=[];
tt=[];
Data=[];
for j=1:7
    D=T2(day(T2.time)==j,:);
    N=height(D);
    t=[];
    for i=N-1:-1:1
          if D.timestamp(i)==-1
            continue;
           end
```

```
        n=D.timestamp(i+1)-D.timestamp(i);
        if n>1
            t=[t n-1];
            signal_not_GPS=T2(1:n-1,:);
              signal_not_GPS.V_engine(:)=nan;
              signal_not_GPS.percentage_torque(:)=nan;
                signal_not_GPS.opening_of_accelerator_pedal(:)=nan;
                signal_not_GPS.air_fuel_ratio(:)=nan;
                signal_not_GPS.Percentage_of_engine_load(:)=nan;
              signal_not_GPS.Air_intake_flow(:)=nan;
            signal_not_GPS.V(:)=-1;
            signal_not_GPS.timestamp(:)=-1;
            D=[D(1:i,:);signal_not_GPS;D(i+1:end,:)];
        end
    end
    tt=[tt sum(t)];
    Data=[Data;D];
end
D2=Data;
disp('表二不连续的个数为')
disp(sum(tt))

t=[];
tt=[];
Data=[];
for j=1:6
    D=T3(day(T3.time)==j,:);
    N=height(D);
    t=[];
    for i=N-1:-1:1
       if D.timestamp(i)==-1
            continue;
        end
        n=D.timestamp(i+1)-D.timestamp(i);
        if n>1
            t=[t n-1];
            signal_not_GPS=T3(1:n-1,:);
              signal_not_GPS.V_engine(:)=nan;
              signal_not_GPS.percentage_torque(:)=nan;
                signal_not_GPS.opening_of_accelerator_pedal(:)=nan;
                signal_not_GPS.air_fuel_ratio(:)=nan;
                signal_not_GPS.Percentage_of_engine_load(:)=nan;
              signal_not_GPS.Air_intake_flow(:)=nan;
```

```
                signal_not_GPS.V(:)=-1;
                signal_not_GPS.timestamp(:)=-1;
                D=[D(1:i,:);signal_not_GPS;D(i+1:end,:)];
            end
        end
        tt=[tt sum(t)];
        Data=[Data;D];
end
D3=Data;
disp('表三不连续的个数为')
disp(sum(tt))
T1=D1;
T2=D2;
T3=D3;

%%  %%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
%%%%%%%%%%%%
%---------------delete continuous time of zero more than 180 --------------
%      long park,idling which is more than 180s. we deleted the excess part
%      input argumets:
%                           1.T1. in which exsits missing position mark
%                           2.T2. in which exsits missing position mark
%                           3.T3. in which exsits missing position mark
%      output arguments:                                                        ------
%                           1.T1:in which doesn't exsit ontinuous time of time more
%                           than 180s
%                           2.T2:in which doesn't exsit ontinuous time of time more
%                           than 180s
%                           3.T3:in which doesn't exsit ontinuous time of time more
%                           than 180s
%  %%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
%%%%%%%%%%%%%%%%%%%%%
deleted_rows_T1=0;
t=0;
N=height(T1);
V=T1.V;
d=[];
for i=1:N
        if T1.timestamp(i)==-1
            t=0;
            continue;
        end
        if V(i)<10
            t=t+1;
```

```
    else
        t=0;
    end
    if t>180
       d=[d i];
    end

end
T1(d,:)=[];
deleted_rows_T1=length(d);
%delete continuous time more than 180 for T2
t=0;
N=height(T2);
V=T2.V;
d=[];
for i=1:N
    if T2.timestamp(i)==-1
        t=0;
        continue;
    end
    if V(i)<10
        t=t+1;
    else
        t=0;
    end
    if t>180
       d=[d i];
    end

end
T2(d,:)=[];
deleted_rows_T2=length(d);
%delete continuous time more than 180 for T3
t=0;
N=height(T3);
V=T3.V;
d=[];
for i=1:N
    if T3.timestamp(i)==-1
        t=0;
        continue;
    end
    if V(i)<10
        t=t+1;
```

```
        else
              t=0;
        end
        if t>180
           d=[d i];
        end

end
T3(d,:)=[];
deleted_rows_T3=length(d);
disp('对于第一问的 3、4、5，三个文件删除的数据行分别为')
disp([deleted_rows_T1,deleted_rows_T2,deleted_rows_T3])




%%  %%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
%%%%%%%%%%%%
%-------------delete missing parts which accounts for some value-----------
%      For some missing parts identified by last step, it accounts for too big
%      parts of all parts, it is not reasonable to fill this parts,so we
%      should delete based on given threshold value
%      input arguments:
%                               1.T1:in which doesn't exsit ontinuous time of time more
%                              than 180s
%                               2.T2:in which doesn't exsit ontinuous time of time more
%                              than 180s
%                               3.T3:in which doesn't exsit ontinuous time of time more
%                              than 180s
%                               4. missing_ratio: the threshold value
%      output arguments:
%                               1 D1:processed T1
%                               1 D2:processed T2
%                               1 D3:processed T3
%       %%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
%%%%%%%%%%%%%%%%%%%%%%
missing_ratio=0.2;
D=T1;
N=height(D);
s=struct();
s.starti=0;
s.endi=0;
S.N=0;
i_s=0;
```

```
if D.timestamp(1)==-1
      i_s=i_s+1;
      s(i_s).starti=1;
      s(i_s).endi=1;
      s(i_s).N=1;
end
for i=2:N
      if D.timestamp(i)==-1
            if D.timestamp(i-1)~=-1
                  i_s=i_s+1;
                  s(i_s).starti=i;
                  s(i_s).endi=i;
                  s(i_s).N=1;
            else
                  s(i_s).endi=s(i_s).endi+1;
                  s(i_s).N=s(i_s).N+1;
                  if s(i_s).endi>i
                        p=1;
                  end
            end
      end

end
   S1=s;

  D=T2;
N=height(D);
s=struct();
s.starti=0;
s.endi=0;
S.N=0;
i_s=0;

if D.timestamp(1)==-1
      i_s=i_s+1;
      s(i_s).starti=1;
      s(i_s).endi=1;
      s(i_s).N=1;
end
for i=2:N
      if D.timestamp(i)==-1
            if D.timestamp(i-1)~=-1
                  i_s=i_s+1;
```

```
                s(i_s).starti=i;
                s(i_s).endi=i;
                s(i_s).N=1;
            else
                s(i_s).endi=s(i_s).endi+1;
                s(i_s).N=s(i_s).N+1;
            end
        end

end
S2=s;

D=T3;
N=height(D);
s=struct();
s.starti=0;
s.endi=0;
S.N=0;
i_s=0;

if D.timestamp(1)==-1
        i_s=i_s+1;
        s(i_s).starti=1;
        s(i_s).endi=1;
        s(i_s).N=1;
end
for i=2:N
        if D.timestamp(i)==-1
            if D.timestamp(i-1)~=-1
                i_s=i_s+1;
                s(i_s).starti=i;
                s(i_s).endi=i;
                s(i_s).N=1;
            else
                s(i_s).endi=s(i_s).endi+1;
                s(i_s).N=s(i_s).N+1;
            end
        end

end

S3=s;
missing_1=sortrows(struct2table(S1),'N','descend');
missing_2=sortrows(struct2table(S2),'N','descend');
```

```
missing_3=sortrows(struct2table(S3),'N','descend');

missing_1.ratio=missing_1.N./height(T1);
missing_ratio_1=sum(missing_1.N)/height(T1);

missing_2.ratio=missing_2.N./height(T2);
missing_ratio_2=sum(missing_2.N)/height(T2);

missing_3.ratio=missing_3.N./height(T3);
missing_ratio_3=sum(missing_3.N)/height(T3);
%missing value position
h=figure(1);
set(gcf,'outerposition',get(0,'screensize'));
set(h,'visible','off');
for i=1:height(missing_1)

plot(missing_1.starti(i)/3600:missing_1.endi(i)/3600,zeros(size(missing_1.starti(i)/3600:missin
g_1.endi(i)/3600)),'Color',[1 0 0],'linewidth',10)
    hold on
end

xlabel('时间/h');
yticks([])

saveas(h,'../图/表 1-缺失时间分布','jpg');
delete(h)
h=figure(2);
set(gcf,'outerposition',get(0,'screensize'));
set(h,'visible','off');
for i=1:height(missing_2)

plot(missing_2.starti(i)/3600:missing_2.endi(i)/3600,zeros(size(missing_2.starti(i)/3600:missin
g_2.endi(i)/3600)),'Color',[1 0 0],'linewidth',10)
    hold on
end


xlabel('时间/h');
yticks([])

saveas(h,'../图/表 2-缺失时间分布','jpg');
delete(h)

h=figure(3);
```

```
set(gcf,'outerposition',get(0,'screensize'));
set(h,'visible','off');
for i=1:height(missing_3)

plot(missing_3.starti(i)/3600:missing_3.endi(i)/3600,zeros(size(missing_3.starti(i)/3600:missin
g_3.endi(i)/3600)),'Color',[1 0 0],'linewidth',10)
    hold on
end

xlabel('时间/h');
yticks([])

saveas(h,'../图/表 3-缺失时间分布','jpg');
delete(h)

need_deleted_1=[];
need_deleted_2=[];
need_deleted_3=[];
N=height(T1);
for i=1:height(missing_1)
    x=sum(missing_1.N(1:i));
    z=sum(missing_1.N)-x;

    if z/(N-x)<missing_ratio
        need_deleted_1=missing_1(1:i,:);
        break;
    end
end
N=height(T2);
for i=1:height(missing_2)
     x=sum(missing_2.N(1:i));
    z=sum(missing_2.N)-x;

    if z/(N-x)<missing_ratio
        need_deleted_2=missing_2(1:i,:);
        break;
    end
end
N=height(T3);
for i=1:height(missing_3)
     x=sum(missing_3.N(1:i));
    z=sum(missing_3.N)-x;

    if z/(N-x)<missing_ratio
```

```
            need_deleted_3=missing_3(1:i,:);
            break;
        end
end


tt=[];
for i=1:height(need_deleted_1)
        t=need_deleted_1.starti(i):need_deleted_1.endi(i);
        tt=[tt t];
end
T1(tt,:)=[];

tt=[];
for i=1:height(need_deleted_2)
        t=need_deleted_2.starti(i):need_deleted_2.endi(i);
        tt=[tt t];
end
T2(tt,:)=[];
tt=[];
for i=1:height(need_deleted_3)
        t=need_deleted_3.starti(i):need_deleted_3.endi(i);
        tt=[tt t];
end
T3(tt,:)=[];

T1(find(T1.timestamp==-1),:).V(:)=nan;
T2(find(T2.timestamp==-1),:).V(:)=nan;
T3(find(T3.timestamp==-1),:).V(:)=nan;
% export data for R to fill missing value
D1=[T1.V          T1.V_engine          T1.percentage_torque          T1.Percentage_of_engine_load
T1.opening_of_accelerator_pedal T1.air_fuel_ratio T1.Air_intake_flow];
D2=[T2.V          T2.V_engine          T2.percentage_torque          T2.Percentage_of_engine_load
T2.opening_of_accelerator_pedal T2.air_fuel_ratio T2.Air_intake_flow];
D3=[T3.V          T3.V_engine          T3.percentage_torque          T3.Percentage_of_engine_load
T3.opening_of_accelerator_pedal T3.air_fuel_ratio T3.Air_intake_flow];
save RData D1 D2 D3
%% doesn't fill missing value
V_not_fill=[T2.V;T3.V;T1.V];
V_not_fill(isnan(V_not_fill))=[];
save V_not_fill V_not_fill;

%% %%%%%%%%%%%%%%%%%%%%please use R language script pmm.R to    fill
%                                    missing          value          in          FData
```

```
and %%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%  %%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
%%%%%%%%%%%%%%%%%

%% detect speed which more than 120
%% detct speedup
% max_speedup=100/7; %14.2857
% max_reduce_speedup=-1*8/1000/(1/3600); % -28.8000
%
% detected_speedup=[]
%
% for i=2:length(V)
%
%           speedup=V(i)-V(i-1);
%           if speedup>max_speedup |speedup<max_reduce_speedup
%               detected_speedup=[detected_speedup i];
%           end
%
% end
% V(detected_speedup)=[];
%% decided if to use pmm window smmoth
%pmm_window_smooth=
%1           pmm_window_smmoth
%2           pmm_window_smmoth_without_zeros
%3           pmm_smmoth
%4             pmm_smooth_without_zeros
%5           doesn't fill and doesn't smooth
pmm_window_smooth=1;
if pmm_window_smooth==1    %pmm_window_smmoth
    load V_windows_pmm_smooth
    load V_windows_pmm;

     plot(V1_windows_pmm(1:500),'Color',[0,0,1]);
     hold on
     plot(V1_windows_pmm_smooth(1:500),'Color',[1,0,0]);
     hold off
     xlabel('时间(s)');
     ylabel('速度(km/h)');
     legend('源数据','平滑后数据');
     saveas(gca,'../图/平滑后数据对比图','bmp');
     V=V_windows_pmm_smooth;
     V1=V1_windows_pmm_smooth;
     V2=V2_windows_pmm_smooth;
     V3=V3_windows_pmm_smooth;
```

```
end
if pmm_window_smooth==2    %pmm_window_smmoth_without_zeros
      load V_windows_pmm_smooth
      load V_windows_pmm;
      V_windows_pmm_smooth(V_windows_pmm==0)=0;
      V1_windows_pmm_smooth(V1_windows_pmm==0)=0;
      V2_windows_pmm_smooth(V2_windows_pmm==0)=0;
      V3_windows_pmm_smooth(V3_windows_pmm==0)=0;

      plot(V1_windows_pmm(1:500),'Color',[0,0,1]);
      hold on
      plot(V1_windows_pmm_smooth(1:500),'Color',[1,0,0]);
      hold off
      xlabel('时间(s)');
      ylabel('速度(km/h)');
      legend('源数据','平滑后数据');
      saveas(gca,'../图/平滑后数据对比图','bmp');
      V=V_windows_pmm_smooth;
      V1=V1_windows_pmm_smooth;
      V2=V2_windows_pmm_smooth;
      V3=V3_windows_pmm_smooth;

end
if pmm_window_smooth==3    %    pmm_smmoth
      load V_pmm_smooth
      load V_pmm
      plot(V1_pmm(1:500),'Color',[0,0,1]);
      hold on
      plot(V1_pmm_smooth(1:500),'Color',[1,0,0]);
      hold off
      xlabel('时间(s)');
      ylabel('速度(km/h)');
      legend('源数据','平滑后数据');
      saveas(gca,'../图/平滑后数据对比图','bmp');
      V=V_pmm_smooth;
      V1=V1_pmm_smooth;
      V2=V2_pmm_smooth;
      V3=V3_pmm_smooth;
end
if pmm_window_smooth==4    %    pmm_smooth_without_zeros
      load V_pmm_smooth
      load V_pmm
      V_pmm_smooth(V_pmm==0)=0;
```

```
    V1_pmm_smooth(V1_pmm==0)=0;
    V2_pmm_smooth(V2_pmm==0)=0;
    V3_pmm_smooth(V3_pmm==0)=0;

    plot(V1_pmm(1:500),'Color',[0,0,1]);
    hold on
    plot(V1_pmm_smooth(1:500),'Color',[1,0,0]);
    hold off
    xlabel('时间(s)');
    ylabel('速度(km/h)');
    legend('源数据','平滑后数据');
    saveas(gca,'../图/平滑后数据对比图','bmp');
    V=V_pmm_smooth;
    V1=V1_pmm_smooth;
    V2=V2_pmm_smooth;
    V3=V3_pmm_smooth;

end
if   pmm_window_smooth==5                    %5 oesn't fill and doesn't smooth
    load V_not_fill;
    V=V_not_fill;
end
%%%%%%%%%%%%%%%%%%%%%%%%%%%%loading                    SmoothV
data %%%%%%%%%%%%%%%%%%%%%%%%%%
%V(find(V>120))=[];
save V V
disp('firstProblem is OK,the result in SmoothV.mat and in../图片');
```

2 secondeProblem

```
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
%%%%%%%%%%%%%%%%%%%%%%%%%%%
%%%%%%   This code is for   16th mathmatical modeling and this function       %%
%%%%%%   includes:                                                            %%
%%%%%%        1. produce movement sequencess                                  %%
%%%%%%        2. identify some unreasonable movement sequencess adn delete %%
%%%%%%            these unreasonalbe movement sequencess includes:            %%
%%%%%%                  1）  last time is too short and less than 10s          %%
%%%%%%                  2）  speed which more than 120km/h a                  %%
%%%%%%                  3)   speedup which more than reasonable interval       %%
%%%%%%                  4)   driveing distance is less than 10m                %%
%%%%%%   input arguments:                                                     %%
```

```
%%%%%%                  1.T:the time threshold                                  %%
%%%%%%                  2.min_L:thie driving distance threshold                 %%
%%%%%%                  3.V:the results produced by firstProblem.m code         %%
%%%%%%   output arguments:                                                      %%
%%%%%%                                                  1.movementt:the       movement
sequence                        %%
%%%%%%
Email:1378917721@qq.com                                                        %%
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
%%%%%%%%%%%%%%%%%%%%%%

%load V
V=V3;
%% ----------construct movement sequeence----------
s=struct();
s.data=[];
i_s=0;
mark=0;
for i=2:length(V)
    if V(i)==0&V(i-1)~=0
        i_s=i_s+1;
        s(i_s).data=V(i);
        mark=1;
        continue;
    end
    if mark==1
       s(i_s).data=[s(i_s).data V(i)];
    end
end
S=struct2table(s);
for i=1:height(S)
    S.N(i)=length(S.data{i});
end


%% delete movement sequeence which longer than T%%%%%%%%%%%%%%%%%%
%argument:
%           T : the maximum time length
T=20;
S_T=S(find(S.N>T),:);

%% delete movement sequence which road length is mort short than L
%argument
%           min_L    the minimum short road(m）
```

```
min_L=10 ;
L=[];
for i=1:height(S_T)
    t=trapz(S_T.data{i}*1000/3600); % L is decribed by m
    L=[L t];
end
S_T_L=S_T;
S_T_L.L=L';
S_T_L=S_T_L(find(S_T_L.L>min_L),:);

%%  %%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
%%%%%%%%%%%%%%%%%%%%
%     ------delete movement sequence which speedup is not in speedup---------
%     interval [min_speedup max_speedup] and speed reduction interval
%     [min_speeddown max_speeddown]
%argmin
%           [min_speedup max_speedup]
%           [min_speeddown max_speeddown]
min_speedup=0.15/1000/(1/3600);     %0.5400    km/(h*s)
max_speedup=4/1000/(1/3600);        %    14.4000

min_speeddown=-4/1000/(1/3600); % -14.4000
max_speeddown=-0.15/1000/(1/3600);% -0.54

constant_speed=0.1/1000/(1/3600); %0.36
%cumpute the speedup
speedup=[];
p=0;
k=[];
for i=1:height(S_T_L)
    t=S_T_L.data{i};
    for j=2:length(t)
        p(j-1)=t(j)-t(j-1);
    end
    p=[p(1) p];

    speedup{i}=p;
end
S_T_L_S=S_T_L;
S_T_L_S.speedup=speedup';
ave_speedup=[];
ave_speeddown=[];
deleted=[];
for i=1:height(S_T_L_S)
```

```
    t=S_T_L_S.speedup{i};
    ave_speedup(i)=mean(t(find(t>constant_speed)));
    ave_speeddown(i)=mean(t(find(t<-1*constant_speed)));
    if ~isnan(ave_speedup(i))
      if ave_speedup(i)<min_speedup|ave_speedup(i)>max_speedup
              deleted=[deleted i];
      end
    end

    if ~isnan(ave_speeddown(i))
          if ave_speeddown(i)<min_speeddown | ave_speeddown(i)>max_speeddown
              deleted=[deleted i];
          end

    end


end
S_T_L_S(deleted,:)=[];
movement=S_T_L_S;
save movement movement
disp('Seconde Problem is OK,the result in movement.mat');
```

3 thirdproblem

```
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
%%%%%%    This code is for    16th mathmatical modeling and this function        %%
%%%%%%    includes:                                                              %%
%%%%%%          1. principle componets analysis                                  %%
%%%%%%          2. clustering:produce drive cycle                                %%
%%%%%%          3. formalize drive cycle                                         %%
%%%%%%          4. evaluate the drive cycle                                      %%
%%%%%%    input arguments:                                                       %%
%%%%%%                    1.movement:the movement sequence produced by S         %%
%%%%%%                                                              econdProblem.m
code                                        %%
%%%%%%                    3.V:the results produced by firstProblem.m code        %%
%%%%%%    output arguments:                                                      %%
%%%%%%                                                    1.movementt:the      movement
sequence                              %%
%%%%%%
Email:1378917721@qq.com                                                            %%
```

```
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
%%%%%%%%%%%%%%%%%%%%%
%%      %%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
%%%%%%%%%%%
%---------------------------Index computing--------------------------------
warning off
load movement
load V            %input V
constant_speed=0.1/1000/(1/3600); %0.36

avg_speed=[];
avg_drive_speed=[];
idling_time_ratio=[];
avg_speedup=[];
avg_speeddown=[];
speedup_time_ratio=[];
speeddown_time_ratio=[];
speed_std=[];
speedup_std=[];

%1 computing average speed

for i=1:height(movement)
    t=movement.data{i};
    speedup=movement.speedup{i};
    avg_speed=[avg_speed;mean(t)];
    avg_drive_speed=[avg_drive_speed;mean(t(find(t>0)))];
    idling=length(t(t==0));
    idling_time_ratio=[idling_time_ratio;idling/length(t)];
    speedup_collect=speedup(speedup>constant_speed);
    speeddown_collect=speedup(speedup<-1*constant_speed);
    avg_speedup=[avg_speedup;mean(speedup_collect)];
    avg_speeddown=[avg_speeddown;mean(speeddown_collect)];
    speedup_time_ratio=[speedup_time_ratio;length(speedup_collect)/length(t)];
    speeddown_time_ratio=[speeddown_time_ratio;length(speeddown_collect)/length(t)];
    speed_std=[speed_std;std(t)];
    speedup_std=[speedup_std;std(speedup_collect)];
end

movement.avg_speed=avg_speed;
movement.avg_drive_speed=avg_drive_speed;
movement.idling_time_ratio=idling_time_ratio;
movement.avg_speedup=avg_speedup;
movement.avg_speeddown=avg_speeddown;
```

```
movement.speedup_time_ratio=speedup_time_ratio;
movement.speeddown_time_ratio=speeddown_time_ratio;
movement.speed_std=speed_std;
movement.speedup_std=speedup_std;

%%  %%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
%%%%%%%%%%%%
%--------------------------------pca-------------------------------------
samples=table2array(movement(:,5:13));
samples_= zscore(samples);
[coeff,score,latent,tsquared,explained,mu]=pca(samples_);
%1 contribution table
contribution_table=table();
contribution_table.eigenvalue=latent;
contribution_table.contribution=explained;
contribution_table.Cum_contribution=cumsum(explained);

k=find(contribution_table.Cum_contribution>85);
num_pca=k(1);
%Gravel figure
h=figure(1);
set(h,'outerposition',get(0,'screensize'));
set(h,'visible','off');
plot(1:9,latent,'^');
hold on
plot(1:9,latent);
xlabel('主成分');
ylabel('特征值');
saveas(h,'../图/碎石图.bmp')
save('../图/pca,','contribution_table','score','coeff');
delete(h)
pca_sample=score(:,1:num_pca);
%%  %%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
%%%%%%%%%%%%
%-------------------------Clustering-------------------------------------
%      input arguments:
%                         1. cluster_Method: the cluster method,the 1 represents
%                         Gaussian mixture clustering, the 2 represents the
%                         kmeans clustering
%                         2.pca_sample: the data produced by pca
%    ouput arguments:
%                         1.idx:the cluster label each example in pca_sample
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
%%%%%%%%%%%%%%%%%%%%%%
```

```
cluster_Method=2;
% GMM clustering
if cluster_Method==1
    gmm_eva =
evalclusters(pca_sample,'gmdistribution','CalinskiHarabasz','KList',[1:6]);  %DaviesBouldin
silhouette CalinskiHarabasz
                                                                              %g
mdistribution
                                                                              %k
means
    gmm_num=gmm_eva.OptimalK;
    gm = fitgmdist(pca_sample,gmm_num);
    gmm_idx =cluster(gm ,pca_sample);
    gmm_c=gm.mu;
    disp(['GMM  聚类数为',num2str(gmm_num)]);
    disp(['CalinskiHarabasz 系数为',num2str(gmm_eva.CriterionValues(gmm_num))])
    cmap = hsv(gmm_num);
    h=figure(1);

    set(h,'outerposition',get(0,'screensize'));
    set(h,'visible','off');
    for i=1:gmm_num
        clusters=pca_sample(gmm_idx==i,:);
        scatter3(clusters(:,1),clusters(:,2),clusters(:,3),'.','MarkerEdgeColor',cmap(i,:));
        hold on
    end
    xlabel('第一主成分');
    ylabel('第二主成分');
    zlabel('第三主成分');
    saveas(h,'../图/高斯混合聚类图','bmp');
    delete(h);

%2 kmeans
elseif cluster_Method==2
    kmeans_eva =
evalclusters(pca_sample,'kmeans','CalinskiHarabasz','KList',[1:6]);  %DaviesBouldin  silhouette
CalinskiHarabasz
                                                                              %g
mdistribution
                                                                              %k
means
    kmeans_num=kmeans_eva.OptimalK;
    disp(['kmeans  聚类数为',num2str(kmeans_num)]);
    disp(['CalinskiHarabasz 系数为',num2str(kmeans_eva.CriterionValues(kmeans_num))])
```

```
        [kmeans_idx,kmeans_c] = kmeans(pca_sample,kmeans_num);
        cmap = hsv(kmeans_num);
        for i=1:kmeans_num
                clusters=pca_sample(kmeans_idx==i,:);
                scatter3(clusters(:,1),clusters(:,2),clusters(:,3),'.','MarkerEdgeColor',cmap(i,:));
                hold on
        end
        h=figure(1);
        set(h,'outerposition',get(0,'screensize'));
        set(h,'visible','off');
        xlabel('第一主成分');
        ylabel('第二主成分');
        zlabel('第三主成分');
        saveas(h,'../图/kmeans 聚类图','bmp');
        delete(h);
end
%3 hiearchy clustering
% T = clusterdata(pca_sample,'maxclust',cluster_number)
% Y = pdist(pca_sample);
% w=squareform(Y);
% Z = linkage(Y);
% dendrogram(Z,30)%view tree of 30 node

%evaluate
%plot
%%   %%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
%%%%%%%%%%%%
%----------extract movement sequences which near clustering centers--------
if cluster_Method==1
        cluster_number=gmm_num;
        cluster_c=gmm_c;
        idx=gmm_idx;
elseif cluster_Method==2
        cluster_number=kmeans_num;
        cluster_c=kmeans_c;
        idx=kmeans_idx;
end
%1 computing movement sequence length
driver_cycle_time=1200;
T_all=sum(movement.N);
T=[]; %the time for each kinde movement sequence
for i=1:cluster_number
        x=movement(idx==i,:);
        T(i)=sum(x.N)/T_all*driver_cycle_time;
```

```
end

%2 find
G=cell(cluster_number,1);
for i=1:cluster_number
      t=[];
      index=find(idx==i);
      y=pca_sample(index,:);
      [r ~]=size(y);
      center=repmat(cluster_c(i,:),r,1);
    [~,near_index] = sort(vecnorm(y-center,2,2));
      new_index=index(near_index(1));
      j=1;
      t=[t movement.data{new_index}];
      while movement.N(new_index)<T(i)
            j=j+1;
            new_index=index(near_index(j));
            t=[t movement.data{new_index}];
      end
      t=t(1:T(i));
      G{i}=t;
end
driver_cycle=[];
for i=1:cluster_number
      driver_cycle=[driver_cycle G{i}];
end
x=[];
y=[];
L=0;
for i=1:length(G)
      e=length(G{i});
      L=L+e;
      x=[x repmat(L,80)];
      y=[y 0:79];
end
h=figure(1);
set(h,'outerposition',get(0,'screensize'));
set(h,'visible','off');
driver_cycle_without_negative=driver_cycle;
driver_cycle_without_negative(driver_cycle_without_negative<0)=0;
plot(driver_cycle_without_negative)
hold on
plot(x,y,'.');
xlabel('时间(s)')
```

```
ylabel('速度(km/h)')
title('工况图');
saveas(gca,'../图/工况图','bmp')
delete(h)

%%  %%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
%----------------------------precision evaluation -------------------------
driver_cycle_index=computing_driver_cycle_index(driver_cycle);
driver_cycle_index=abs(driver_cycle_index);
original_index=computing_driver_cycle_index(V');      %input row vector
original_index=abs(original_index);
relative_error=abs(driver_cycle_index-original_index)./abs(original_index);
save('../图/results','driver_cycle','relative_error');
x=1:9;
set(gcf,'outerposition',get(0,'screensize'));
p1=bar(x,[driver_cycle_index' original_index']);
hold on
text(x(1)-0.3,-1,'平均速度');
text(x(2)-0.3,-1,'平均行驶速度');
text(x(3)-0.3,-1,'怠速时间比');
text(x(4)-0.3,-1,'平均加速度');
text(x(5)-0.3,-1,'平均减速度');
text(x(6)-0.3,-1,'加速时间比');
text(x(7)-0.3,-1,'减速时间比');
text(x(8)-0.3,-1,'速度标准差');
text(x(9)-0.3,-1,'加速度标准差');


for i=1:9

    text(i-0.5,driver_cycle_index(i)+0.7,num2str(driver_cycle_index(i)))
      text(i-0.1,original_index(i)+0.7,num2str(original_index(i)))
      if i==3
           text(i-0.2,relative_error(i)*100+5,num2str(relative_error(i)*100))
           continue;
      end

      text(i,relative_error(i)*100+1,num2str(relative_error(i)*100))

end
p2=plot(relative_error*100,'LineWidth',3);
plot(relative_error*100,'^','LineWidth',5)
legend([p1 p2],{'构建的工况指标','原数据指标','相对误差'},'location','northeast')
```

```
legend('boxoff')
xticks([]);
saveas(gca,'../图/精度评定图','bmp')

%%
disp('the third problem is OK,the data is store in driver_cycle.mat and images is in ../图片');
```

4 shang

```
function [s,w]=shang(x,ind)
%实现用熵值法求各指标(列）的权重及各数据行的得分
%x 为原始数据矩阵, 一行代表一个样本, 每列对应一个指标
%ind 指示向量，指示各列正向指标还是负向指标，1 表示正向指标，2 表示负向指标
%s 返回各行（样本）得分，w 返回各列权重
[n,m]=size(x); % n 个样本, m 个指标
%%数据的归一化处理
for i=1:m
    if ind(i)==1 %正向指标归一化
        X(:,i)=guiyi(x(:,i),1,0.002,0.996);      %若归一化到[0,1], 0 会出问题
    else %负向指标归一化
        X(:,i)=guiyi(x(:,i),2,0.002,0.996);
    end
end
%%计算第 j 个指标下，第 i 个样本占该指标的比重 p(i,j)
for i=1:n
    for j=1:m
        p(i,j)=X(i,j)/sum(X(:,j));
    end
end
%%计算第 j 个指标的熵值 e(j)
k=1/log(n);
for j=1:m
    e(j)=-k*sum(p(:,j).*log(p(:,j)));
end
d=ones(1,m)-e; %计算信息熵冗余度
w=d./sum(d); %求权值 w
s=100*w*X'; %求综合得分
```

5 guiyi

```
function y=guiyi(x,type,ymin,ymax)
%实现正向或负向指标归一化，返回归一化后的数据矩阵
%x 为原始数据矩阵, 一行代表一个样本, 每列对应一个指标
```

```
%type 设定正向指标 1,负向指标 2
%ymin,ymax 为归一化的区间端点
[n,m]=size(x);
y=zeros(n,m);
xmin=min(x);
xmax=max(x);
switch type
    case 1
        for j=1:m
            y(:,j)=(ymax-ymin)*(x(:,j)-xmin(j))/(xmax(j)-xmin(j))+ymin;
        end
    case 2
        for j=1:m
            y(:,j)=(ymax-ymin)*(xmax(j)-x(:,j))/(xmax(j)-xmin(j))+ymin;
        end
end
```

6 computing_driver_cycle_index

```
function driver_cycle_index=computing_driver_cycle_index(driver_cycle)

%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
%   input arguments:
%               driver_cycle :the driver cycle
%   output arguments:
%               driver_cycle_index: the index value of driver cycle
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
constant_speed=0.1/1000/(1/3600); %0.36

avg_speed=[];
avg_drive_speed=[];
idling_time_ratio=[];
avg_speedup=[];
avg_speeddown=[];
speedup_time_ratio=[];
speeddown_time_ratio=[];
speed_std=[];
speedup_std=[];

speedup=diff(driver_cycle);
speedup=[speedup(1) speedup];
t=driver_cycle;



avg_speed=[avg_speed;mean(t)];
```

```
avg_drive_speed=[avg_drive_speed;mean(t(find(t>0)))];
idling=length(t(t==0));
idling_time_ratio=[idling_time_ratio;idling/length(t)];
speedup_collect=speedup(speedup>constant_speed);
speeddown_collect=speedup(speedup<-1*constant_speed);
avg_speedup=[avg_speedup;mean(speedup_collect)];
avg_speeddown=[avg_speeddown;mean(speeddown_collect)];
speedup_time_ratio=[speedup_time_ratio;length(speedup_collect)/length(t)];
speeddown_time_ratio=[speeddown_time_ratio;length(speeddown_collect)/length(t
)];
speed_std=[speed_std;std(t)];
speedup_std=[speedup_std;std(speedup_collect)];

driver_cycle_index=[avg_speed...
    avg_drive_speed...
idling_time_ratio...
avg_speedup...
avg_speeddown...
speedup_time_ratio...
speeddown_time_ratio...
speed_std...
speedup_std];
```

7 Pmm

```
rm(list=ls())
cat(rep("\n", 50))
library(mice)
library('R.matlab')
N <- 30#ROLLING LENGTH

pathname<-file.path('C:\\Users\\Administrator\\Desktop\\newRData.mat')
data11<-readMat(pathname)
#View(data[[1]])

################################################
# Q1
################################################
data[[1]] <- as.data.frame(data[[1]])
d1 <- !(is.nan(data[[1]][,1]))
#choose proper independent

dd1 <- data[[1]][d1,]
tlm<-lm(dd1[,1]~dd1[,2]+dd1[,3]+dd1[,4]+
              dd1[,5]+dd1[,6]+dd1[,7],data=dd1)
```

```
summary(tlm)
## pmm filling rolling
n1 <- nrow(data[[1]])
ddd1 <- which(d1==FALSE)
for(i in ddd1){
   if(i<=N/2 & sum(is.nan(data[[1]][1:N+1,1]))/(N+1) < 0.2){
      imp_single <- mice(data[[1]][1:(N+1),], m = 2, method = "pmm") # Impute
missing values
      D1 <- complete(imp_single)
      data[[1]][1:(N+1),] <- D1
      }else if(i>=n1-N/2+1 & sum(is.nan(data[[1]][n1-(N+1):n1,1]))/(N+1) < 0.2){
      imp_single <- mice(data[[1]][n1-(N+1):n1,], m = 2, method = "pmm") # Impute
missing values
      D1 <- complete(imp_single)
      data[[1]][n1-(N+1):n1,] <- D1
   }else{
   if (i>N/2 & i<n1-N/2+1 & sum(is.nan(data[[1]][(i-N/2):(i+N/2),1]))/(N+1) < 0.2){
imp_single <- mice(data[[1]][(i-N/2):(i+N/2),], m = 2, method = "pmm") # Impute
missing values
D1 <- complete(imp_single)
data[[1]][(i-N/2):(i+N/2),] <- D1
   }
   }
d1 <- !(is.nan(data[[1]][,1]))
ddd1 <- which(d1==FALSE)
}
write.csv(data[[1]],"C:\\Users\\Administrator\\Desktop\\data[[1]].csv",fileEncoding
= "GBK")
#AAA <- read.table("C:\\Users\\Administrator\\Desktop\\data[[1]].csv",header =
T,sep = ",")
#F1 <- na.omit(AAA)

################################################
# Q2
################################################
data[[2]] <- as.data.frame(data[[2]])
d1 <- !(is.nan(data[[2]][,1]))
#choose proper independent

dd1 <- data[[2]][d1,]
tlm<-lm(dd1[,1]~dd1[,2]+dd1[,3]+dd1[,4]+
                dd1[,5]+dd1[,6]+dd1[,7],data=dd1)
summary(tlm)
## pmm filling rolling
```

```
n1 <- nrow(data[[2]])
ddd1 <- which(d1==FALSE)
data[[2]] <- data[[2]][,-7]
for(i in ddd1){
  if(i<=N/2 & sum(is.nan(data[[2]][1:N+1,1]))/(N+1) < 0.2){
    imp_single <- mice(data[[2]][1:(N+1),], m = 2, method = "pmm") # Impute
missing values
    D1 <- complete(imp_single)
    data[[2]][1:(N+1),] <- D1
  }else if(i>=n1-N/2+1 & sum(is.nan(data[[2]][n1-(N+1):n1,1]))/(N+1) < 0.2){
    imp_single <- mice(data[[2]][n1-(N+1):n1,], m = 2, method = "pmm") # Impute
missing values
    D1 <- complete(imp_single)
    data[[2]][n1-(N+1):n1,] <- D1
  }else{
    if (i>N/2 & i<n1-N/2+1 & sum(is.nan(data[[2]][(i-N/2):(i+N/2),1]))/(N+1) < 0.2){
      imp_single <- mice(data[[2]][(i-N/2):(i+N/2),], m = 2, method = "pmm") #
Impute missing values
      D1 <- complete(imp_single)
      data[[2]][(i-N/2):(i+N/2),] <- D1
    }
  }
  d1 <- !(is.nan(data[[2]][,1]))
  ddd1 <- which(d1==FALSE)
}
write.csv(data[[2]],"C:\\Users\\Administrator\\Desktop\\data[[2]].csv",fileEncoding
= "GBK")

#F1 <- na.omit(data[[1]])
################################################
# Q3
################################################
data[[3]] <- as.data.frame(data[[3]])
d1 <- !(is.nan(data[[3]][,1]))
#choose proper independent

dd1 <- data[[3]][d1,]
tlm<-lm(dd1[,1]~dd1[,2]+dd1[,3]+dd1[,4]+
          dd1[,5]+dd1[,6]+dd1[,7],data=dd1)
summary(tlm)
## pmm filling rolling
n1 <- nrow(data[[3]])
ddd1 <- which(d1==FALSE)
for(i in ddd1){
```

```
   if(i<=N/2 & sum(is.nan(data[[3]][1:N+1,1]))/(N+1) < 0.2){
      imp_single <- mice(data[[3]][1:(N+1),], m = 2, method = "pmm") # Impute
missing values
      D1 <- complete(imp_single)
      data[[3]][1:(N+1),] <- D1
   }else if(i>=n1-N/2+1 & sum(is.nan(data[[3]][n1-(N+1):n1,1]))/(N+1) < 0.2){
      imp_single <- mice(data[[3]][n1-(N+1):n1,], m = 2, method = "pmm") # Impute
missing values
      D1 <- complete(imp_single)
      data[[3]][n1-(N+1):n1,] <- D1
   }else{
      if (i>N/2 & i<n1-N/2+1 & sum(is.nan(data[[3]][(i-N/2):(i+N/2),1]))/(N+1) < 0.2){
         imp_single <- mice(data[[3]][(i-N/2):(i+N/2),], m = 2, method = "pmm") #
Impute missing values
         D1 <- complete(imp_single)
         data[[3]][(i-N/2):(i+N/2),] <- D1
      }
   }
   d1 <- !(is.nan(data[[3]][,1]))
   ddd1 <- which(d1==FALSE)
}
write.csv(data[[3]],"C:\\Users\\Administrator\\Desktop\\data[[3]].csv",fileEncoding
= "GBK")

#F3 <- na.omit(data[[3]])
#writeMat('RData_OK.mat',R1=F1,R2=F2,R3=F3)
```