# Experiment report

|  |  |  |  |
| --- | --- | --- | --- |
| Student Number | 1120182525 | College | 徐特立学院 |
| Name | 梁瑛平 | Major | 计算机科学与技术 |

## Intelligent Transportation Experiment

## Introduction .

With the opening of the era of mobile Internet, every traveler has become a contributor of traffic information. Super-large location data are processed and fused in the cloud to generate traffic information of the whole time without blind spots in the city. Smart transportation hopes to realize the prediction of traffic conditions and help social smart travel and intelligent control of urban traffic.

## Aim

Based on the Internet traffic information to establish the algorithm model. Accurately predict the average travel time of each key road section in a certain period of time

## Data set

Link data

Historical travel time data

## Evaluation index

MAPE（Average absolute percentage error）

TTP: Predicted travel time

TTR: True travel time

N: The predicted number of Links

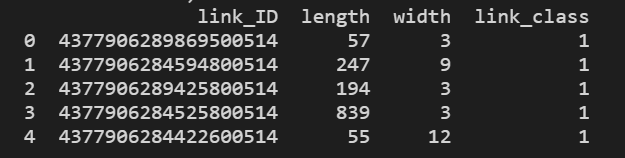
Ti: Number of time slices predicted on the i link



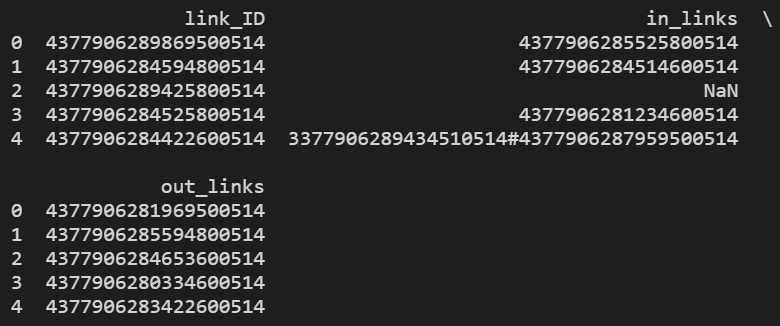
## Experimental process

### 原始数据：

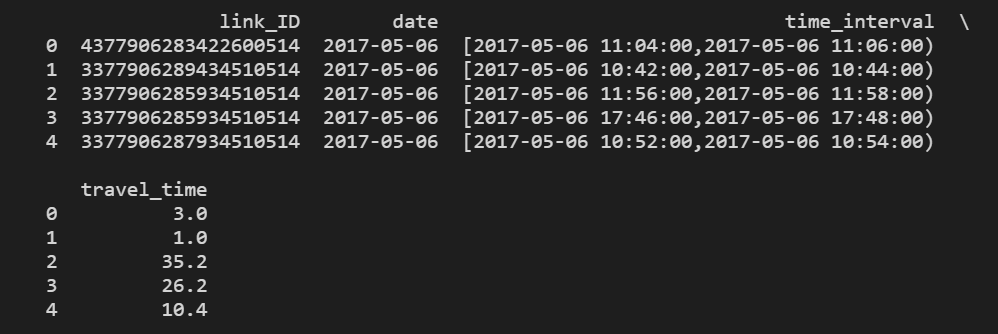
link\_info表里共存着每条路的id, 长度, 宽度和类型,共有132条路：



link\_top里储存每一条路的上下游关系, in\_links里放着这条路的上游路id, 中间用#分割, 而out\_links里则给出了这条路的下游路id; 下游路可以理解为出路, 上游路为入路：



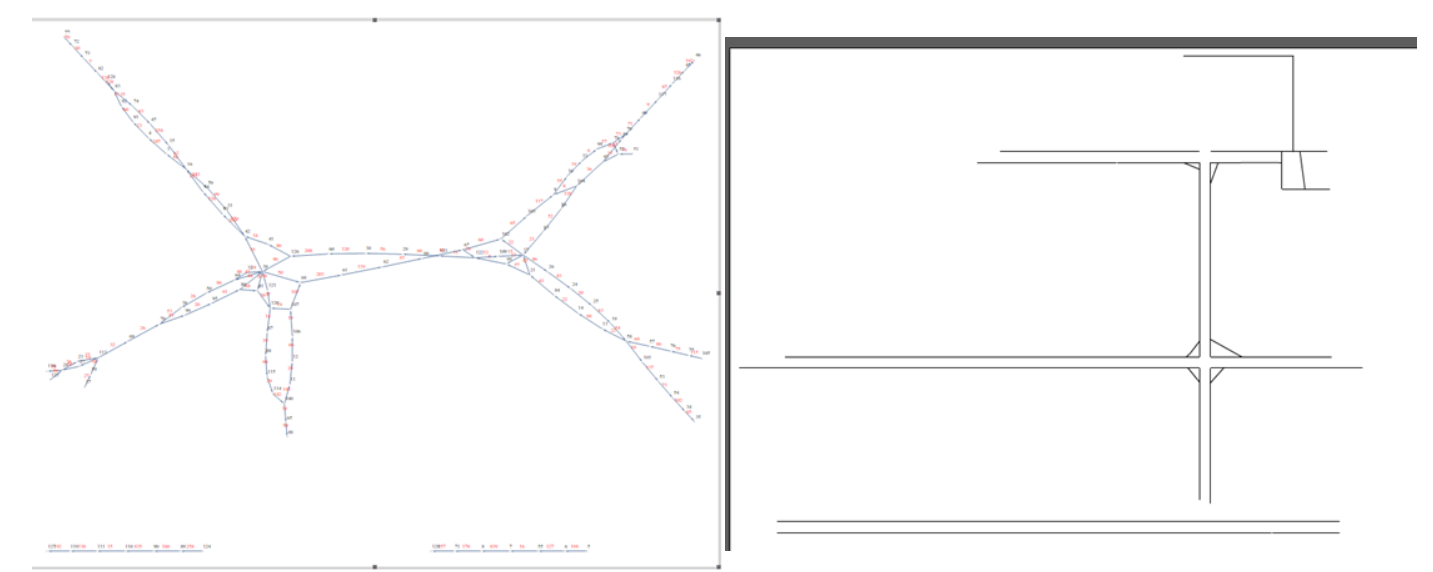
travel\_time表里存着这132条路从2017.4-2017.6以及2016.7每天车通过路的平均旅行时间, 统计的时间间隔为2分钟; 除了2016.4到6月每天的信息, 还有2017.7月每天6:00-8:00, 13:00-15:00, 16:00-18:00的记录, 然后我们需要预测的就是7月每天在早高峰, 午平峰, 晚高峰三个时间段(8:00-9:00, 15:00-16:00, 18:00-19:00)每条路上的车平均旅行时间：



可以使用的基本特征如图：



地理图可视化后的结果：



### 基本思路：

这是一个关于时间序列预测的问题, 并不是普通的回归问题, 而是自回归, 一般的回归问题比如最简单的线性回归模型:Y=a\*X1+b\*X2, 我们讨论的是因变量Y关于两个自变量X1和X2的关系, 目的是找出最优的a和b来使预测值y=a\*X1+b\*X2逼近真实值Y. 而自回归模型不一样, 在自回归中, 自变量X1和X2都为Y本身, 也就是说Y(t)=a\*Y(t-1)+ b\*Y(t-2),其中Y(t-1)为Y在t-1时刻的值, 而 Y(t-2)为Y在t-2时刻的值, 换句话说, 现在的Y值由过去的Y值决定, 因此自变量和因变量都为自身, 这种回归叫自回归。

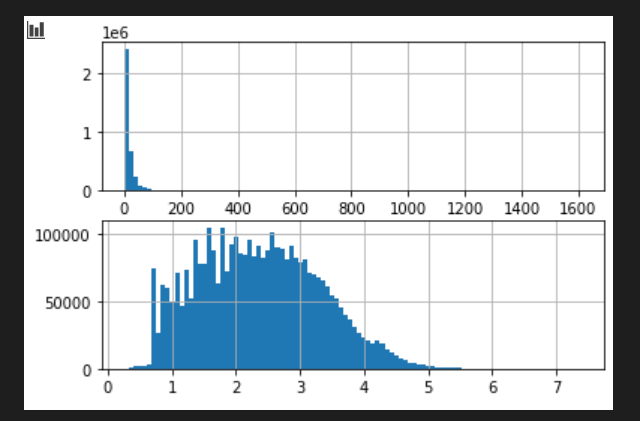
根据题目给出的信息, 除了路本身的信息外, 训练数据基本上只有旅行时间, 而我们要预测的也是未来的平均旅行时间, 而且根据我们的常识, 现在的路况跟过去一段时间的路况是很有关系的, 因此该问题应该是一个自回归问题, 用过去几个时刻的交通状况去预测未来时刻的交通状况。

传统的自回归模型有自回归模型（AR）、移动平均模型（MA）、自回归移动平均模型（ARMA）以及差分自回归移动平均模型（ARIMA）, 这些自回归模型都有着严格理论基础,讲究时间的平稳性, 需要对时间序列进行分析才能判断是否能使用此类模型。

### 特征工程：

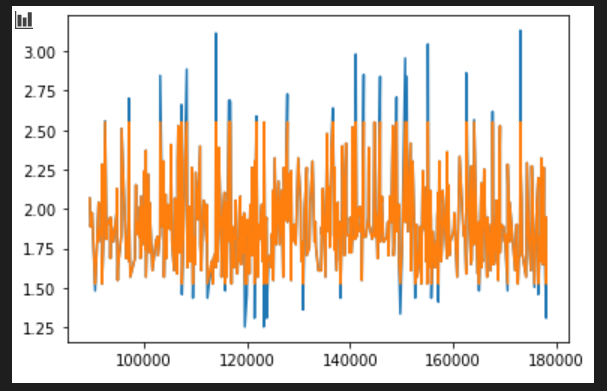
**分布均衡化：**

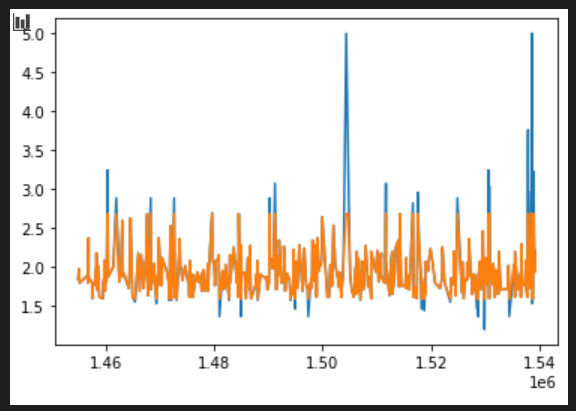
查看数据中预测时间分布，时间分布较为集中（数值较小的label比较多，数值较大的label比较少），因此加上log变换再查看均衡化后的分布。处理后类似正态分布, 比较适合模型来处理：



**处理离群点：**

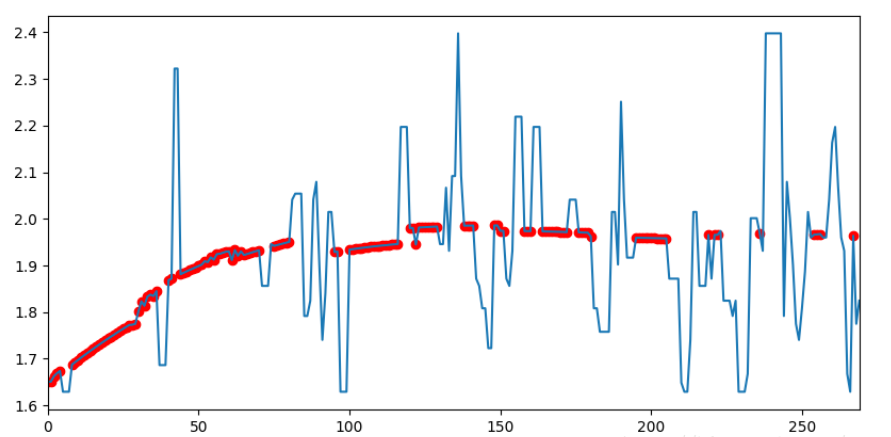
即使做了log变换后, 还是有部分travel\_time值过于大, 为了消除一些离群点的影响, 我们对travel\_time做一个百分位的裁剪clip, 我们把上下阈值设为95百分位和5百分位, 即将所有大于上阈值的travel\_time归为95百分位数, 而小于小阈值的travel\_time设为05百分位数：





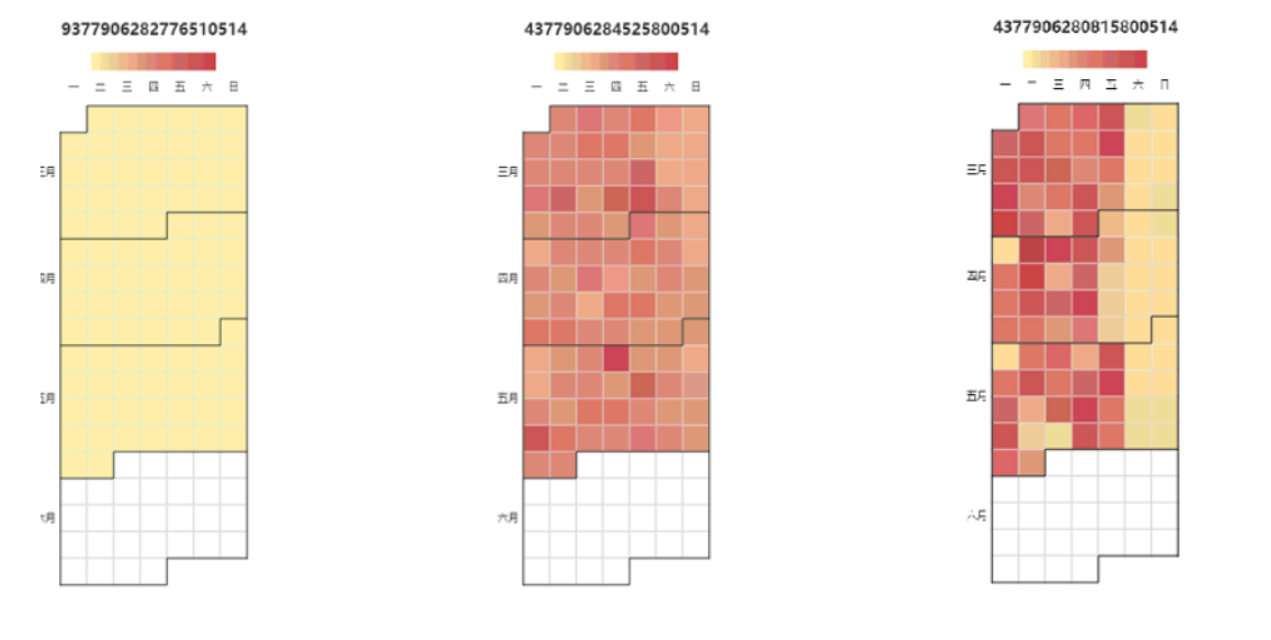
**补全缺失值：**

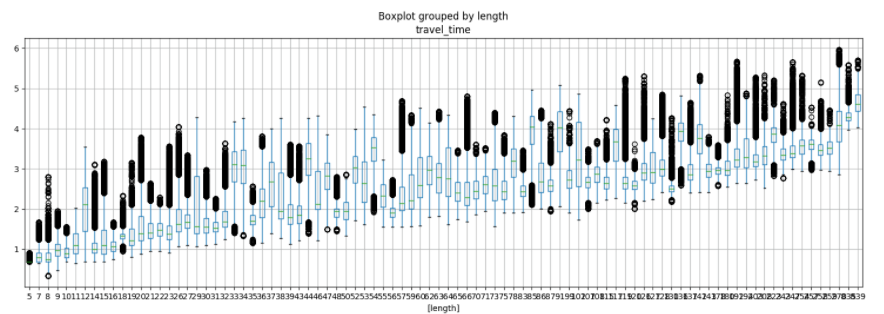
自动补全的方法就是用预训练一个模型去补全缺失值：训练已有的数据，把缺失的数据当做是要预测的数据，有很多模型可以补全缺失值，例如随机森林等，只要feature构造合理，这种补全的方法要比前面提到的手动的方法效果要好一些。我们可以看一下补全的效果, 我们画出某路某天的travel\_time变化如下图，红色的部分是补全的数据，蓝线为原来的数据，可以看出补全的数据比较保守，基本贴近于hour\_trend：

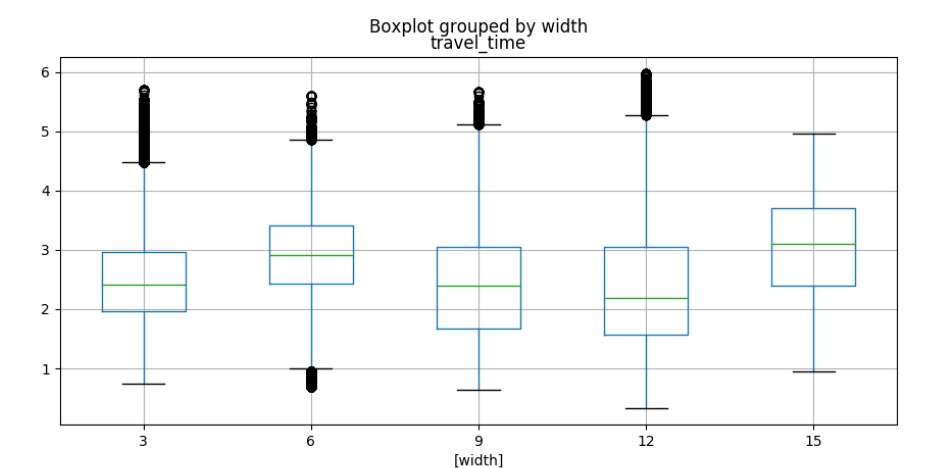


**相关性分析：**

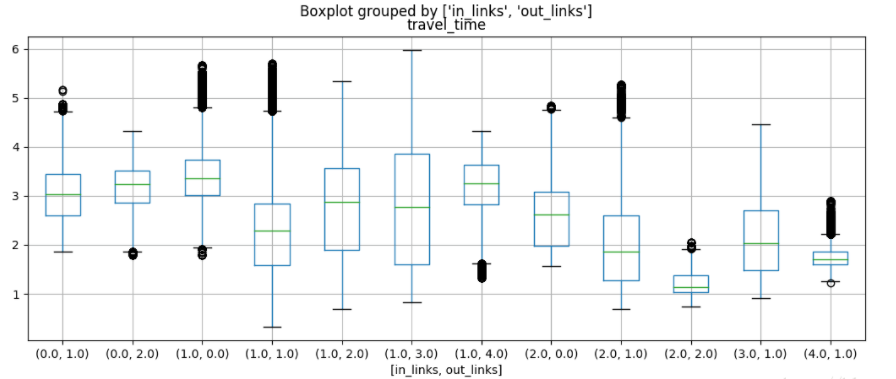
路的长度和宽度：毫无疑问，路的长度与travel\_time是成正比的，路越长，travel\_time越大，所以路的长度特征应该是非常重要的，由下图也可以看出他们存在正相关关系。



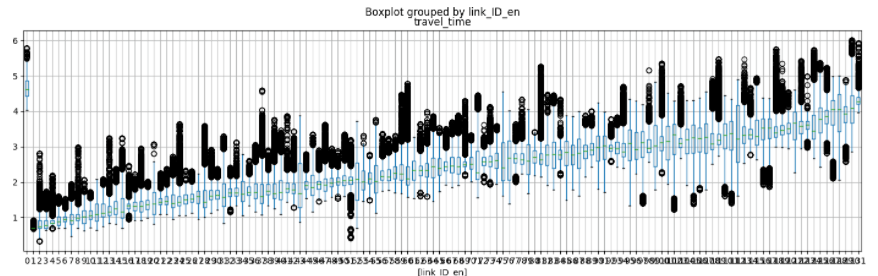




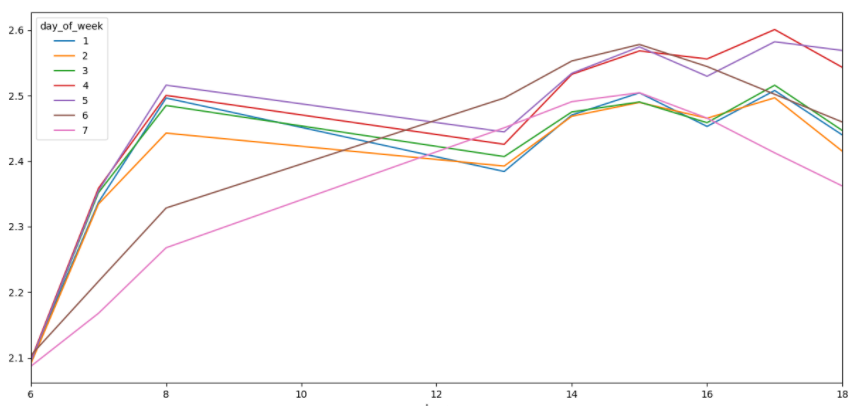
上下游关系：对于每一条路来说, 基本上都有上游和下游，但是有的路处于尽头，本身就没有上游或者下游，我们这里只根据上游和下游的数量进行划分，统计每一条路的上游和下游路的个数，然后画出箱线图。其中in\_links和out\_links分别表示的上游路和下游路的数量。



ID特征： scikit库有直接将ID映射到标签的工具，映射后ID特征变为了从1到132离散的数字：



基本时间特征：根据常识来说，车在工作日是比较多的，travel\_time相对大，而在一天之内，上下班高峰期也直接影响travel\_time，而且假期也是很影响大家的出行的．我们可以跟上面一样对week\_day和hour以及vacation分别画出箱线图看看，但我后来发现了week\_day和hour是有一定的关联的，比如周一的早上8点与周末的早上8点是完全不一样的，下面给一天之内每个小时平均travel\_time的变化情况。



### 训练模型：

**输入数据：**

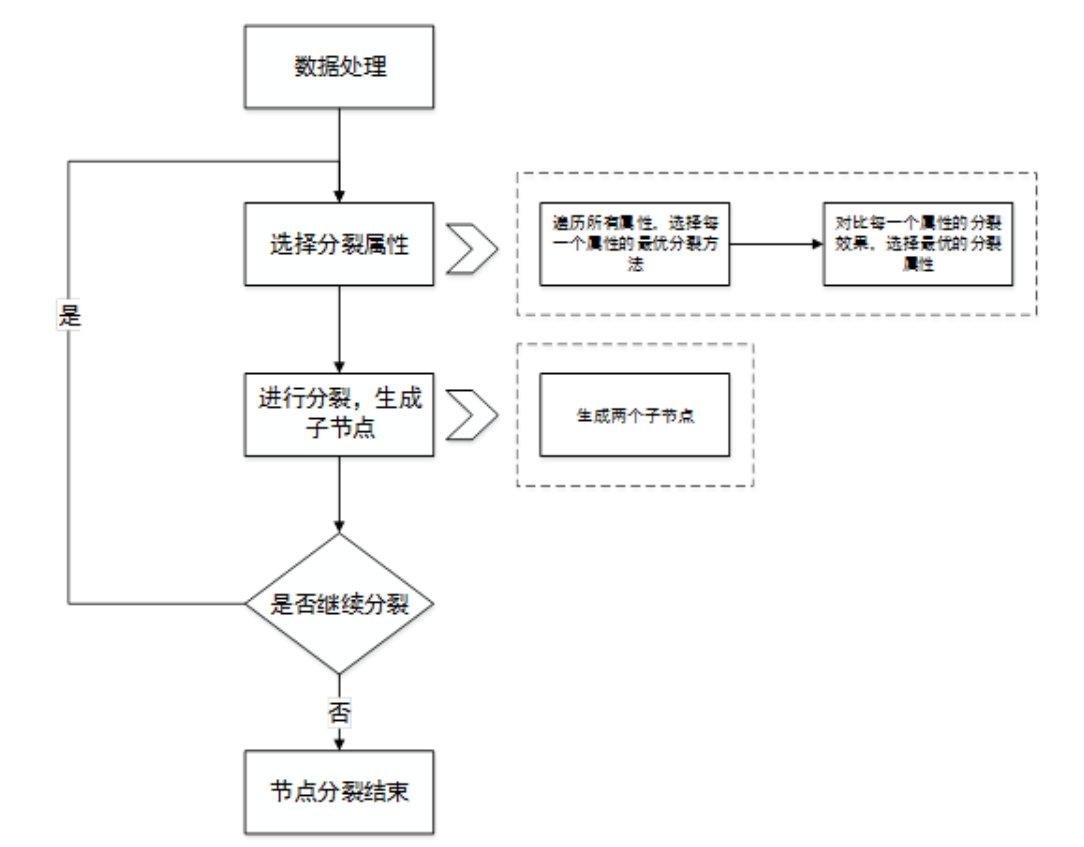
训练模型时直接用上面的feature训练对应的travel\_time就可以，但是时间序列feature在交叉验证和测试的时候就不能跟训练一样了，因为当我们在预测出t时刻的travel\_time后，需要把这个travel\_time作为预测t+1时刻travel\_time的lagging1特征，这个lagging特征是需要根据上次预测的结果进行更新的，如此反复直到预测到最后一个时刻的travel\_time。

**验证数据：**

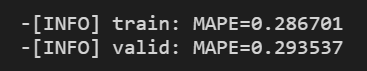
５次交叉验证。

**模型选择：**

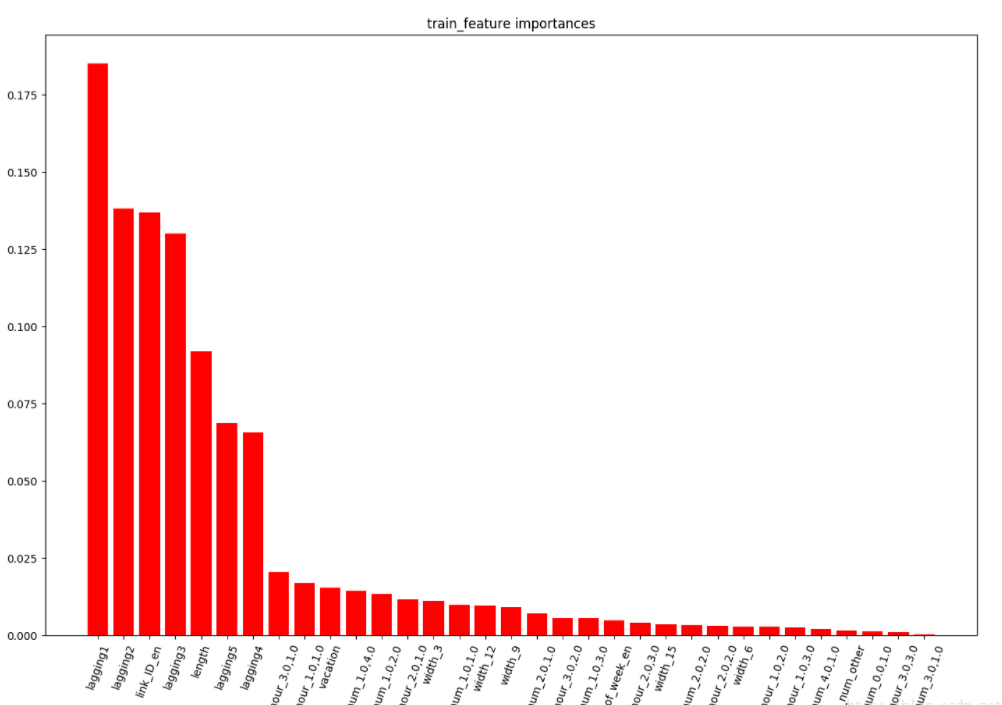
模型使用XGBRegressor提供的gbtree，即CART。分类与回归树（Classification and Regression Trees, CART）是由四人帮Leo Breiman, Jerome Friedman, Richard Olshen与Charles Stone于1984年提出，既可用于分类也可用于回归。该算法对于回归树，采用样本方差衡量节点纯度。节点越不纯，节点分类或者预测的效果就越差。



## Experimental results and analysis



分数不是很高的原因可能是有些无用特征没有删除而导致了过拟合。



## Gain and experience