

2017 年吉林省大学生数学建模竞赛

承 诺 书

我们仔细阅读了《全国大学生数学建模竞赛章程》和《全国大学生数学建模竞赛参赛规则》（以下简称为“竞赛章程和参赛规则”，可从全国大学生数学建模竞赛网站下载）。

我们完全明白，在竞赛开始后参赛队员不能以任何方式（包括电话、电子邮件、网上咨询等）与队外的任何人（包括指导教师）研究、讨论与赛题有关的问题。

我们知道，抄袭别人的成果是违反竞赛章程和参赛规则的，如果引用别人的成果或其他公开的资料（包括网上查到的资料），必须按照规定的参考文献的表述方式在正文引用处和参考文献中明确列出。

我们郑重承诺，严格遵守竞赛章程和参赛规则，以保证竞赛的公正、公平性。如有违反竞赛章程和参赛规则的行为，我们将受到严肃处理。

我们授权全国大学生数学建模竞赛组委会，可将我们的论文以任何形式进行公开展示（包括进行网上公示，在书籍、期刊和其他媒体进行正式或非正式发表等）。

我们参赛选择的题号是（从 A/B/C/D/E 中选择一项填写）： A

我们的报名参赛队号为（8 位数字组成的编号）：

所属学校（请填写完整的全名）： 吉林大学

参赛队员（打印并签名）： 1. 成仕骐

2. 颜正熙

3. 刘瀚霆

指导教师或指导教师组负责人（打印并签名）：

（论文纸质版与电子版中的以上信息必须一致，只是电子版中无需签名。以上内容请仔细核对，提交后将不再允许做任何修改。如填写错误，论文可能被取消评奖资格。）

日期： 2017 年 5 月 2 日

赛区评阅编号（由赛区组委会评阅前进行编号）：

2017 年全国大学生数学建模竞赛

编号专用页

赛区评阅编号（由赛区组委会评阅前进行编号）：

赛区评阅记录（可供赛区评阅时使用）：

[illegible]

关于用户体验变量与网络测变量之间函数关系的研究

摘要

本文针对基于 TCP 视频传输及播放产生的卡顿现象,用题目所给的大量数据进行分析整理,以多元回归理论为基础进行了完整的建模工作。由于题目所求为用户体验评价变量(初始缓冲时延、卡顿时长占比)量与网络侧变量(初始缓冲峰值速率、播放阶段平均下载速率、端到端环回时间(E2E RTT))的函数关系。

于是先将本题分为两个问题来分别建模:

- (1) 初始缓冲时延与网络侧变量的函数关系。
- (2) 卡顿时长占比网络测变量的函数关系。

针对问题(1),查阅资料并且利用 MATLAB 软件处理分析数据并且作出图像关系,得到(1)中初始缓冲时延只与初始缓冲峰值速率和端到端环回时间(E2E RTT)有关,筛选一些偏差过大的点后用多元线性回归方法拟合出(1)所求的函数关系。而针对问题(2),研究小组依旧先使用求相关度的方法确定卡顿时长占比只与播放阶段平均速率有关。接下来由数据分析将视频全程感知速率作为中间变量,寻找它与其他变量的相关关系,发现其中具有服从正态分布、线性等关系,分别依次求出变量之间的函数关系,联立起来后即得到卡顿时长与播放阶段平均速率之间的函数关系。

关键词: TCP MATLAB 多元线性回归 正态分布 数据分析

一、问题重述

1.1 引言

随着宽带网络的升级，以及智能终端的普及，越来越多的用户选择在移动智能终端上用应用客户端 APP 观看网络视频。每个人都希望观看的视频清晰且流畅，无论是初始缓冲时间过长或是视频播放中卡顿时间过长均会严重影响客户体验，毕竟没有用户愿意在难得休闲的时刻花上时间去等待视频无穷无尽的加载。许多人市场将卡顿及缓冲完全归结于网速，其实其中的因素错综复杂，远没有想象的那么简单。因此，将其中诸多因素的相关性整合清楚进而改进 APP 客户端播放视频的流畅度与清晰度，成为了各大网络公司优先需要解决的问题。

1.1.1 TCP 工作机制的了解

1. 在源主机上，应用层将一串应用数据流传递给传输层。
2. 传输层将应用层的数据流截成分组，并加上 TCP 报头形成 TCP 段，送交网络层。
3. 在网络层给 TCP 段加上包括源、目的主机 IP 地址的 IP 报头，生成一个 IP 数据包，并将 IP 数据包送交链路层。
4. 链路层在其 MAC 帧的数据部分装上 IP 数据包，再加上源、目的主机的 MAC 地址和帧头，并根据其目的 MAC 地址，将 MAC 帧发往目的主机或 IP 路由器。
5. 在目的主机，链路层将 MAC 帧的帧头去掉，并将 IP 数据包送交网络层。
6. 网络层检查 IP 报头，如果报头中校验和与计算结果不一致，则丢弃该 IP 数据包；若校验和与计算结果一致，则去掉 IP 报头，将 TCP 段送交传输层。
7. 传输层检查顺序号，判断是否是正确的 TCP 分组，然后检查 TCP 报头数据。若正确，则向源主机发确认信息；若不正确或丢包，则向源主机要求重发信息。
8. 在目的主机，传输层去掉 TCP 报头，将排好顺序的分组组成应用数据流送给应用程序。这样目的主机接收到的来自源主机的字节流，就像是直接接收来自源主机的字节流一样。

1.2 问题的提出

研究表明影响初始缓冲时延和卡顿时长占比的主要因素有初始缓冲峰值速率、播放阶段平均下载速率、端到端环回时间（E2E RTT），以及视频参数有关。因此本文提出如下问题：试根据附件 1 提供的实验数据建立用户体验评价变量（初始缓冲时延，卡顿时长占比）与网络侧变量（初始缓冲峰值速率，播放阶段平均下载速率，E2E RTT)之间的函数关系。附件 1-2017A 题华为网络侧视频体验.zip

二、问题分析

2.1 问题（1）的分析

问题（1）需要解决的是大量样本数据以及多个变量之间的相关关系，研究小组先利用 MATLAB 软件作出散点关系图，根据查询相关资料，用多元线性拟合的方法，建立数学模型，从而求出问题（1）的函数关系。

而问题（2）开始的思路也和问题（1）是相同的，但复杂的是需要找到中间变量，建立相关数学模型，多个函数联立才能最终求得问题（2）的函数关系。

综合来说两题的解题思路都是利用多元线性回归的方式来建立模型，通过 MATAB 拟合求得方程系数从而解决问题

三、基本假设

- (1) 同一华为型号无线终端。
- (2) 视频码率为 2934 kbps。
- (3) 播放阶段总时长为 30000 ms。
- (4) 卡顿门限为 0，即播放阶段缓冲区没有数据时出现卡顿。
- (5) 重播放门限为 2.7s，即缓冲区下载 7921.8kb 数据时重新播放。
- (6) 使用客户端观看视频时网络链接保持正常。

四、相关符号

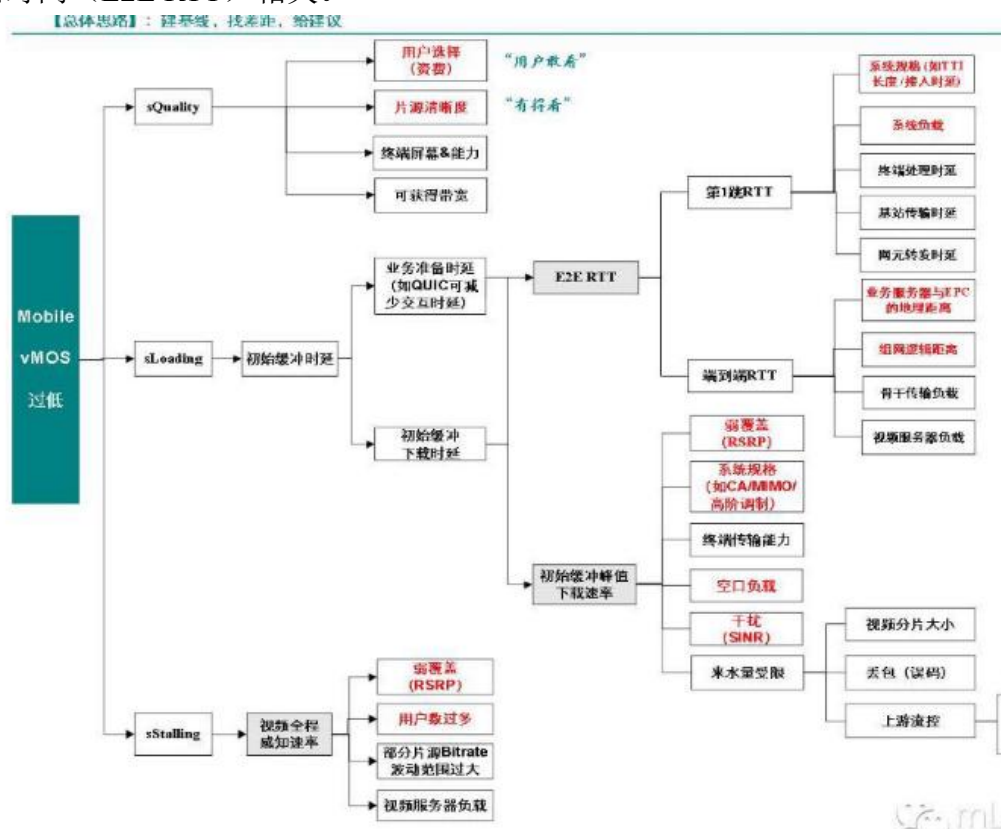
x_1	初始缓冲峰值速率
x_2	$E2E\ RTT$
x_3	播放阶段平均速率
y_1	初始缓冲时延
y_2	卡顿占比
t	卡顿时长
\bar{t}	播放总时长（恒为30000ms）
v_1	视频全程感知速率

五、模型的建立与求解

4.1 问题（1）的建模与解答

4.1.1 数据预处理

查阅大量资料后研究小组得到初始缓冲时延只与初始缓冲峰值速率以及端到端环回时间（E2E RTT）相关。



为了验证该信息的准确性，决定先利用编码来求出初始缓冲时延与播放阶段平均速率的相关度。

$$b = \text{corrcoef}(D_{lie}, E_{lie})$$

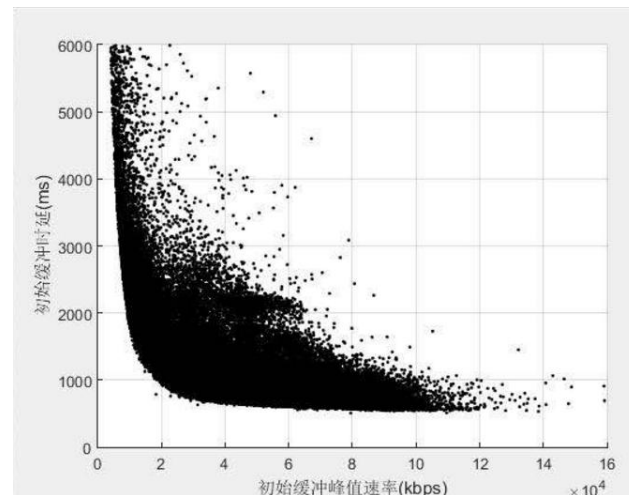
$$b =$$

$$\begin{bmatrix} 1.0000 & -0.2421 \\ -0.2421 & 1.0000 \end{bmatrix}$$

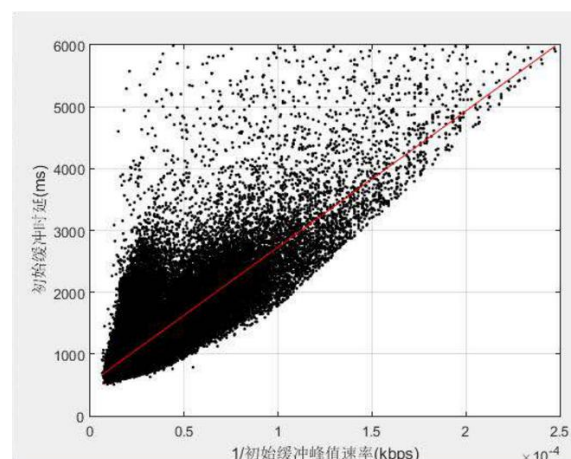
得到二者相关度的绝对值为 0.2421，相关度较小，而初始缓冲时延与初始缓冲峰值速率和端到端环回时间相关度为 0.69 和 0.54。这也正与小组所查阅的资

料得到的信息一致。于是研究小组决定针对初始缓冲时延与初始缓冲峰值速率以及端到端环回时间（E2E RTT）来建立相关模型。

首先我们先研究初始缓冲时延和初始缓冲峰值速率之间的关系。首先研究小组使用 matlab 绘制散点图，寻找二者之间的关系如图所示



通过散点图，小组决定将所有视频缓冲峰值速率的数值取倒数，再次进行作图。通过作图小组发现有一些点偏离得过于明显，而且这些点是极少数的情况，这是由于未知的原因造成的，为了不影响数学模型的拟合程度，小组决定筛选掉一些取值极特殊的点。再次通过 matlab 进行拟合，发现初始缓冲峰值速率的倒数和初始缓冲时延有一定的线性相关关系。拟合程度达到了 0.66。



程序代码如下：

```
zhuanzhi=[ones(size(Elie)) fanBlie]  
[b,bint,r,rint,stats]=regress(Elie,zhuanzhi)
```

```
b=  
528.273792132611  
22026650.2034691
```

```
stats=  
0.658695231760567
```

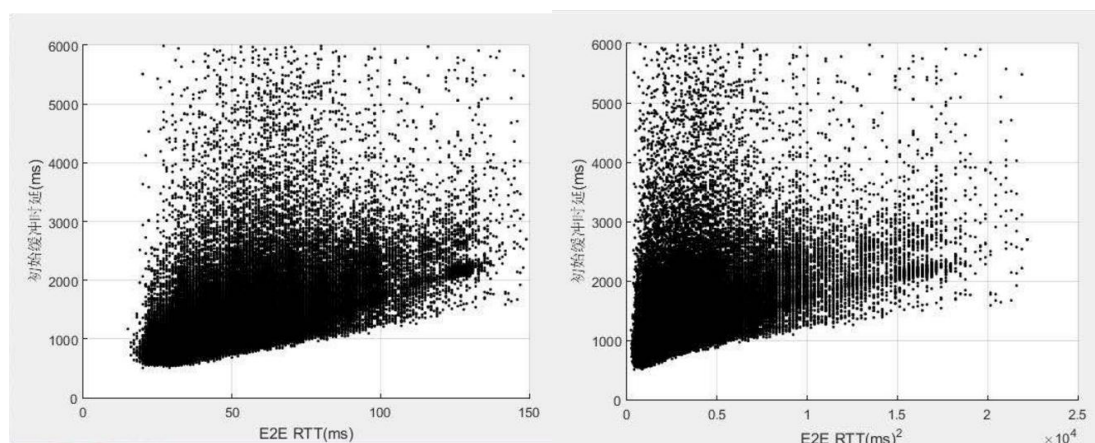

169216.498845536

0

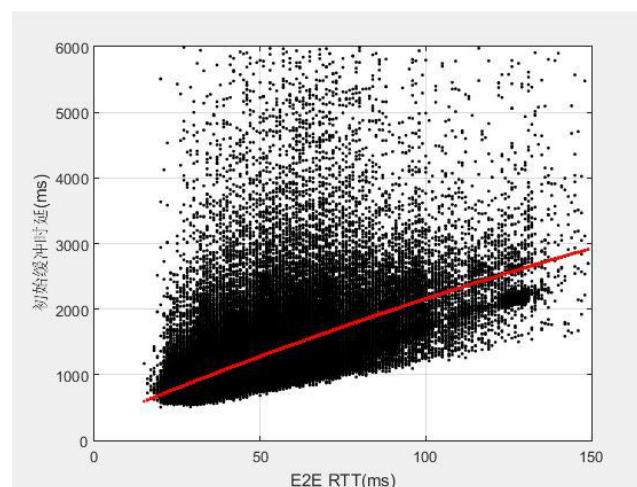
154724.217262991

```
y=528.2738+2.2027*10.^7.*fanBlie  
plot(fanBlie,Elie,'k.',fanBlie,y,'r')  
xlabel('1/初始缓冲峰值速率(kbps)')  
ylabel('初始缓冲时延(ms)')  
grid on
```

最后拟合的函数模型可能初始缓冲时延与初始缓冲峰值速率的倒数成正相关关系。同理，小组继续研究了初始缓冲时延与端对端环回时间之间的关系。使用 `matlab` 软件进行散点图的绘制，得到了初始缓冲时延与端对端环回时间，端对端环回时间的整数次方的关系，如图所示



通过研究小组的尝试，最终发现初始缓冲时延和端对端环回时间的平方有较高的线性相关关系。通过筛选边缘的点之后再次使用 `matlab` 进行拟合，得到的曲线方程拟合度为 0.431。



程序代码如下：

```
zhuanzhi=[ones(size(Elie)) E2ERTTms.^2 E2ERTTms]  
[b,bint,r,rint,stats]=regress(Elie,zhuanzhi)
```

b =

```
284.8581  
-0.0225  
20.9781
```

stats =

```
0.4309  
0.0000    0.2168    0    3.0333  
yy=284.8581-0.0225*(E2ERTTms).^2+20.9781*E2ERTTms  
plot(E2ERTTms,Elie,'k.',E2ERTTms,yy,'r')  
xlabel('E2ERTT(ms)')  
Ylabel('初始缓冲时延(ms)')  
grid on
```

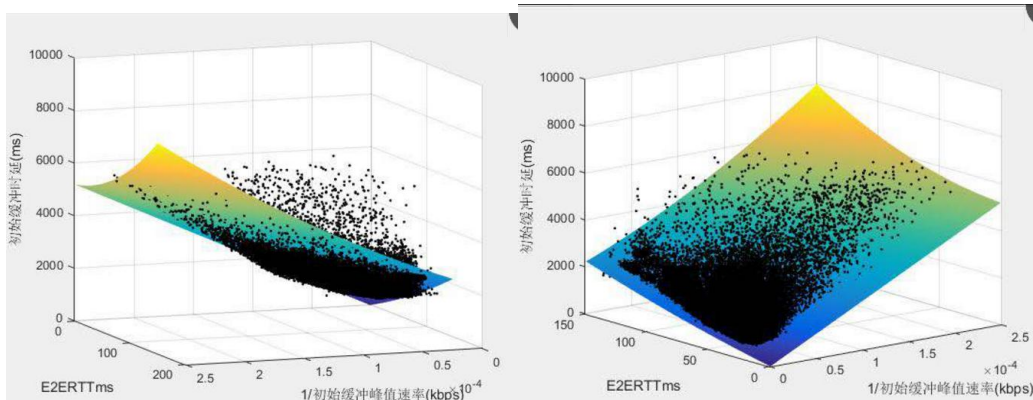
Y1 与 X2 的平方有一定的拟合度，但是拟合度不是很高，但是依照图形与算法分析存在明显的线性关系，需要进行筛点，若为苛求两个模型更高的拟合度需要筛掉大量的点。所以研究小组考虑到交叉项可能会使模型的拟合度更高。

至此小组决定使用多元线性回归的方法描述初始缓冲速率与初始缓冲峰值速率以及端对端环回速率之间的关系。在这里小组将初始缓冲峰值速率的倒当做做一个自变量。在此小组假设 y1 可以用含 x1,x2 的多项式线性表示。

$$y_1 = p_1 + p_2 \frac{1}{x_1} + p_3 x_2^2 + p_4 x_2 + p_5 \left(\frac{x_2}{x_1} \right)^2 + p_6 \left(\frac{x_2}{x_1} \right)$$

再通过计算机来实现系数的求解以及拟合度的计算，经过研究发现，交叉项并不能被去掉，如果在拟合的时候不考虑交叉项(1/x1)x2 时，拟合度会显著下降。

最终通过拟合，得到了最终的函数三维图像如图所示



程序代码如下

```
zhuanzhi=[ones(size(E2ERTTms)) fanBlie (E2ERTTms.^2) E2ERTTms  
(fanBlie.*E2ERTTms).^2 (fanBlie.*E2ERTTms)];  
[b,bint,r,rint,stats]=regress(Elie,zhuanzhi);  
stats  
b
```

```

stats =
    0.8425
    0.0001    9.3818    0    7.1391

b=

-32.3065244325960
20905754.7673380
0.0242875502362586
11.2551802180072
1200937.26317936
-29753.9564865302
>> x=0:1:150
y=0.000001:0.000001:0.000250
[X,Y]=meshgrid(x,y)
f=-32.3065+Y.*20905754.767+0.02428755023*(X.^2)+11.255180*X+1200937.2631
79*(Y.*X).^2-29753.95647*(Y.*X)
surf(Y,X,f)
shading flat

```

```

>>hold on
plot3(fanBlie,E2ERTTms,Elie,'k.')
最后求出的函数解析式为

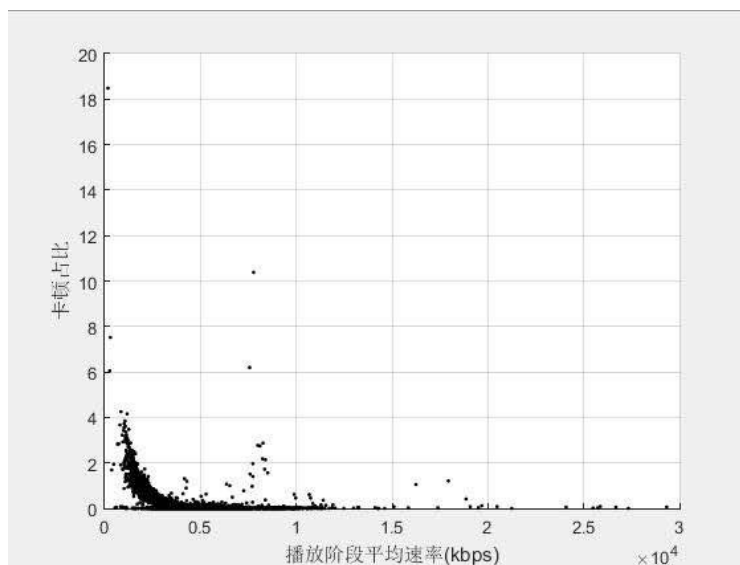
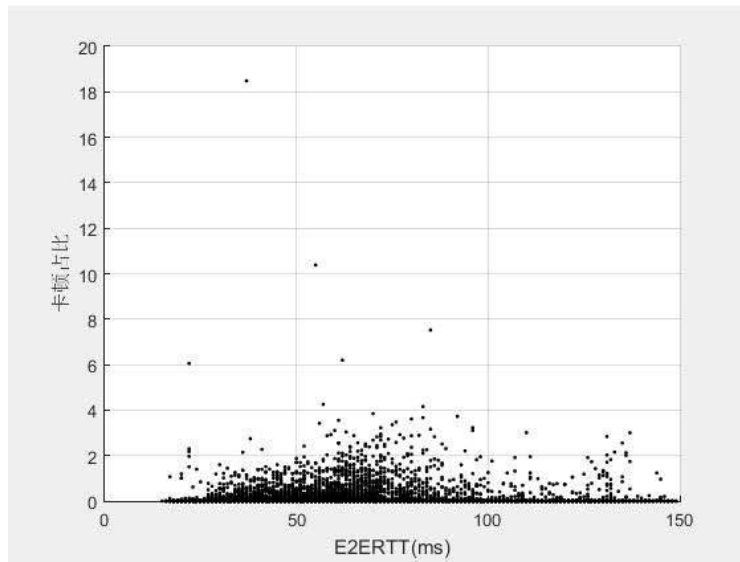
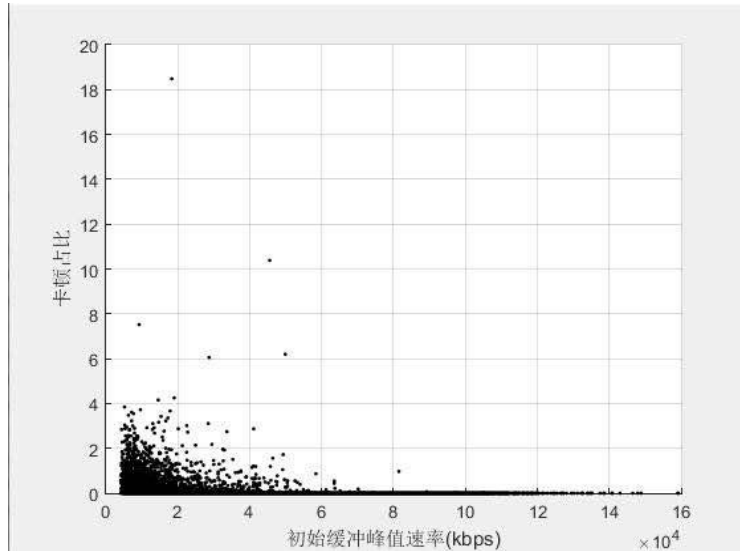
```

$$y_1 = -32.3065 + 2.0906 \times 10^7 \frac{1}{x_1} + 0.0243x_2^2 + 11.2552x_2 + 1.2009 \times 10^6 \left(\frac{x_2}{x_1} \right)^2 - 2.9754 \times 10^4 \left(\frac{x_2}{x_1} \right)$$

最终通过 matlab 进行拟合度的检验，拟合度达到了 0.8425。

4.2 问题（2）的建模与求解

由于问题（2）研究的是卡顿时长占比网络侧变量的函数关系。通过 matlab 软件描点，分别作出卡顿时长占比与网络测变量的散点图如下所示。



利用如下编码求 E2E RTT 与卡顿时长占比的相关度
 $c = \text{corrcoef}(\text{E2ERTTms}, \text{Flie})$

c =

```
1.0000    0.0666
0.0666    1.0000
```

即二者之间相关度为 0.0666，是微相关。

同样的利用如下代码求出初始峰值速率和卡顿市场占比的相关度

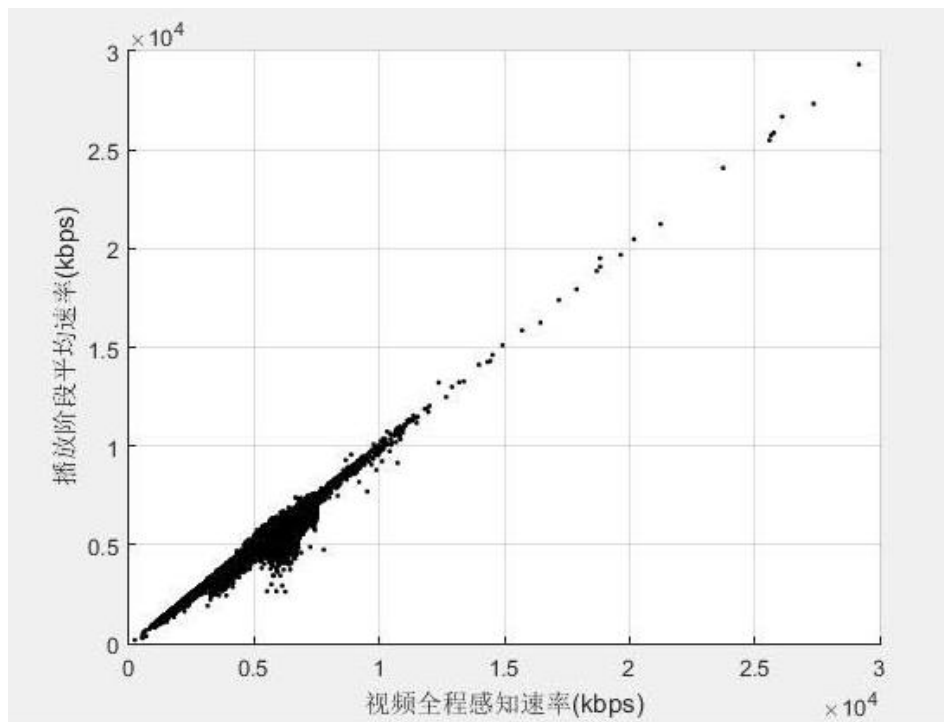
```
b=corrcoef(Blie,Flie)
```

b =

```
1.0000   -0.1150
-0.1150    1.0000
```

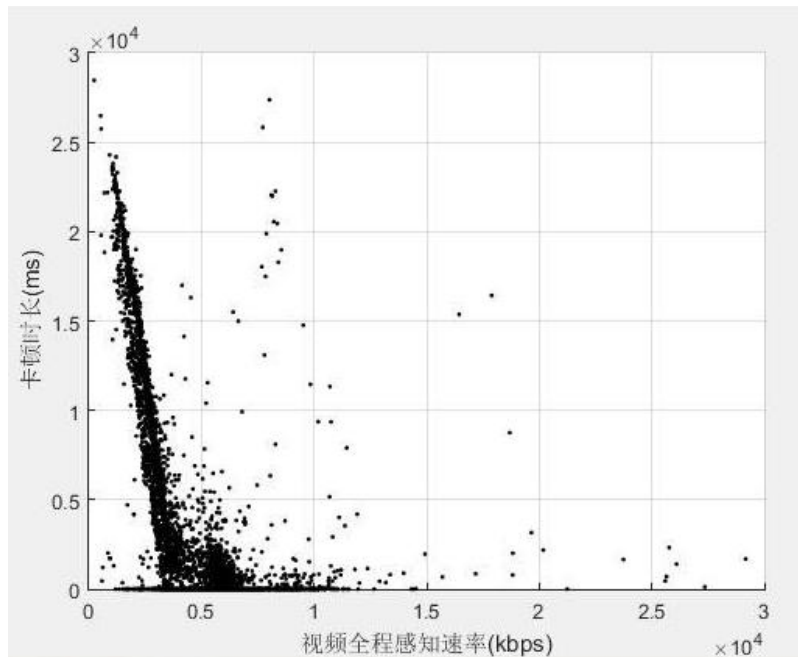
b 的绝对值较小，说明与其相关性不大。与解决第（1）个模型时相同，用多元线性拟合时，多项式 $y_2 = \sigma(x_1, x_2, x_3)$ 中，含 x_1, x_2 的项的系数的绝对值大多为 0 或是极其接近 0 的数。于是研究小组开始着手研究卡顿占比与播放阶段平均速率的关系，即着手去寻找函数 $y_2 = f(x_3)$ 。

先着手于播放阶段平均速率与其他变量的关系，发现视频全程感知速率与其有着明显的相关关系，于是利用 matlab 软件作出二者之间的散点关系图如下所示。

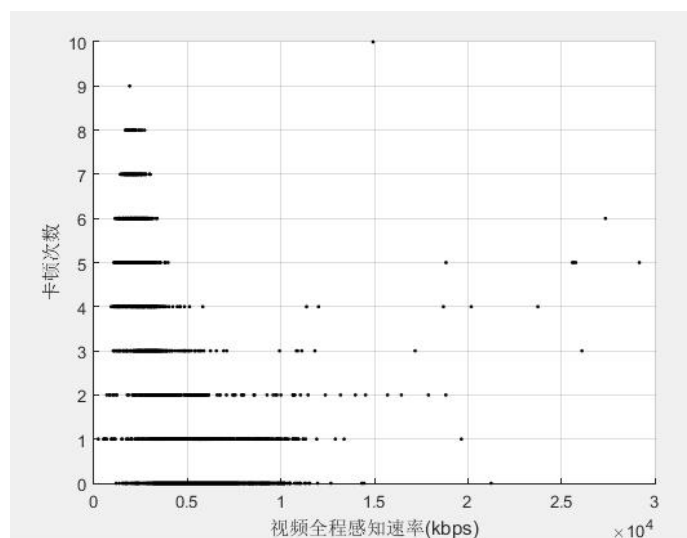


研究小组惊喜的发现，二者有惊人的线性相关关系，并且二者比值接近于 1，于是给出二者之间的模型为 $x_3 = v_1 + \varepsilon$ ，其中 ε 是一个常数，代表二者之间

由于外界因素而产生的一定的有规律的误差。至此研究小组发现视频全程感知速率是一个重要的中间变量。接下来开始分析重要的中间变量视频全程感知速率与其他变量之间的关系。用 Matlab 软件逐一绘图分析时，发现其与卡顿时长有着较强的相关关系如下图所示。



研究小组尝试用线性回归方法得到其相关方程，但是发现得到的直线与图中较为明显的一条直线有较大偏差，经研究发现这是由于卡顿时长为 0 的点集中在坐标轴上造成的。由此发现，是否出现卡顿很大程度上决定了视频感知速率的数值。经过大量查阅资料，视频全程感知速率小于或接近于视频码率时，卡顿概率大大增加。于是研究小组又调查了卡顿次数与视频全程感知速率的关联，用 matlab 绘制两者之间的散点图如图所示



研究小组希望最后建立的模型适用于视频出现卡顿和不出现卡顿两种情况，通过对上图的研究，小组发现对于某个确定的 n ，视频全程感知速率的分布有一定规律，即中间及其密集，两侧比较稀疏，两侧有对称的明显趋势，，于是组员

大胆使用了正态分布模型来拟合这种现象。研究小组是这样操作的，首先根据图像，小组将所有样本点分为 $n=0, n=1, \dots, n=8$ 这几组，然后固定某个的数值，研究视频全程感知速率的分布。为了能够较为准确的使用正态分布模型，研究小组对 $n=0$ 的较为集中的一个区域所有样本点取了一个等距分划。研究小组取出了数据比较密集的 (1000, 13000) 区间，并将其进行距离为 1000 的等距分划，并取出每个区间的中点的值，记为 x ，统计落在每个小区间里包含的点数，记为 y 。由于数据较多，论文只列出部分数据进行示例

随着 n 的变化，视频全程感知速率的范围如图所示：

$n=0$	(1000, 13000)
$n=1$	(1000, 12000)
$n=2$	(1000, 12000)
$n=3$	(1000, 7500)
$n=4$	(1000, 6000)
$n=5$	(1000, 4500)
$n=6$	(1000, 4000)
$n=7$	(1000, 3000)
$n=8$	(1700, 2700)

在 $n=0$ 时，研究小组将 1000-13000 的数据取一个距离为 1000 的等距分划，并统计落在每个分划中的数据个数，并取每个分划的中点，用一个向量 (x, y) 来表示。 x 表示分划区间中点的值， y 表示这个分划中的数据个数。作为示例，下图表示了 $n=0$ 时通过以上方法处理数据得到的结果，如图所示：

(1500, 12)
(2500, 30)
(3500, 1179)
(4500, 1910)
(5500, 28994)
(6500, 50925)
(7500, 1593)
(8500, 80)
(9500, 58)
(10500, 21)
(11500, 7)

用 matlab 软件处理这一组数据，再根据正态分布的表达式 $v_1 \sim N((\mu(n), \sigma^2(n)))$ ，并用 $\rho(n)$ 表示它的密度函数，即：

$$\rho(n) = \frac{1}{\sqrt{2\pi}\sigma(n)} e^{-\frac{[v_1 - \mu(n)]^2}{2\sigma^2(n)}}$$

经过计算机拟合得到的参数 μ 和 σ ，记为 μ 和 σ 。根据 n 的变化，我们可以重复以上操作得到一系列 μ 和 σ 。再次使用 matlab 软件，通过绘制散点图进行线性

回归的方法，我们发现了 μ 和 σ 在经过处理后和 n 之间构成了线性关系！

$$\frac{1}{\mu} = k_{\mu} n + b_{\mu}$$

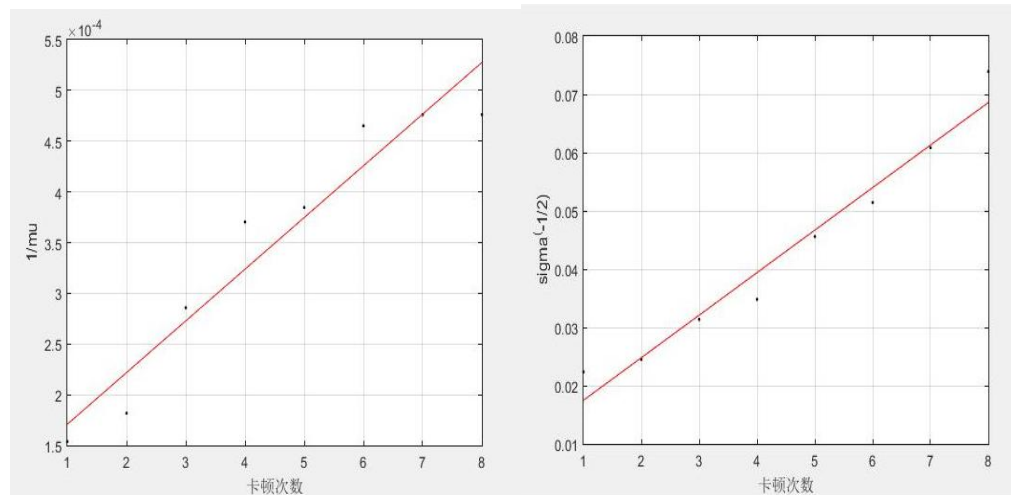
$$\frac{1}{\sqrt{\sigma}} = k_{\sigma} n + b_{\sigma}$$

这正符合我们的期望，于是 $v_i \sim N((\mu(n), \sigma^2(n)))$ 。

再次研究时，通过以上关系式，我们编写了以下代码作图。

<code>zhuanzhi=[ones(8,1) z]</code>	<code>zhuanzhi=[ones(8,1) z]</code>
<code>[b,bint,r,rint,stats]=regress(1./x,zhuanzhi)</code>	<code>[b,bint,r,rint,stats]=regress(y.^(-1/2),zhuanzhi)</code>
<code>b =</code>	<code>b =</code>
<code>1.0e-03 *</code>	<code>0.0102</code>
<code>0.1199</code>	<code>0.0073</code>
<code>0.0510</code>	
<code>stats =</code>	<code>stats =</code>
<code>0.9275 76.7459 0.0001 0.0000</code>	<code>0.9649 164.9325 0.0000 0.0000</code>
<code>yy1=1.1991*10.^(-4)+5.0961*10.^(-5)*z</code>	<code>yy2=0.0102+0.0073*z</code>
<code>plot(z,1./x,'k.',z,yy1,'r')</code>	<code>plot(z,y.^(-1/2),'k.',z,yy2,'r')</code>
<code>xlabel('卡顿次数')</code>	<code>xlabel('卡顿次数')</code>
<code>ylabel('1/mu')</code>	<code>ylabel('sigma^(-1/2)')</code>
<code>grid on</code>	<code>grid on</code>

得到的图像如下所示。

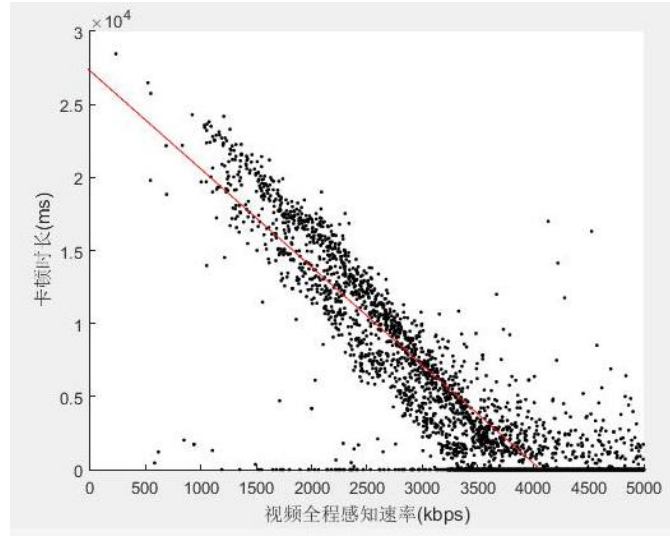


拟合后得到参数的值为

$$\begin{cases} k_{\mu} = 5.096 \times 10^{-5} \\ b_{\mu} = 1.1991 \times 10^{-4} \end{cases} \begin{cases} k_{\sigma} = 0.0073 \\ b_{\sigma} = 0.0102 \end{cases}$$

到这里视频全程感知速率与卡顿次数 n 的模型已建立完毕并求解完成。并且对之前卡顿时长和视频全程感知速率之间的关系有了新的认识。

接下来研究小组重新着手于视频全程感知速率与卡顿时长的关系，通过对二者之间数据与图像的处理，得到了清晰度更高、相关性更强的关系图，如下所示。



可以用函数 $t = k_t v_1 + b_t$ 来拟合这条直线。经过 matlab 调用 polyfit 函数求

解出参数的值： $\begin{cases} k_t = -6.67497 \\ b_t = 27327.69 \end{cases}$ 。

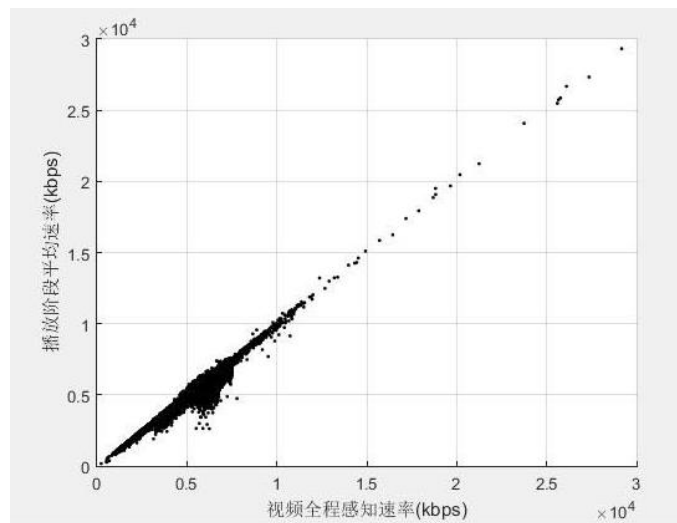
综上，研究小组将所得到的函数关系联立起来，并由提示给出的卡顿占比 = 卡顿时长 / 播放时长的关系，经过如下推导，得到卡顿占比与播放阶段平均速率的函数关系：

$$\begin{aligned} y_2 &= \frac{t}{\bar{t} - t} = \frac{t(v_1)}{\bar{t} - t(v_1)} = \frac{t[v_1(x_3)]}{\bar{t} - k_t v_1(x_3) - b_t} = \frac{k_t(x_3 + \varepsilon(x_3)) + b_t}{\bar{t} - k_t(x_3 + \varepsilon(x_3)) - b_t} \\ &= \frac{k_t \frac{1}{\sqrt{2\pi}\sigma(n)} e^{-\frac{[(x_3 + \varepsilon(x_3)) - \mu(n)]^2}{2\sigma^2(n)}} + b_t}{\bar{t} - k_t \frac{1}{\sqrt{2\pi}\sigma(n)} e^{-\frac{[(x_3 + \varepsilon(x_3)) - \mu(n)]^2}{2\sigma^2(n)}} - b_t} \end{aligned}$$

$$\text{其中: } \begin{cases} k_t = -6.67497 \\ b_t = 27327.69 \end{cases} \begin{cases} k_{\mu} = 5.096 \times 10^{-5} \\ b_{\mu} = 1.1991 \times 10^{-4} \end{cases} \begin{cases} k_{\sigma} = 0.0073 \\ b_{\sigma} = 0.0102 \end{cases}。$$

4.3 问题（2）的模型的改进与优化

在进行以上拟合过程中，对下图进行细致的观察。



在视频全程感知速率取值在 6000~7500 之间时，出现了明显的数据集中现象，但这些数据具有一定的规律，可以用 $\varepsilon(x_3)$ 来表示这个误差。经过 matlab 大量试验最终确定 $\varepsilon(x_3)=700$ （在多次试验中， $\varepsilon(x_3)$ 的值在 661~726 之间浮动，为了模型的简洁和美观，故取较为中间的数值 700）。最终优化后的模型表达式：

$$y_2 = \frac{k_t \frac{1}{\sqrt{2\pi\sigma(n)}} e^{-\frac{[(x_3 + \varepsilon(x_3)) - \mu(n)]^2}{2\sigma^2(n)}} + b_t}{\bar{t} - k_t \frac{1}{\sqrt{2\pi\sigma(n)}} e^{-\frac{[(x_3 + \varepsilon(x_3)) - \mu(n)]^2}{2\sigma^2(n)}} - b_t}$$

其中：

$$\begin{cases} \frac{1}{\mu} = k_{\mu}n + b_{\mu} \\ \frac{1}{\sqrt{\sigma}} = k_{\sigma}n + b_{\sigma} \end{cases}$$

$$\varepsilon(x_3) = \begin{cases} 700 & (6000 < x_3 < 7500) \\ 0 & \text{其他} \end{cases}$$

$$\bar{t} = 30000$$

$$\begin{cases} k_t = -6.67497 \\ b_t = 27327.69 \end{cases} \begin{cases} k_{\mu} = 5.096 \times 10^{-5} \\ b_{\mu} = 1.1991 \times 10^{-4} \end{cases} \begin{cases} k_{\sigma} = 0.0073 \\ b_{\sigma} = 0.0102 \end{cases}$$

4.4 模型（2）的检验

在求出卡顿时长占比与播放阶段平均速率的函数关系后，研究小组为了检验模型的准确性，提升整个建模过程的严谨性，编写了以下代码进行了模型（2）的检验，意在检测模型推测的数据与真实数据的拟合程度，代码如下所示

```
cy1=-6.67497*((0.0073*Llie+0.0102).^2/sqrt(2*pi)).*exp((-1)*((0.0073*Llie+0.0102).^4).*(Dlie-xx1).^2/2)+27327.69 %方便计算与检查，故将分母与分子分开，下列的 jiexi 才是 y2 最终的表达式
```

```
cy2=30000-((-6.67497*(0.0073*Llie+0.0102).^2/sqrt(2*pi)).*exp((-1)*((0.0073*Llie+0.0102).^4.*((Dlie-xx1).^2/2))-27327.69
```

```
jiexi=cy1./cy2
```

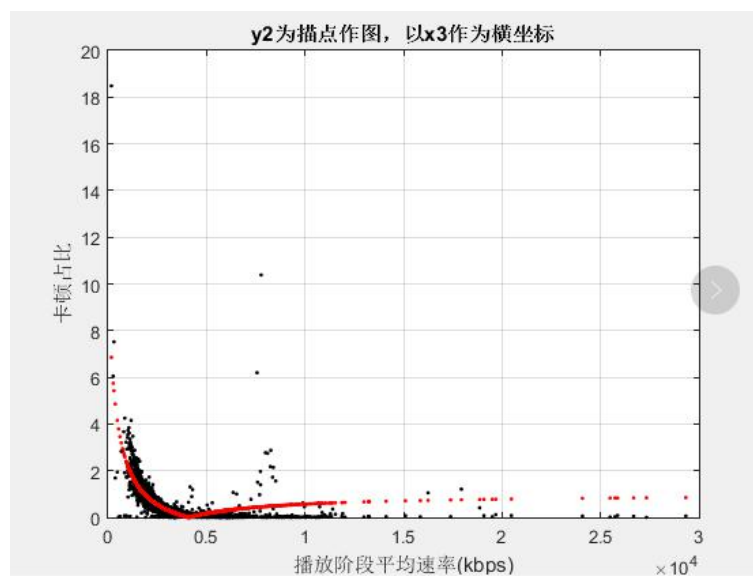
```
r=corrcoef(jiexi,Flie)
```

```
r =
```

```
1.0000    -0.9425  
-0.9425     1.0000
```

```
plot(Dlie,Flie,'k.',Dlie,jiexi,'r.') %本图形为描点作图  
xlabel('播放阶段平均速率(kbps)')  
ylabel('卡顿占比')  
grid on
```

结果表明，当排除卡顿次数为 0 时，通过研究小组求得的卡顿占比与实际样本数据中的卡顿占比的拟合程度达到了 94.25%，说明这个函数关系非常贴合实际数据。从研究小组所求得的关系图也可以得到这个信息。



六、参考文献

- [1]姜启源, 谢金星, 叶俊, 数学模型 (第四版), 高等教育出版社
- [2]卓金武, MATLAB 在数学建模中的应用, 北京航空航天大学出版社
- [3]刘保东, 宿洁, 陈建良, 数学建模基础教程, 高等教育出版社,
- [4]维基百科, TCP 机制, <https://en.wikipedia.org/wiki/TCP>
- [5]张大陆, 祝嘉麒, 网络传输中 IPTV 的 QoE 评估模型的研究[A], 计算机工程与应用, 2013, 49 (20): 71-76