# 根据摩拜单车出行轨迹画直方图

1752221 潘帅鑫

## 一，利用python处理数据

1.处理思路：

（1）.读入txt文件，建立数据list；

（2）.利用循环，计算出每个摩拜单车的出行时间，单位为s；

（3）.自定义函数，根据haversine公式，定义计算两个经纬度之间的实际距离的函数；

（4）.调用上述函数，计算出出行距离和速度

（5）.将出行时间，距离，速度合成list，转置一下

（6）.写入csv文件

2.具体代码（带有注释，便于读懂）

#!/usr/bin/env python

import csv

file = 'mobike\_reordered.txt'

bike\_list = []

with open(file, newline='') as f:

reader = csv.reader(f,delimiter=',',quotechar='"')

next(reader)

for row in reader:

order\_id = row[2]

bike\_id = row[3]

user\_id = row[4]

start\_time = float(row[0])

start\_time = int(start\_time)

end\_time = float(row[1])

end\_time = int(end\_time)

start\_lon = float(row[5])

end\_lon = float(row[7])

start\_lat = float(row[6])

end\_lat = float(row[8])

track = row[9]

trip\_info = [order\_id, bike\_id, user\_id, start\_time, end\_time, (start\_lon, start\_lat),(end\_lon, end\_lat), track]

bike\_list.append(trip\_info) #读入文件

use\_time = []

for i in range(len(bike\_list)):

use\_time.append(bike\_list[i][4] - bike\_list[i][3])#计算出行时间,单位为s

from math import radians,cos,sin,asin,sqrt

def haversine(lon1,lat1,lon2,lat2):#定义计算两经纬度之间的距离的函数

#将十进制数转化为弧度

lon1, lat1, lon2, lat2 = map(radians, [lon1, lat1, lon2, lat2])

#haversine公式

dlon = lon2 - lon1

dlat = lat2 - lat1

a = sin(dlat/2)\*\*2 + cos(lat1) \* cos(lat2) \* sin(dlon/2)\*\*2

c = 2 \* asin(sqrt(a))

r = 6371 #地球平均半径，单位为km

return c \* r \* 1000

path = []

for i in range(len(bike\_list)):

track\_list = bike\_list[i][7].split('#')#将str转为list

path\_t = 0

for j in range(len(track\_list)):

track\_list1 = track\_list[j].split(',')

track\_list1 = list(map(float,track\_list1))#将字符型转化为数值型

if j < len(track\_list)-1:#防止出现列表调用超出

track\_list2 = track\_list[j+1].split(',')

track\_list2 = list(map(float,track\_list2))

path\_f = haversine(track\_list1[0], track\_list1[1], track\_list2[0], track\_list2[1]) #调用函数计算相邻两经纬度之间的距离

path\_t += path\_f

path.append(path\_t) #计算出出发到目的地的距离，单位为m

speed = []

for i in range(len(bike\_list)):

speed\_f = (path[i] / use\_time[i]) \* 3.6

speed.append(speed\_f) #计算速度，单位为km/h

mobike\_list = [use\_time, path, speed]

mobike\_list = zip(\*mobike\_list) #转置为列

with open("mobike\_bike\_list.csv", 'w') as f:

writer = csv.writer(f, lineterminator='\n')

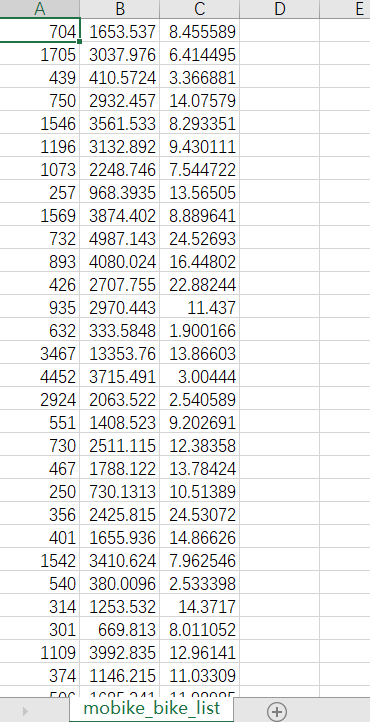
writer.writerows(mobike\_list) #写入csv文件

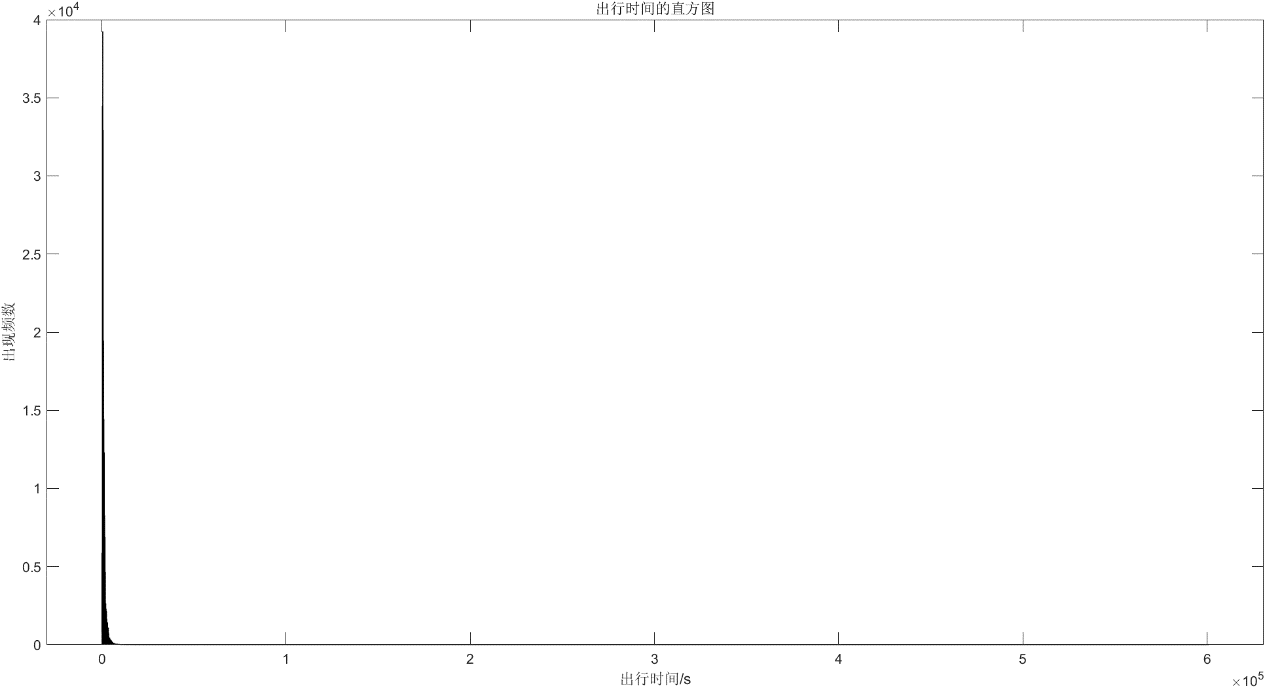
## 二，利用matlab画图

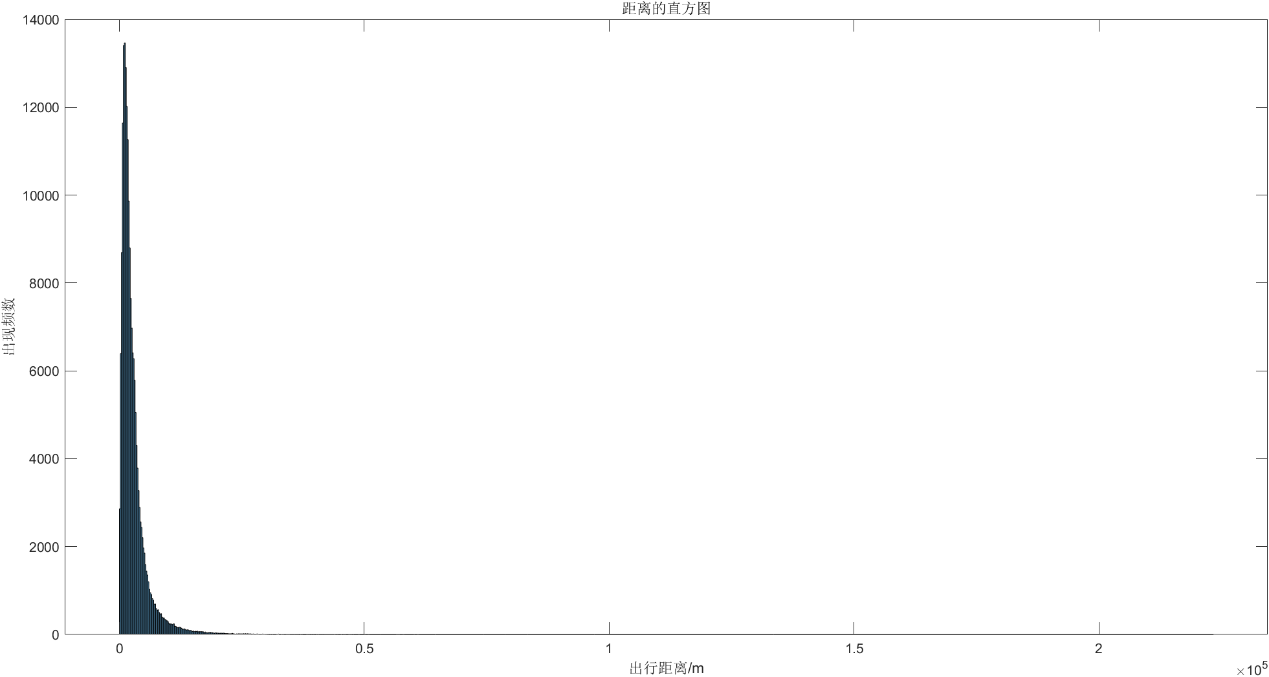
1.处理思路：

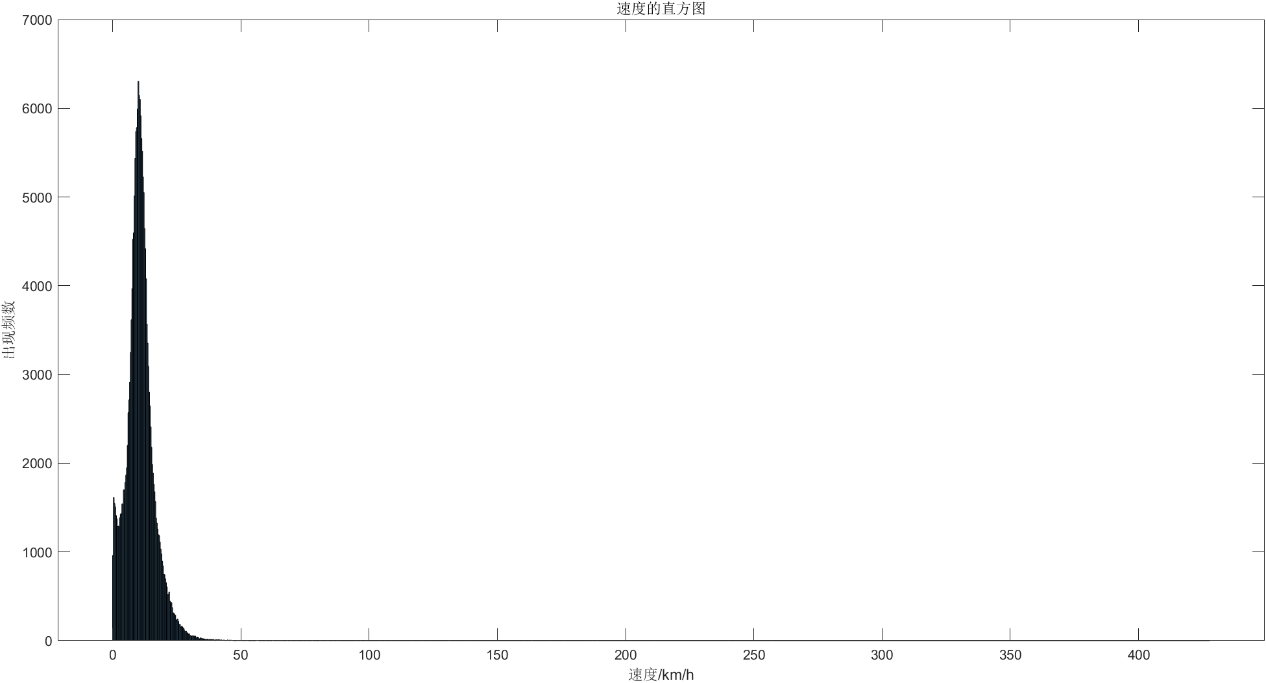
将上文输出的csv文件读入matlab中，利用histogram()函数，画出出行时间，距离和速度的直方图，

csv文件如下图（部分）：

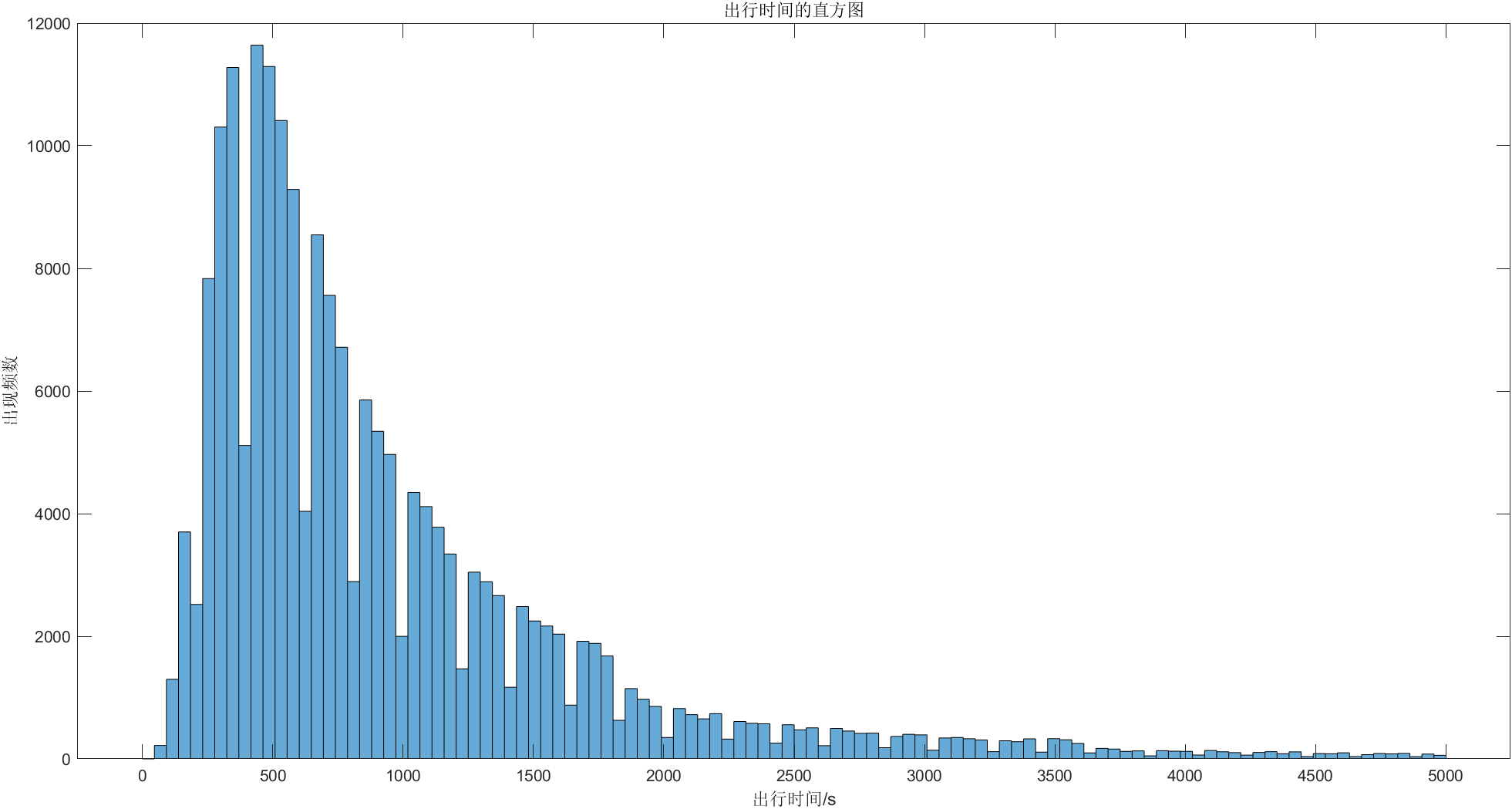


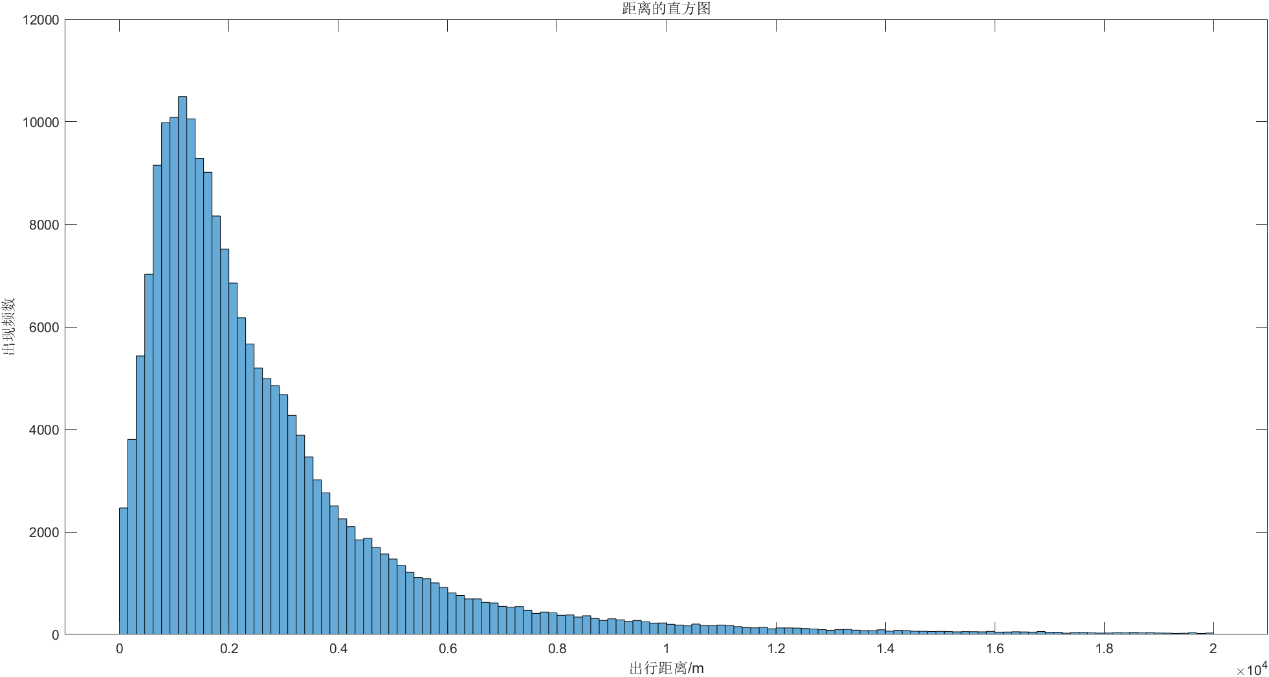


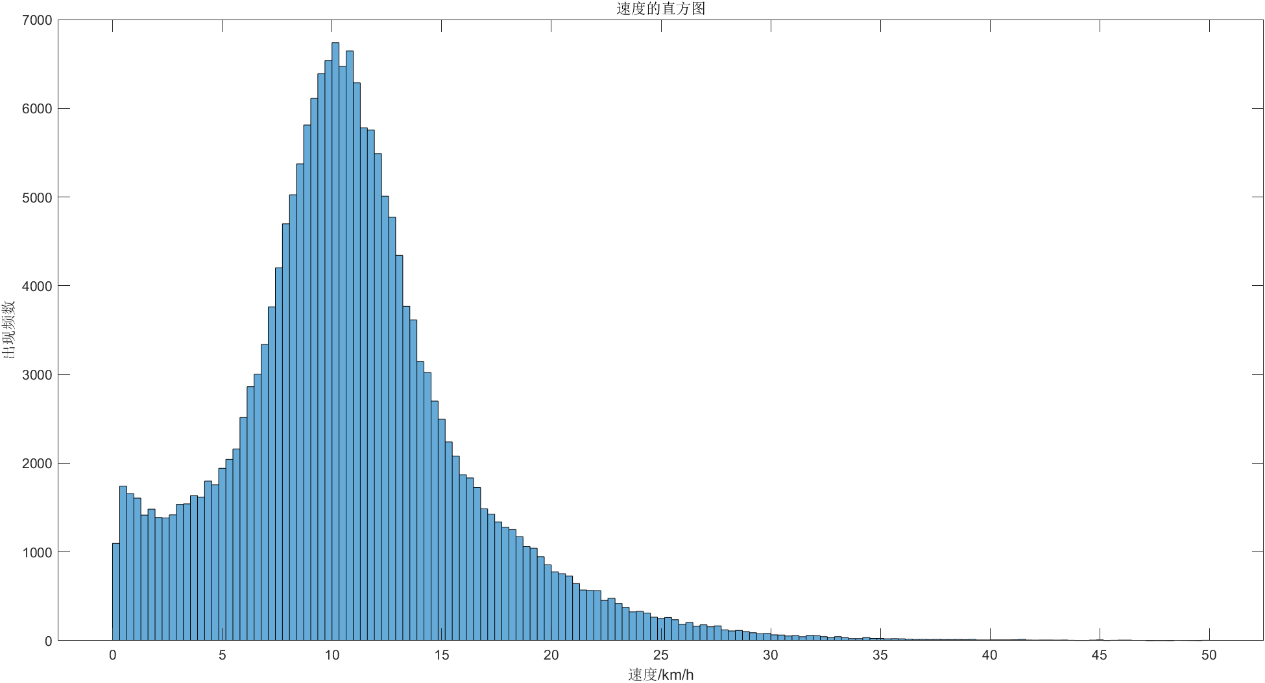




从上面的三个图可以看出，后面大部分的出现的频数很低，因此我们剔除频次过低的值，选取合适范围，重新作出了如下的图：







2.具体代码如下：

clear;clc;

A = readmatrix('mobike\_bike\_list.csv');

figure();

histogram(A(:,1));

xlabel('出行时间/s')

ylabel('出现频数')

title('出行时间的直方图')

figure();

histogram(A(:,2));

xlabel('出行距离/m')

ylabel('出现频数')

title('距离的直方图')

figure();

histogram(A(:,3));

xlabel('速度/km/h')

ylabel('出现频数')

title('速度的直方图')

figure();

histogram(A(:,1),'BinLimits',[0,5000]);

xlabel('出行时间/s')

ylabel('出现频数')

title('出行时间的直方图')

figure();

histogram(A(:,2),'BinLimits',[0,20000]);

xlabel('出行距离/m')

ylabel('出现频数')

title('距离的直方图')

figure();

histogram(A(:,3),'BinLimits',[0,50]);

xlabel('速度/km/h')

ylabel('出现频数')

title('速度的直方图')