Newyork_Taxi EDA

Roysen

12/14/2020

探索性数据分析-纽约出租车行驶数据

```
library(tidyverse)
library(forcats)
library(corrplot)
library(lubridate)
library(geosphere)
library(patchwork)
```

#读取数据

```
train <- read_csv("~/workspace/train.csv")
```

```
##
## ── Column specification ────────────────────────────────────────────────────────
## cols(
##   id = col_character(),
##   vendor_id = col_double(),
##   pickup_datetime = col_datetime(format = ""),
##   dropoff_datetime = col_datetime(format = ""),
##   passenger_count = col_double(),
##   pickup_longitude = col_double(),
##   pickup_latitude = col_double(),
##   dropoff_longitude = col_double(),
##   dropoff_latitude = col_double(),
##   store_and_fwd_flag = col_character(),
##   trip_duration = col_double()
## )
```

```
glimpse(train)
```

```
## Rows: 1,458,644
## Columns: 11
## $ id                <chr> "id2875421", "id2377394", "id3858529", "id3504673"…
## $ vendor_id         <dbl> 2, 1, 2, 2, 2, 2, 1, 2, 1, 2, 2, 2, 2, 2, 2, 1, 2,…
## $ pickup_datetime   <dttm> 2016-03-14 17:24:55, 2016-06-12 00:43:35, 2016-01…
## $ dropoff_datetime  <dttm> 2016-03-14 17:32:30, 2016-06-12 00:54:38, 2016-01…
## $ passenger_count   <dbl> 1, 1, 1, 1, 1, 6, 4, 1, 1, 1, 1, 4, 2, 1, 1, 1, 1,…
## $ pickup_longitude  <dbl> -73.98215, -73.98042, -73.97903, -74.01004, -73.97…
## $ pickup_latitude   <dbl> 40.76794, 40.73856, 40.76394, 40.71997, 40.79321, …
## $ dropoff_longitude <dbl> -73.96463, -73.99948, -74.00533, -74.01227, -73.97…
## $ dropoff_latitude  <dbl> 40.76560, 40.73115, 40.71009, 40.70672, 40.78252, …
## $ store_and_fwd_flag <chr> "N", "N", "N", "N", "N", "N", "N", "N", "N", "N", …
## $ trip_duration     <dbl> 455, 663, 2124, 429, 435, 443, 341, 1551, 255, 122…
```

##变量解释 |序号|变量|注释| |:—|:—|:—| | 1| id | ID| | 2| vendor_id | 出租车公司id| | 3| pickup_datetime | 上车时间| | 4| dropoff_datetime| 下车时间| | 5| passenger_count | 乘客人数| | 6| pickup_longitude| 上车经度| | 8| pickup_latitude | 上车纬度| | 9| dropoff_longitude | 下车经度| | 10| dropoff_latitude | 下车维度| | 11| store_and_fwd_flag| 是否分享行程记录 Y=是，N= 不| | 12| trip_duration | 旅行时间（秒)|

##案例分析 数据共有观测145万多行，变量12个,是一个非常大的数据集，抽取一个10000行的样本进行分析。 从11个变量的数据纬度来看，主要是关于纽约出租车用户出行时间、地点、人数，是否分享行程记录的数据，分析思路偏向用户画像分析。 *整体出行用户的时间和距离分布描述，并结合vendor_id查看两家公司的差异性* 不同乘客人数占比分析，看看出行用户中，单人还是多人出行人数最多 *描述用户行程记录分享数据* 计算速度，分析拥堵状况在全年、星期、每日的各自表现情况 *可结合地图包分析用户上车和下车主要集中在哪些区域，哪些区域拥堵，有没有躲避拥堵的有效方案

检查缺失值

```
train %>%
  summarise(
    across(everything(), ~sum(is.na(.)))
  )
```

```
## # A tibble: 1 x 11
##      id vendor_id pickup_datetime dropoff_datetime passenger_count
##   <int>     <int>           <int>            <int>           <int>
## 1     0         0               0                0               0
## # … with 6 more variables: pickup_longitude <int>, pickup_latitude <int>,
## #   dropoff_longitude <int>, dropoff_latitude <int>, store_and_fwd_flag <int>,
## #   trip_duration <int>
```

没有缺失值，太好了。

按行随机抽样10，000人

```
set.seed(1110)
test <- sample_n(train, 10000)
```

#提取经纬度变量，将经纬度转换为距离（km），并添加到数据框中

```
pickup_location <- test %>%
    select(pickup_longitude,pickup_latitude)
dropoff_location <- test %>%
    select(dropoff_longitude,dropoff_latitude)

test <- test %>%
    mutate(distance = distHaversine(pickup_location,dropoff_location)/1000)
```

```
test <- test %>%
  mutate(speed = distance/trip_duration*3600)
```

#日期格式转换，将vendor_id作为因子，添加速度（km/h）列

```
test <- test %>%
    mutate(store_and_fwd_flag = factor(store_and_fwd_flag),
           pickup_datetime = ymd_hms(pickup_datetime),
           dropoff_datetime = ymd_hms(dropoff_datetime),
           vendor_id = factor(vendor_id))

test
```

```
## # A tibble: 10,000 x 13
##    id    vendor_id pickup_datetime     dropoff_datetime    passenger_count
##    <chr> <fct>     <dttm>              <dttm>                        <dbl>
##  1 id26… 1         2016-03-08 19:50:57 2016-03-08 20:16:22               1
##  2 id23… 1         2016-04-02 18:23:41 2016-04-02 18:32:20               2
##  3 id16… 2         2016-01-14 11:36:37 2016-01-14 11:53:18               5
##  4 id30… 1         2016-06-08 13:36:04 2016-06-08 13:40:53               1
##  5 id09… 2         2016-03-18 03:39:47 2016-03-18 03:52:46               1
##  6 id02… 1         2016-06-29 22:33:36 2016-06-29 23:10:26               1
##  7 id11… 1         2016-05-20 14:19:17 2016-05-20 14:21:35               1
##  8 id21… 2         2016-02-23 05:57:36 2016-02-23 06:12:57               1
##  9 id11… 1         2016-05-01 02:24:13 2016-05-01 02:29:51               1
## 10 id15… 2         2016-05-07 03:21:26 2016-05-07 03:25:31               1
## # … with 9,990 more rows, and 8 more variables: pickup_longitude <dbl>,
## #   pickup_latitude <dbl>, dropoff_longitude <dbl>, dropoff_latitude <dbl>,
## #   store_and_fwd_flag <fct>, trip_duration <dbl>, distance <dbl>, speed <dbl>
```

#日期处理，将日期转换为数据型，并按年月周日拆分日期

```
test <- test %>%
    mutate(month = as.integer(month(pickup_datetime)),
           hour = hour(pickup_datetime),
           wday = wday(pickup_datetime))

summary(test)
```

```
##       id            vendor_id pickup_datetime
##  Length:10000       1:4573    Min.   :2016-01-01 00:07:29
##  Class :character   2:5427    1st Qu.:2016-02-16 12:26:52
##  Mode  :character             Median :2016-03-31 10:47:35
##                               Mean   :2016-03-31 20:58:20
##                               3rd Qu.:2016-05-15 17:13:56
##                               Max.   :2016-06-30 23:47:52
##  dropoff_datetime              passenger_count pickup_longitude pickup_latitude
##  Min.   :2016-01-01 00:16:03   Min.   :1.000   Min.   :-74.11   Min.   :40.60
##  1st Qu.:2016-02-16 12:34:48   1st Qu.:1.000   1st Qu.:-73.99   1st Qu.:40.74
##  Median :2016-03-31 11:07:38   Median :1.000   Median :-73.98   Median :40.75
##  Mean   :2016-03-31 21:13:21   Mean   :1.666   Mean   :-73.97   Mean   :40.75
##  3rd Qu.:2016-05-15 17:20:53   3rd Qu.:2.000   3rd Qu.:-73.97   3rd Qu.:40.77
##  Max.   :2016-07-01 00:00:26   Max.   :6.000   Max.   :-73.59   Max.   :40.88
##  dropoff_longitude dropoff_latitude store_and_fwd_flag trip_duration
##  Min.   :-74.28    Min.   :40.55    N:9958             Min.   :    3.0
##  1st Qu.:-73.99    1st Qu.:40.74    Y:  42             1st Qu.:  397.0
##  Median :-73.98    Median :40.75                       Median :  671.0
##  Mean   :-73.97    Mean   :40.75                       Mean   :  900.4
##  3rd Qu.:-73.96    3rd Qu.:40.77                       3rd Qu.: 1089.0
##  Max.   :-73.59    Max.   :40.97                       Max.   :86216.0
##     distance          speed            month            hour
##  Min.   : 0.000   Min.   :  0.00   Min.   :1.000   Min.   : 0.00
##  1st Qu.: 1.234   1st Qu.:  9.10   1st Qu.:2.000   1st Qu.: 9.00
##  Median : 2.110   Median : 12.79   Median :3.000   Median :14.00
##  Mean   : 3.421   Mean   : 14.32   Mean   :3.496   Mean   :13.62
##  3rd Qu.: 3.885   3rd Qu.: 17.65   3rd Qu.:5.000   3rd Qu.:19.00
##  Max.   :31.914   Max.   :283.16   Max.   :6.000   Max.   :23.00
##       wday
##  Min.   :1.00
##  1st Qu.:2.00
##  Median :4.00
##  Mean   :4.15
##  3rd Qu.:6.00
##  Max.   :7.00
```

##查看有无异常值

```
test %>%
  mutate(h_trip = trip_duration/3600) %>%
  select(distance, speed, h_trip) %>%
  arrange(-speed,distance, h_trip)
```

```
## # A tibble: 10,000 x 3
##    distance speed  h_trip
##       <dbl> <dbl>   <dbl>
##  1   0.315  283.  0.00111
##  2   1.01    84.9 0.0119
##  3   0.0931  83.8 0.00111
##  4   0.183   73.3 0.0025
##  5   0.130   66.8 0.00194
##  6  10.2     66.6 0.153
##  7  15.8     64.8 0.243
##  8  20.0     60.2 0.332
##  9   2.33    59.0 0.0394
## 10   6.51    57.1 0.114
## # … with 9,990 more rows
```

```
test %>%
  mutate(h_trip = trip_duration/3600) %>%
  select(distance, speed, h_trip) %>%
  arrange(-h_trip, speed, distance)
```

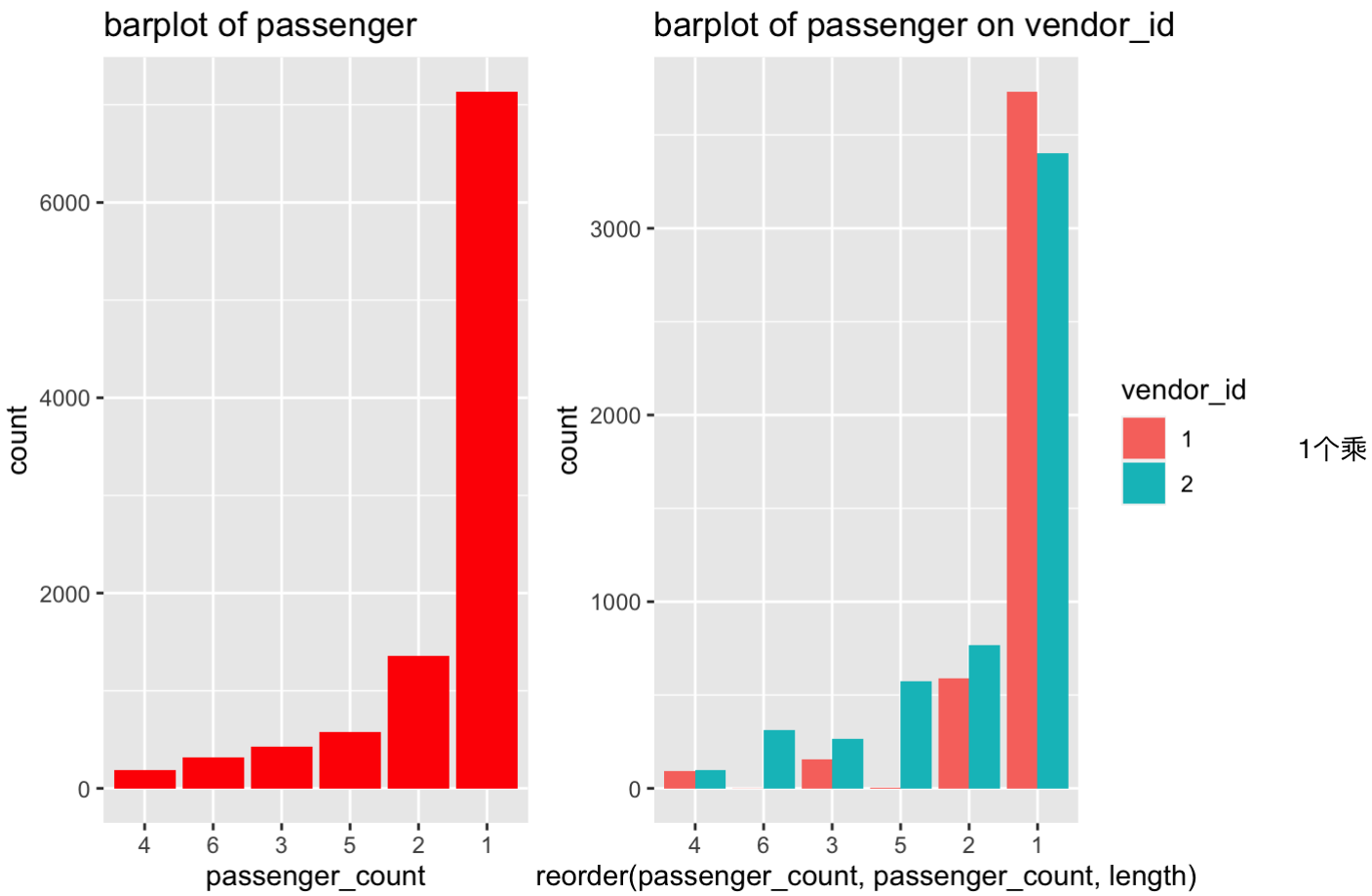
```
## # A tibble: 10,000 x 3
##    distance  speed h_trip
##       <dbl>  <dbl>  <dbl>
##  1   0.989  0.0413  23.9
##  2   0.968  0.0405  23.9
##  3   1.33   0.0560  23.8
##  4   2.83   0.119   23.8
##  5   8.48   0.358   23.7
##  6   1.12   0.0473  23.6
##  7   5.44   0.232   23.5
##  8   0.260  0.0711   3.66
##  9   0.0432 0.0135   3.20
## 10   9.96   5.21     1.91
## # … with 9,990 more rows
```

```
test1 <- test %>%
  mutate(h_trip = trip_duration/3600) %>%
  filter(h_trip < 23 & speed < 280)
```

速度大于280km/h几乎不可能，行驶时长超过23h可能性也不大。因而删除掉这部分数据。

##乘客人数分析

```
p1 <- test1 %>%
ggplot(aes(passenger_count, fill = I("red"))) +
    geom_bar(aes(x = reorder(passenger_count, passenger_count, length))) +
    labs(title = "barplot of passenger")
p2 <- test1 %>%
ggplot(aes(x = reorder(passenger_count, passenger_count, length), fill = vendor_id)) +
    geom_bar( position = "dodge") +
    labs(title = "barplot of passenger on vendor_id ")
p1+p2
```

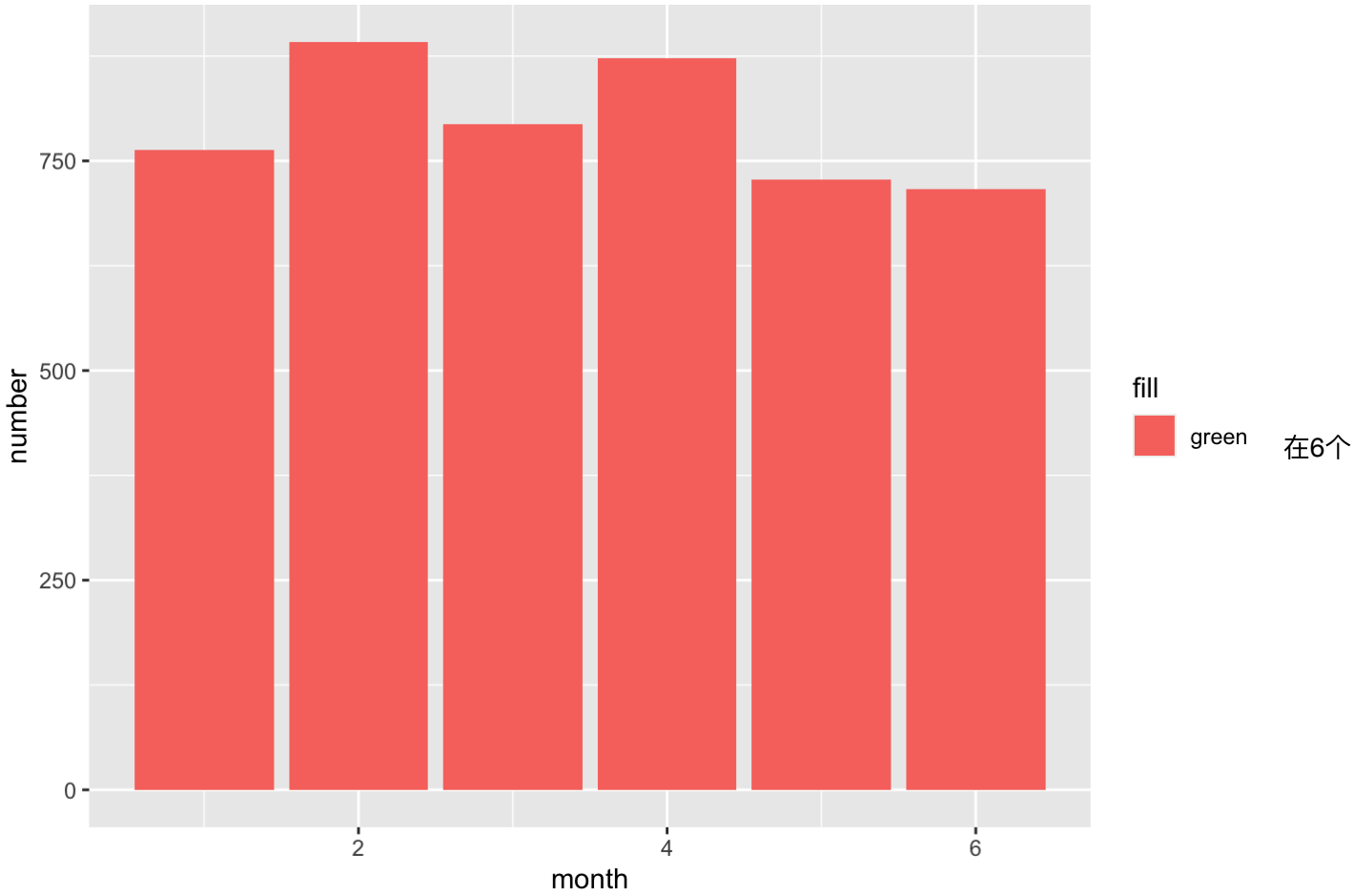


客的人数超过7000人，2个乘客的人数约1300人，4人最少，5人和6人的乘客出行人数排在第三名。 从两家公司来看，出行乘客人数数据相差不大，仅仅只有第二组公司有5人和六人的乘客，或许是因为只有他们在做多人出行业务

#多人出行分组统计

```
filter(test1, passenger_count>4) %>%
    group_by(month) %>%
    summarise(number = sum(passenger_count)) %>%
  ggplot(aes(month, number, fill = "green"))+
  geom_histogram(stat = "identity")
```

```
## Warning: Ignoring unknown parameters: binwidth, bins, pad
```



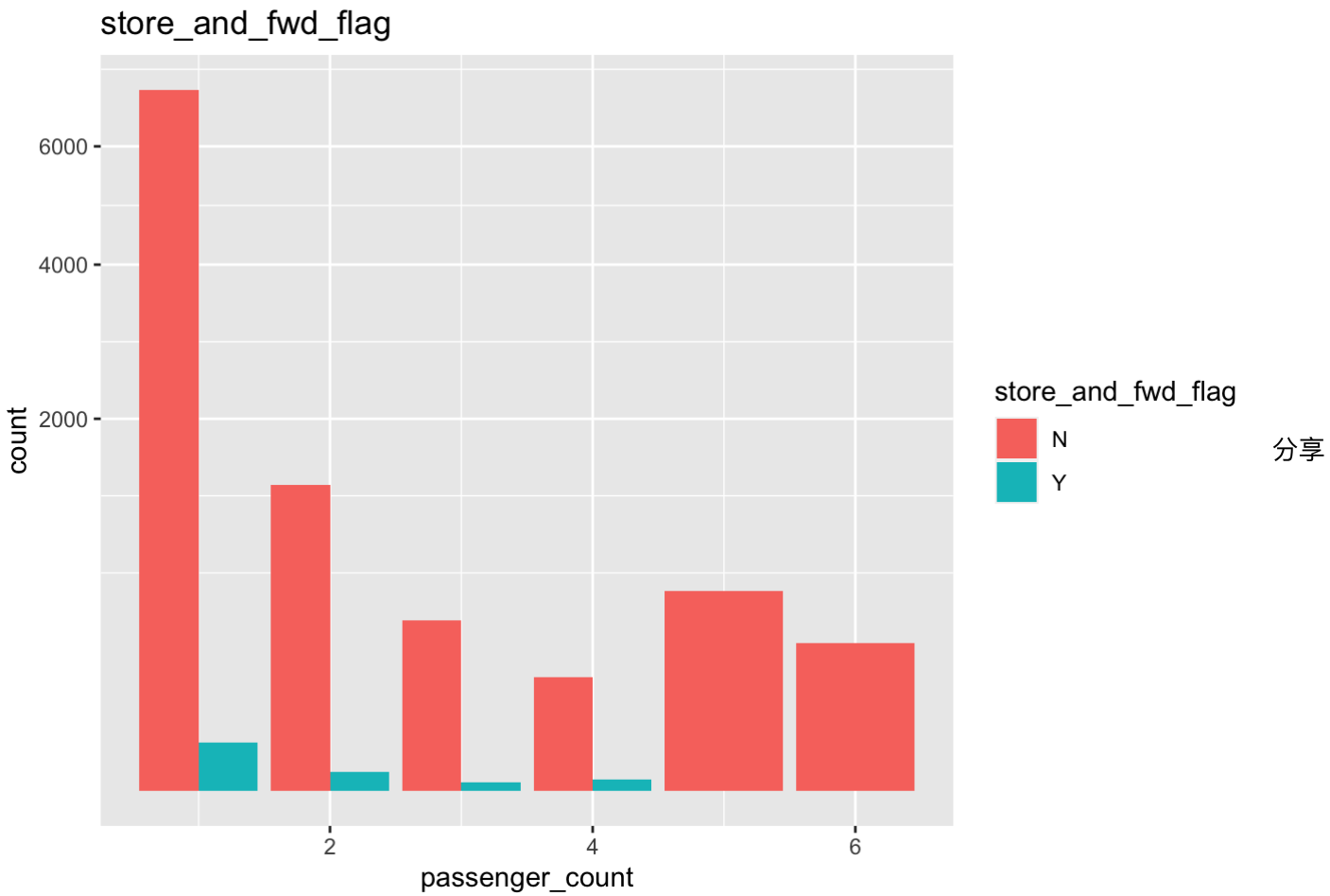
月中5人和6人乘车人数比较平稳

#转发行程人数远小于100人，其中6人乘客无人转发行程

```
test1 %>%
  filter(store_and_fwd_flag == "Y") %>%
  count()
```

```
## # A tibble: 1 x 1
##       n
##   <int>
## 1    42
```

```
test1 %>%
ggplot(aes(passenger_count, fill = store_and_fwd_flag))+
    geom_bar(position = "dodge")+
    labs(title = "store_and_fwd_flag")+
    scale_y_sqrt()
```



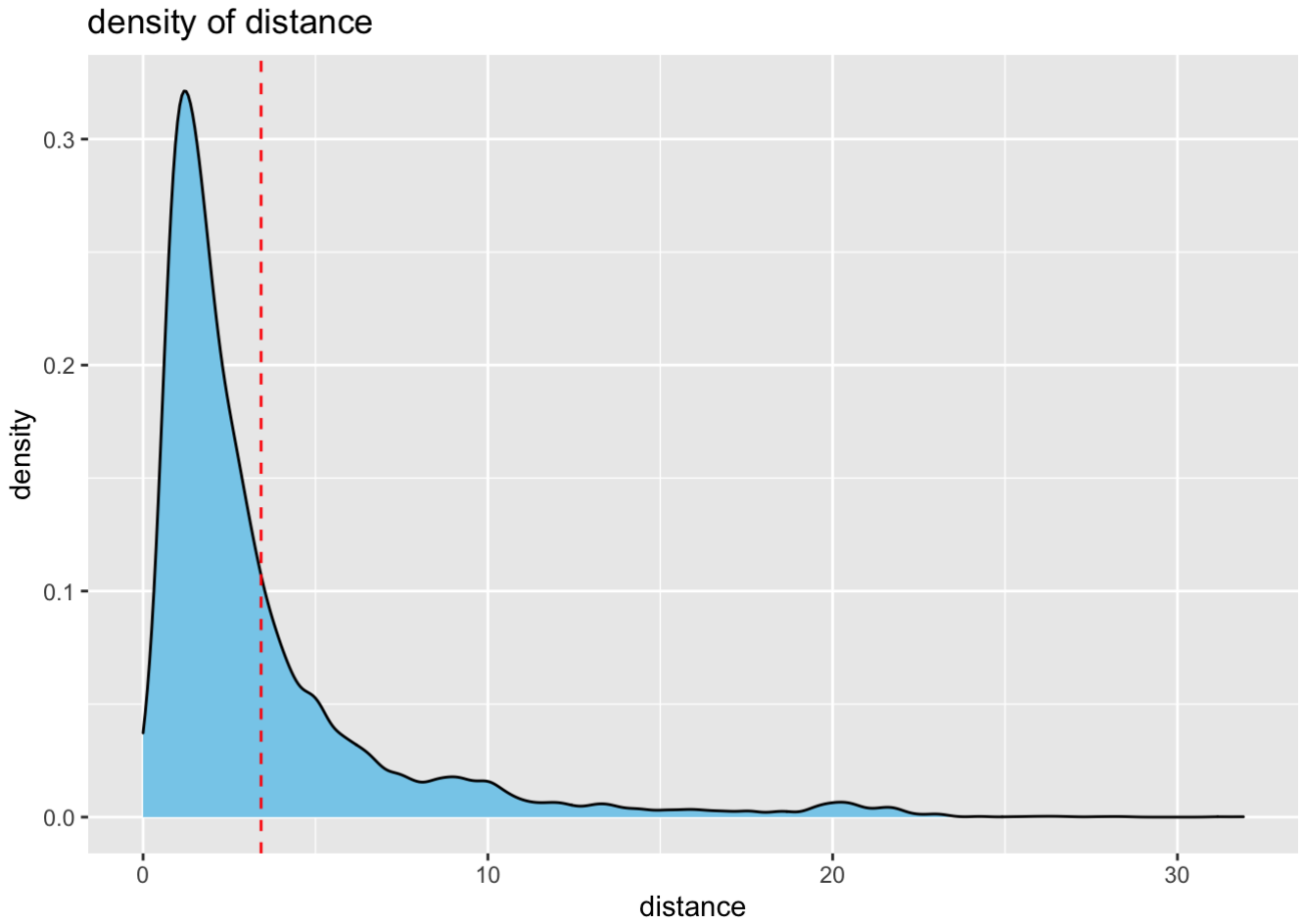
行程的人数只有42人，这42人分布在乘客人数4人以下的组中，说明大部分乘客还是不愿意分享自己的行程数据。

#行驶距离（可能服从右偏的正态分布）

```
test1 %>%
  select(distance) %>%
  summarise(mean = mean(distance),
            sd = sd(distance))
```

```
## # A tibble: 1 x 2
##    mean    sd
##   <dbl> <dbl>
## 1  3.42  3.83
```

```
true.mean <- 3.421247
true.sd <- 3.829957
test1 %>%
ggplot(aes(distance))+
    geom_density(fill = "skyblue")+
    geom_vline(
    xintercept = true.mean,
    color = "red",
    linetype = "dashed"
  )+
    labs(title = "density of distance")
```



样本

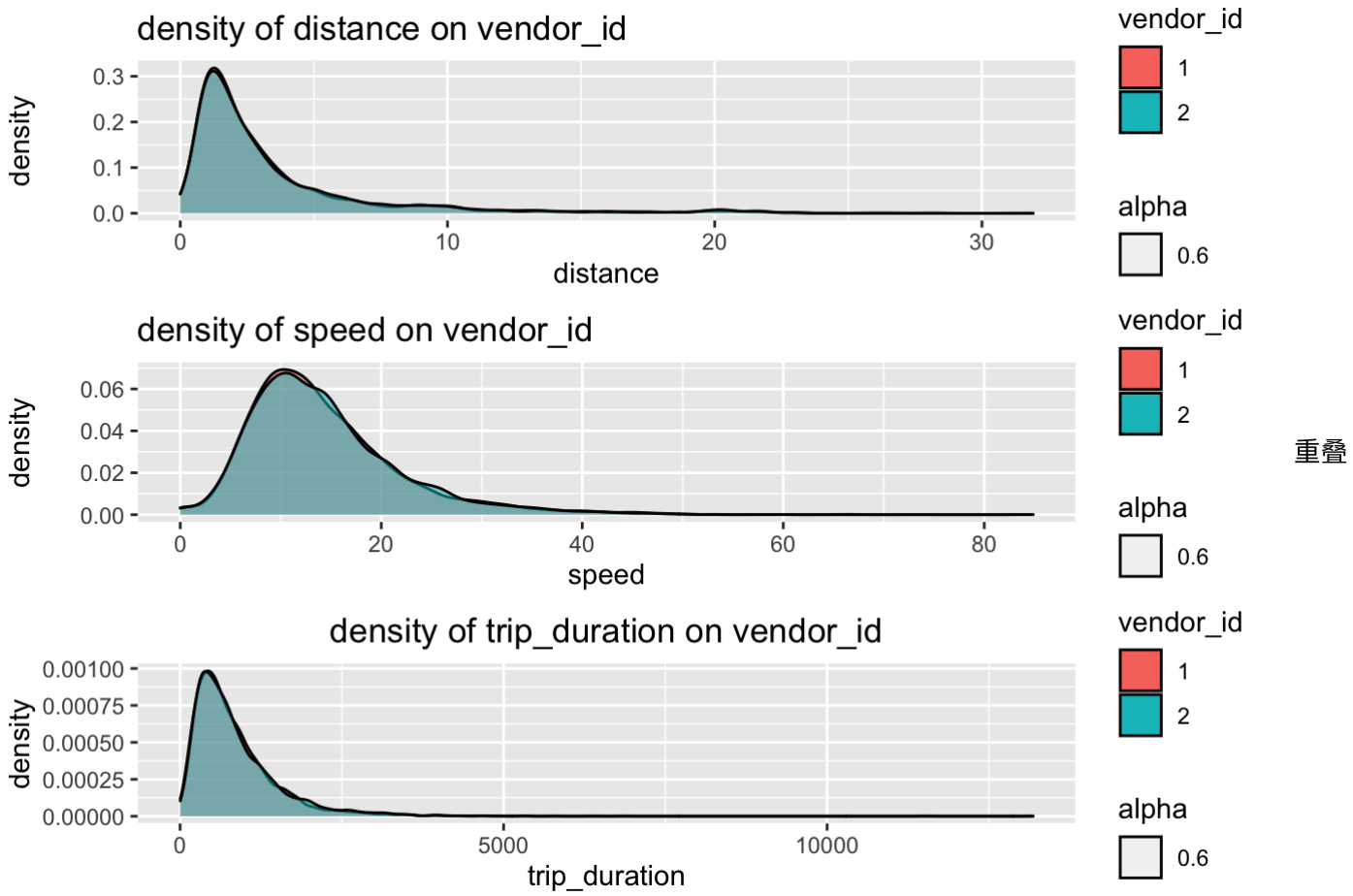
均值为3.42km，分布呈右偏，远距离极大的拉高了均值，大部分用户的行驶距离没有超过3.42km。

#对出租车公司进行分组，查看行驶距离是否有差异

```
p1 <- test1 %>%
ggplot(aes(distance, fill = vendor_id, alpha = 0.6))+
    geom_density()+
    labs(title = "density of distance on vendor_id")

p2 <- test1 %>%
ggplot(aes(speed, fill = vendor_id, alpha = 0.6))+
    geom_density()+
    labs(title = "density of speed on vendor_id")

p3 <- test1 %>%
ggplot(aes(trip_duration, fill = vendor_id, alpha = 0.6))+
    geom_density()+
    labs(title = "density of trip_duration on vendor_id")+
    theme(plot.title = element_text(hjust = 0.5))
p1/p2/p3
```



重叠

非常严重，可以看出两家出租车公司在行驶距离、速度和旅行时长方面差异不大，短途用车和拥堵问题是一个共性的情况。

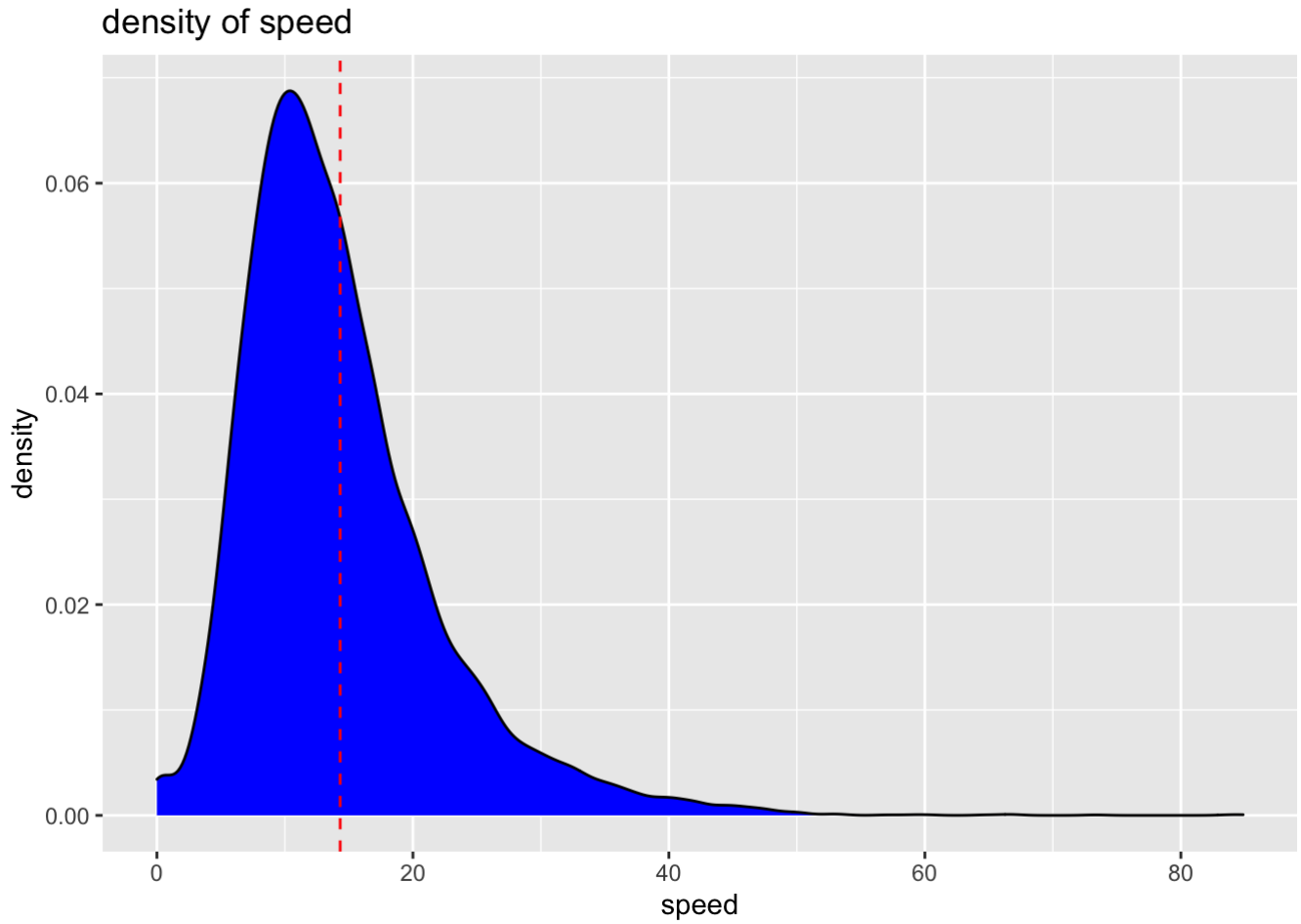
#上下车时间大体正常，1月底左右无人打车，1组和2组分布差异不大

```
test1 %>%
  select(speed) %>%
  summarise(true.mean = mean(speed))
```

```
## # A tibble: 1 x 1
##   true.mean
##       <dbl>
## 1      14.3
```

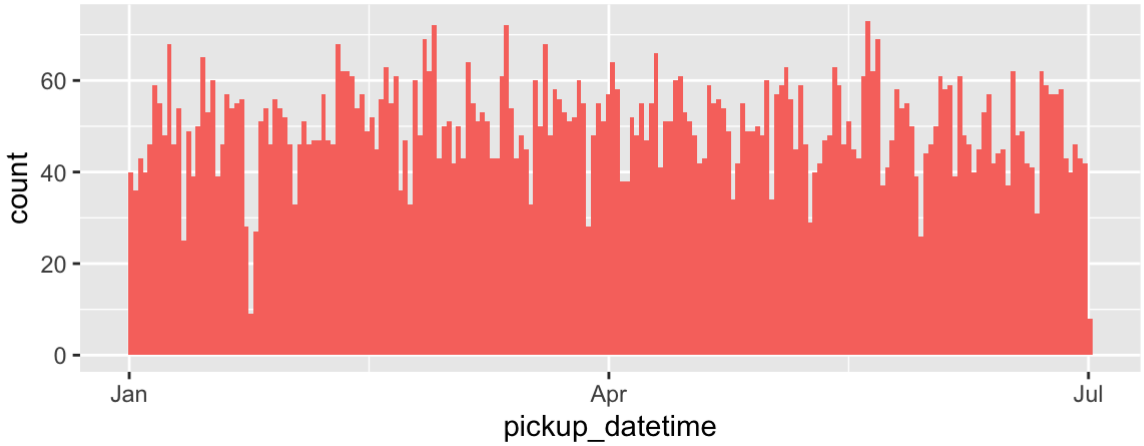
```
true.mean = 14.31743

test1 %>%
ggplot(aes(speed))+
    geom_density( fill = "blue")+
    geom_vline(
    xintercept = true.mean,
    color = "red",
    linetype = "dashed")+
    labs(title = "density of speed ")
```

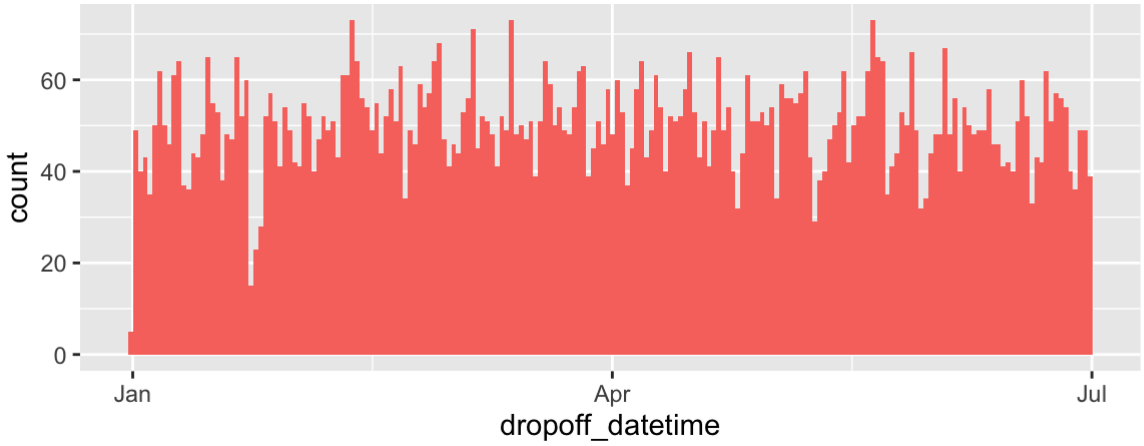


#上下时间对比

```
p1 <- ggplot(test1,aes(pickup_datetime,fill = "red"))+
    geom_histogram( bins = 200)
p2 <- ggplot(test1,aes(dropoff_datetime,fill = "red"))+
    geom_histogram(bins = 200)
p1/p2
```



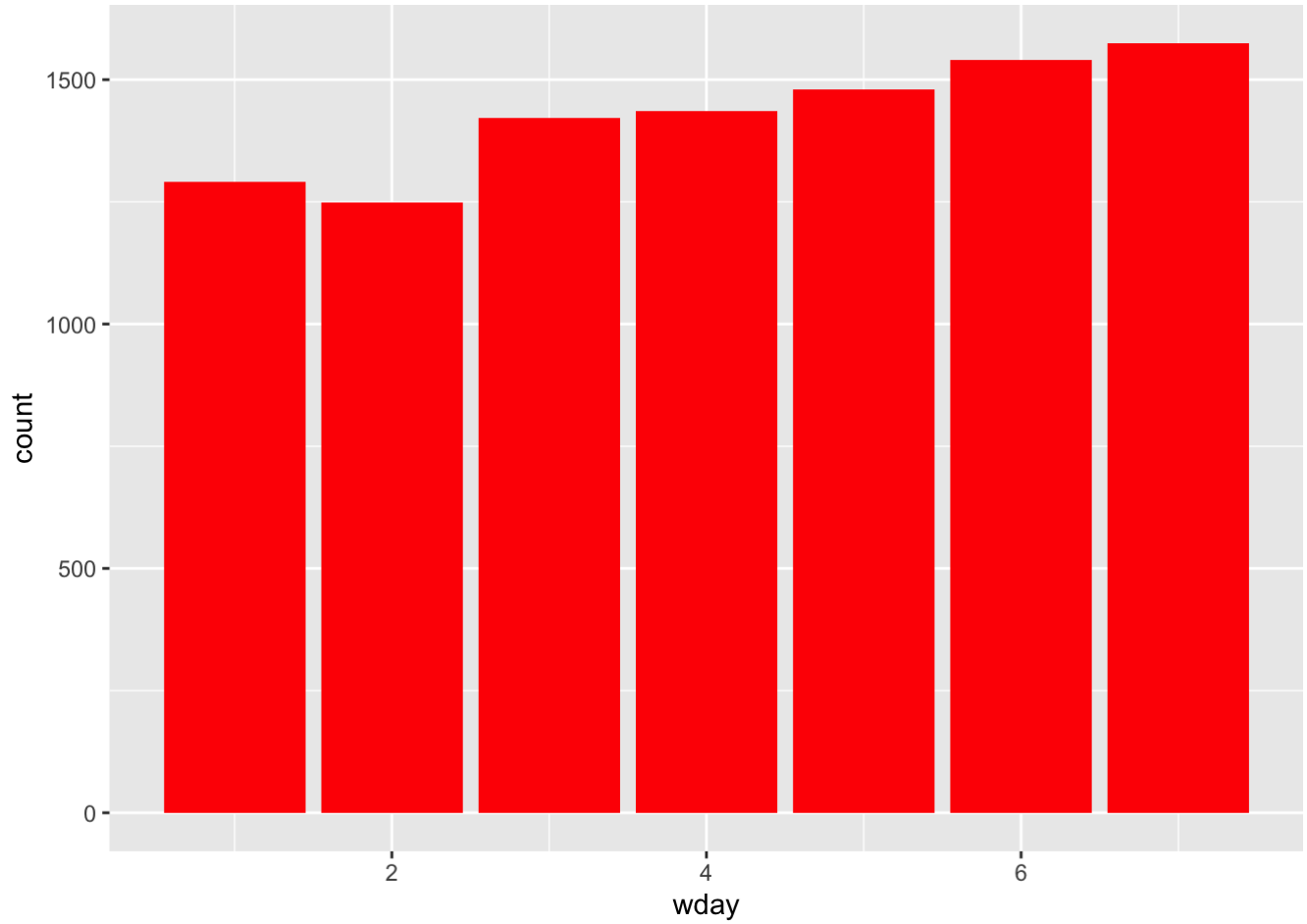
上车



和对应的下车时间大体分布是均匀的，奇怪的是一月底二月初打车人很少，谷歌显示因为城市遭遇暴风雪

##看看从周一到周日打车人数的变化

```
test1 %>%
    ggplot(aes(wday))+
    geom_bar(fill = "red")
```

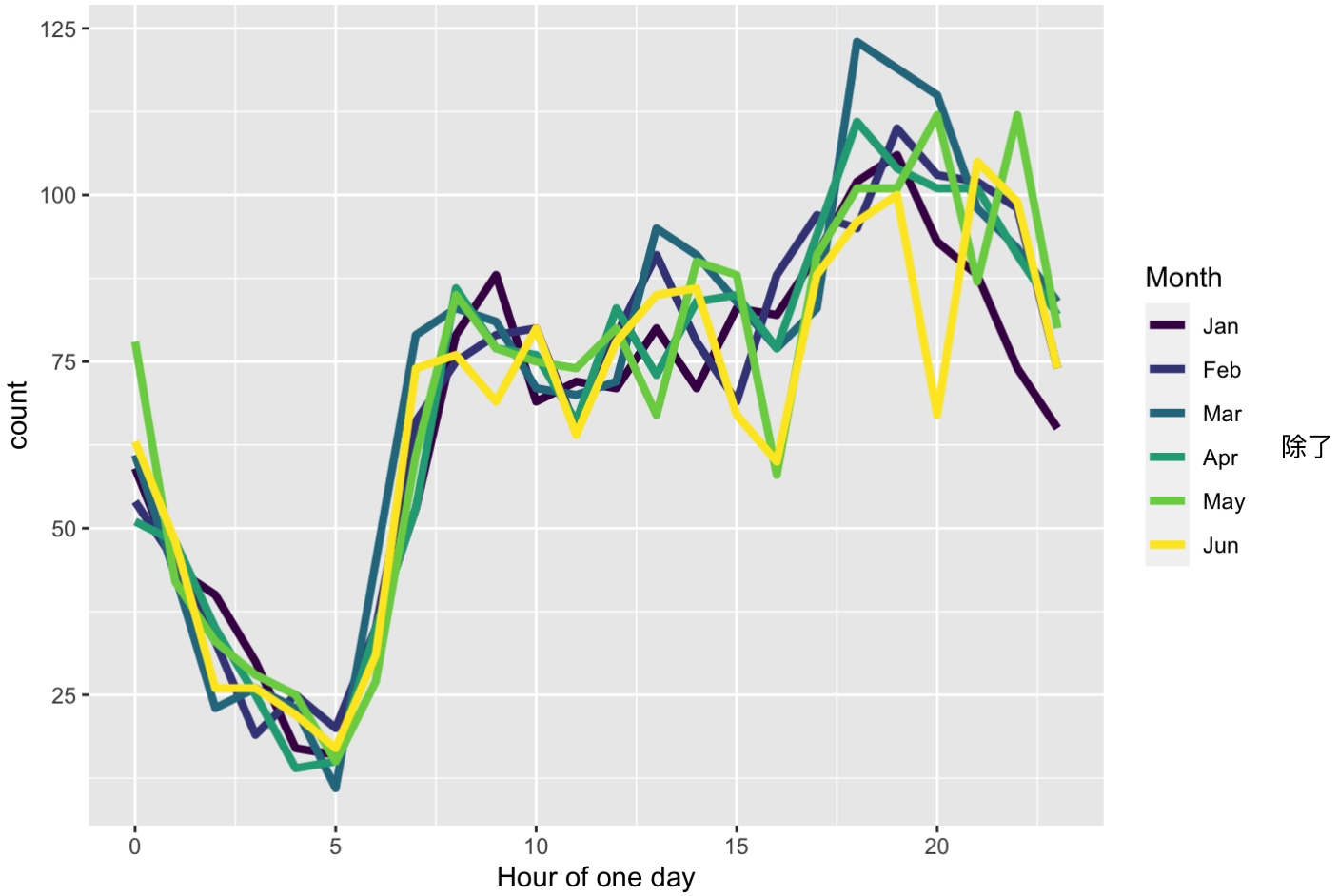


从周

一天到周日打车人数变化不大，周一和周二人数相对少些

##看看一年中每天的乘客人数分布

```
test1 %>%
  mutate(hpick = hour(pickup_datetime),
         Month = factor(month(pickup_datetime, label = TRUE))) %>%
group_by(hpick, Month) %>%
  count() %>%
  ggplot(aes(hpick, n, color = Month)) +
  geom_line(size = 1.5) +
  labs(x = "Hour of one day", y = "count")
```

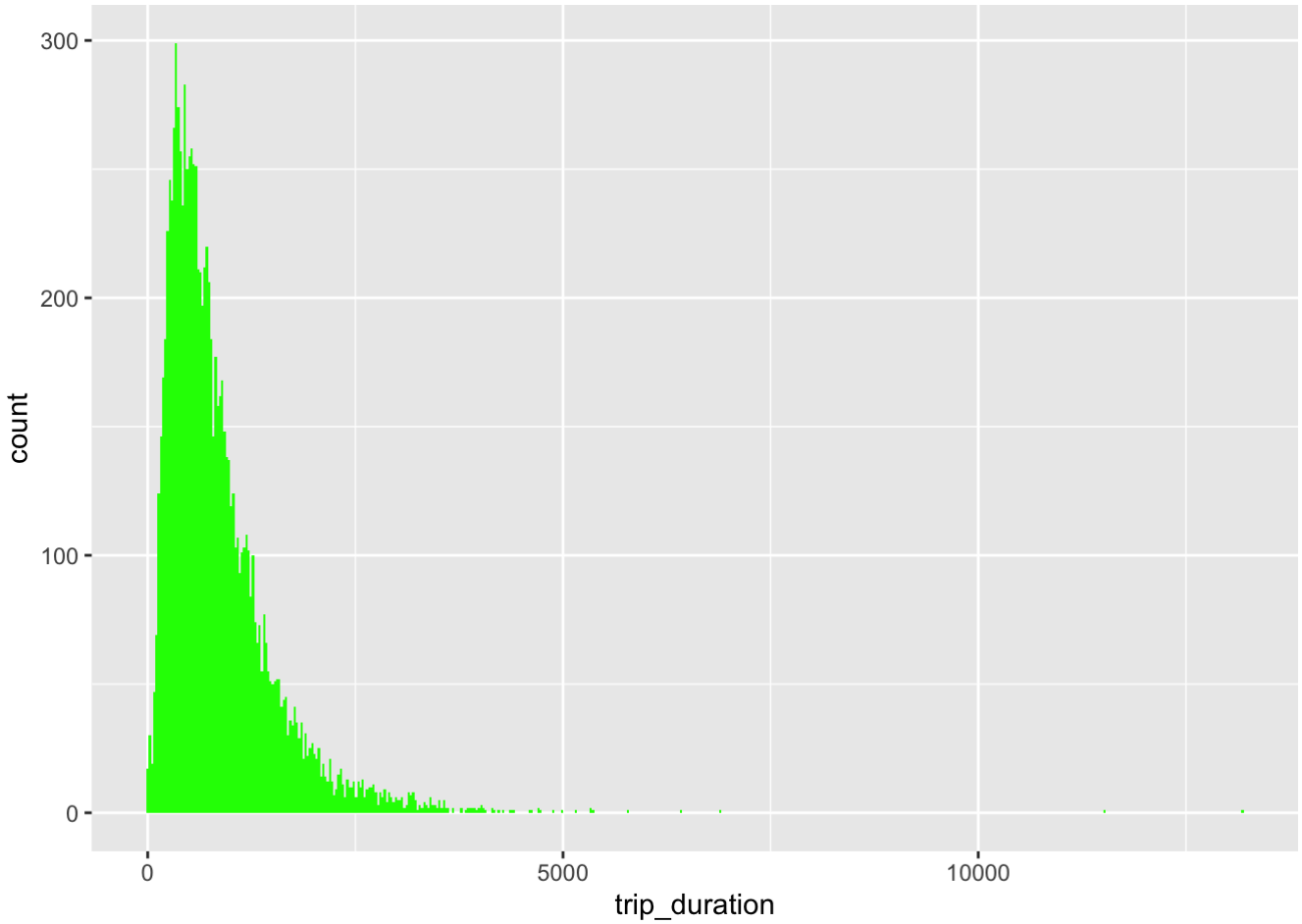


除了

凌晨（2:00-6:00）都是高峰，晚19:00-21:00为打车人数最多的时间，夜生活丰富

#打车时长

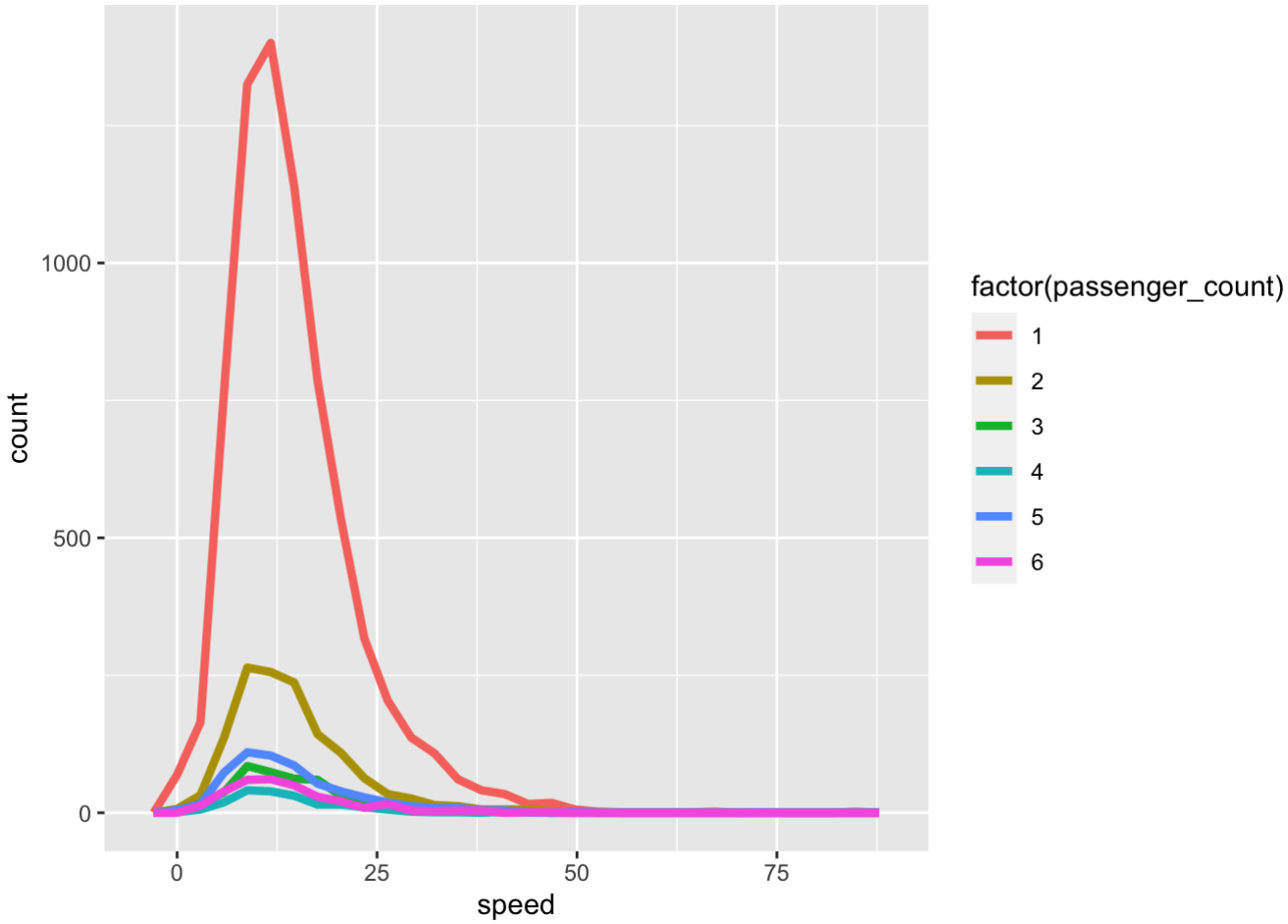
```
test1 %>%
ggplot(aes(trip_duration))+
    geom_histogram(bins = 500,fill = "green")
```



#车速与乘客人数、月份、周无关，与每日具体几点钟有关（凌晨车少，车速快）

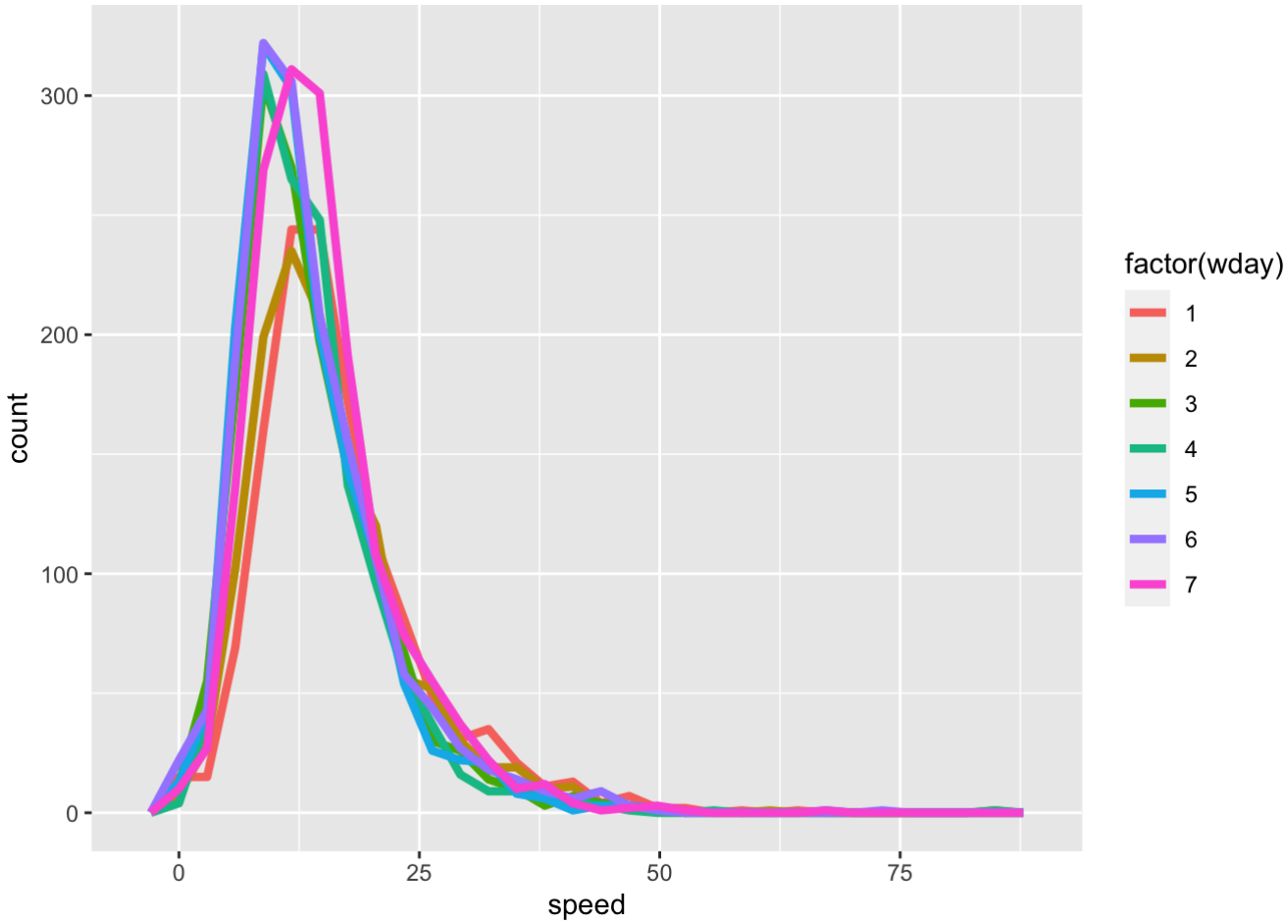
```
test1 %>%
    ggplot(aes(speed,color = factor(passenger_count)))+
    geom_freqpoly(size = 1.5)
```

```
## `stat_bin()` using `bins = 30`. Pick better value with `binwidth`.
```



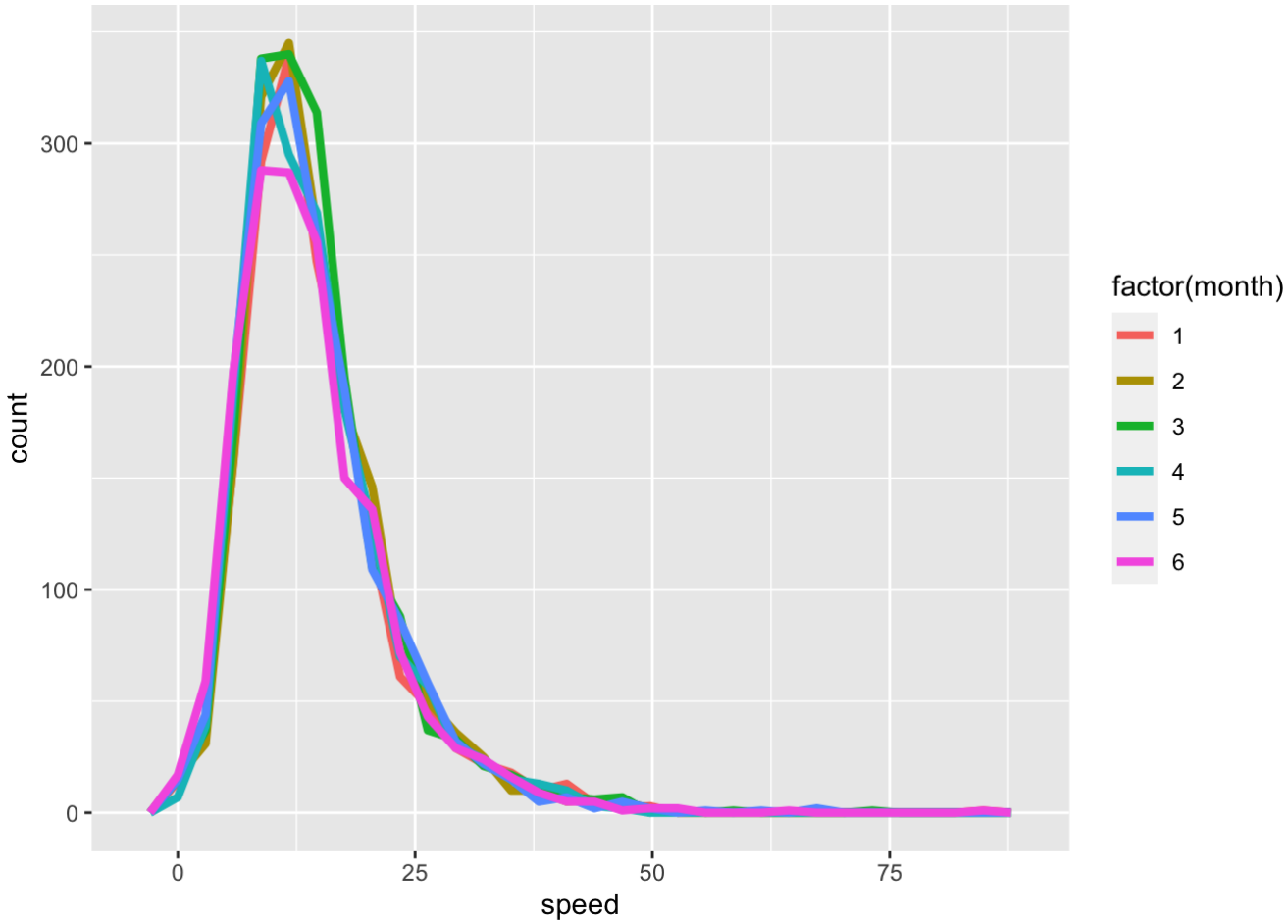

```
test1 %>%
    ggplot(aes(speed,color = factor(wday)))+
    geom_freqpoly(size = 1.5)
```

```
## `stat_bin()` using `bins = 30`. Pick better value with `binwidth`.
```



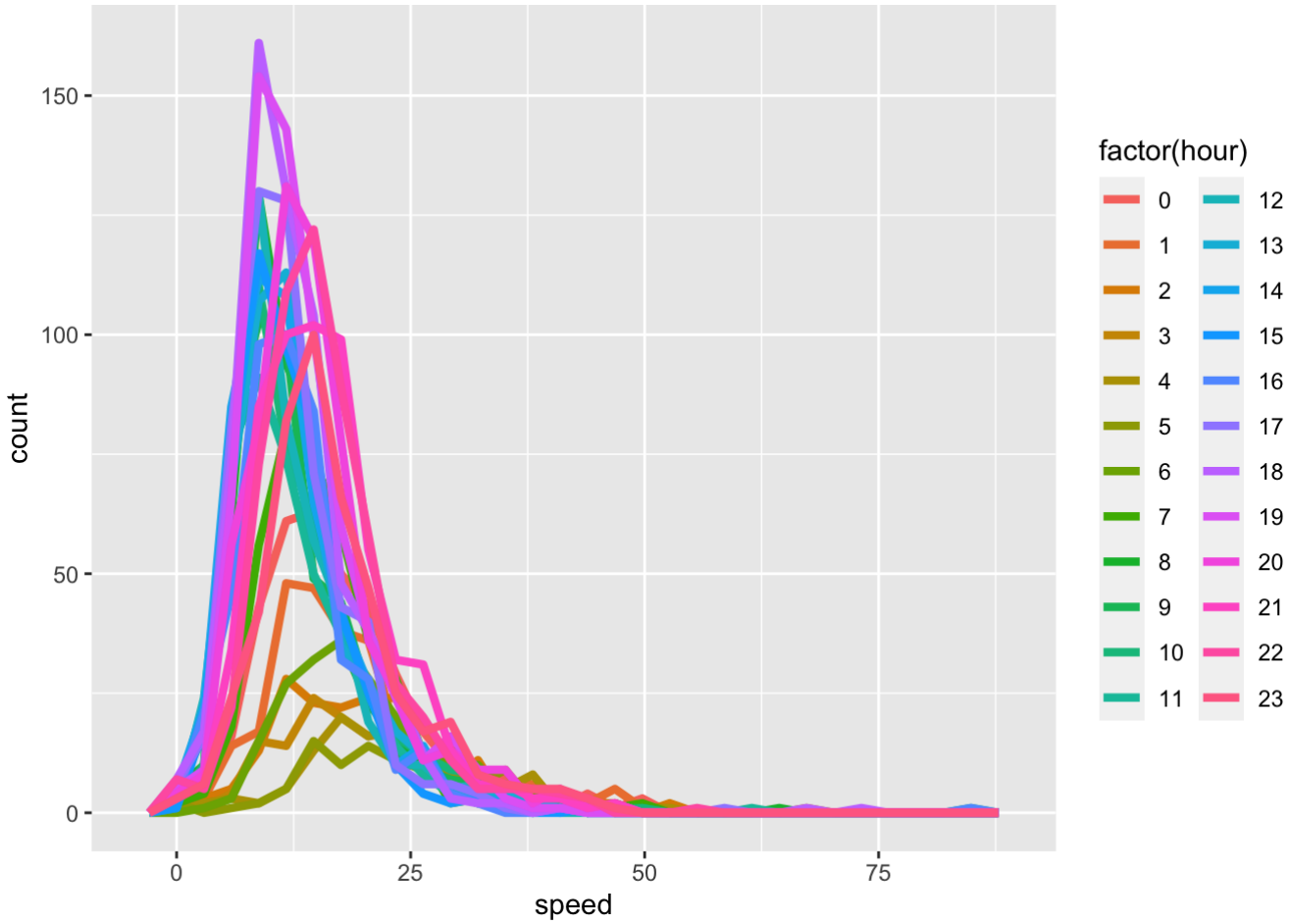

```
test1 %>%
    ggplot(aes(speed,color = factor(month)))+
    geom_freqpoly(size = 1.5)
```

```
## `stat_bin()` using `bins = 30`. Pick better value with `binwidth`.
```

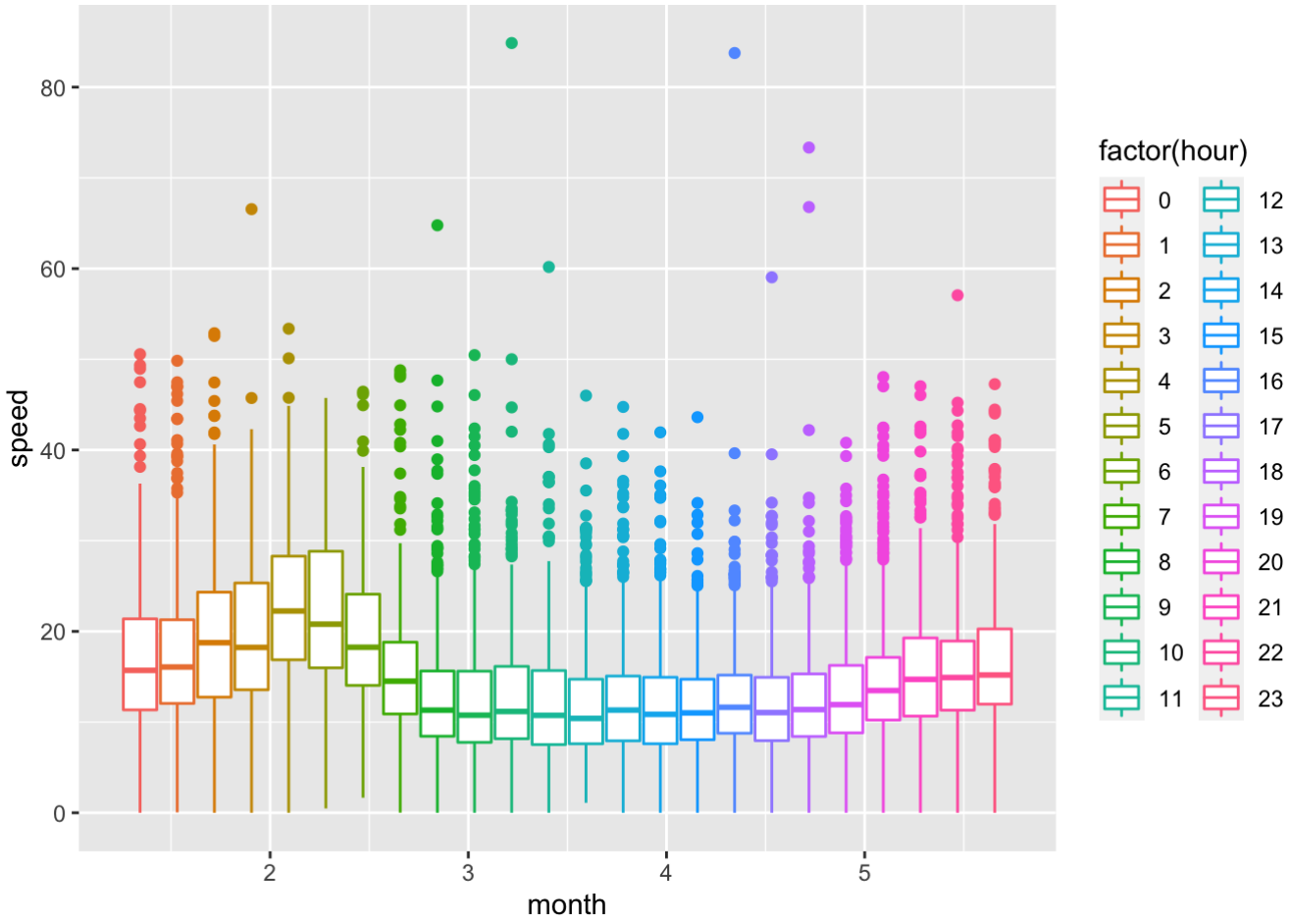



```
test1 %>%
    ggplot(aes(speed,color = factor(hour)))+
    geom_freqpoly(size = 1.5)
```

```
## `stat_bin()` using `bins = 30`. Pick better value with `binwidth`.
```

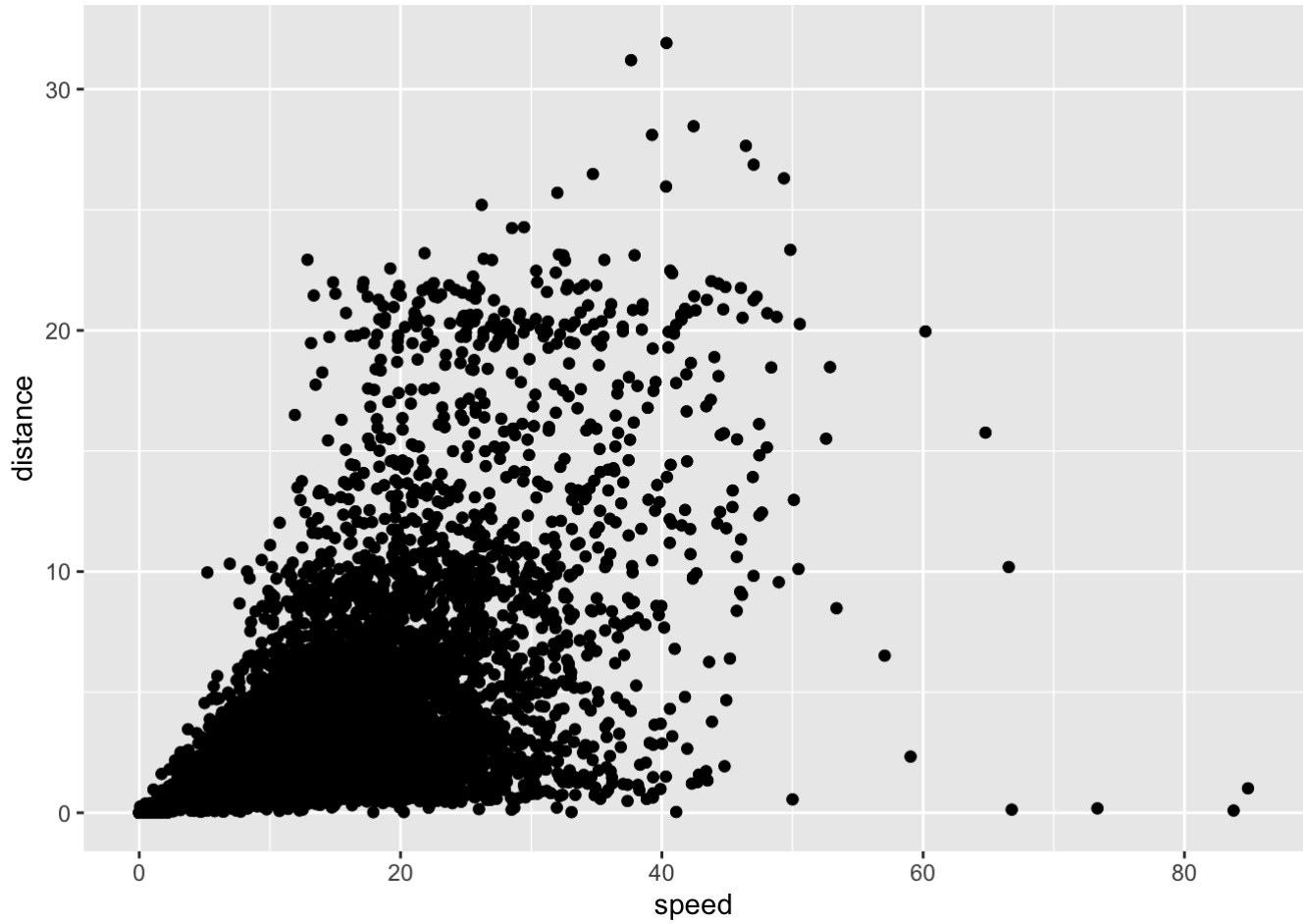



```
test1 %>%
  ggplot(aes(month, speed,color = factor(hour)))+
  geom_boxplot()
```



#大体距离与速度无关，意味着不管开多远，速度变化不大

```
ggplot(test1,aes(speed, distance))+
    geom_point()+
    geom_jitter()
```



距离的远近和行驶速度之间看不到线性关系

```
ggplot(test1,aes(speed, distance))+
  geom_point()+
  geom_smooth()+
  scale_x_log10()+
  scale_y_log10()
```

```
## Warning: Transformation introduced infinite values in continuous x-axis
```

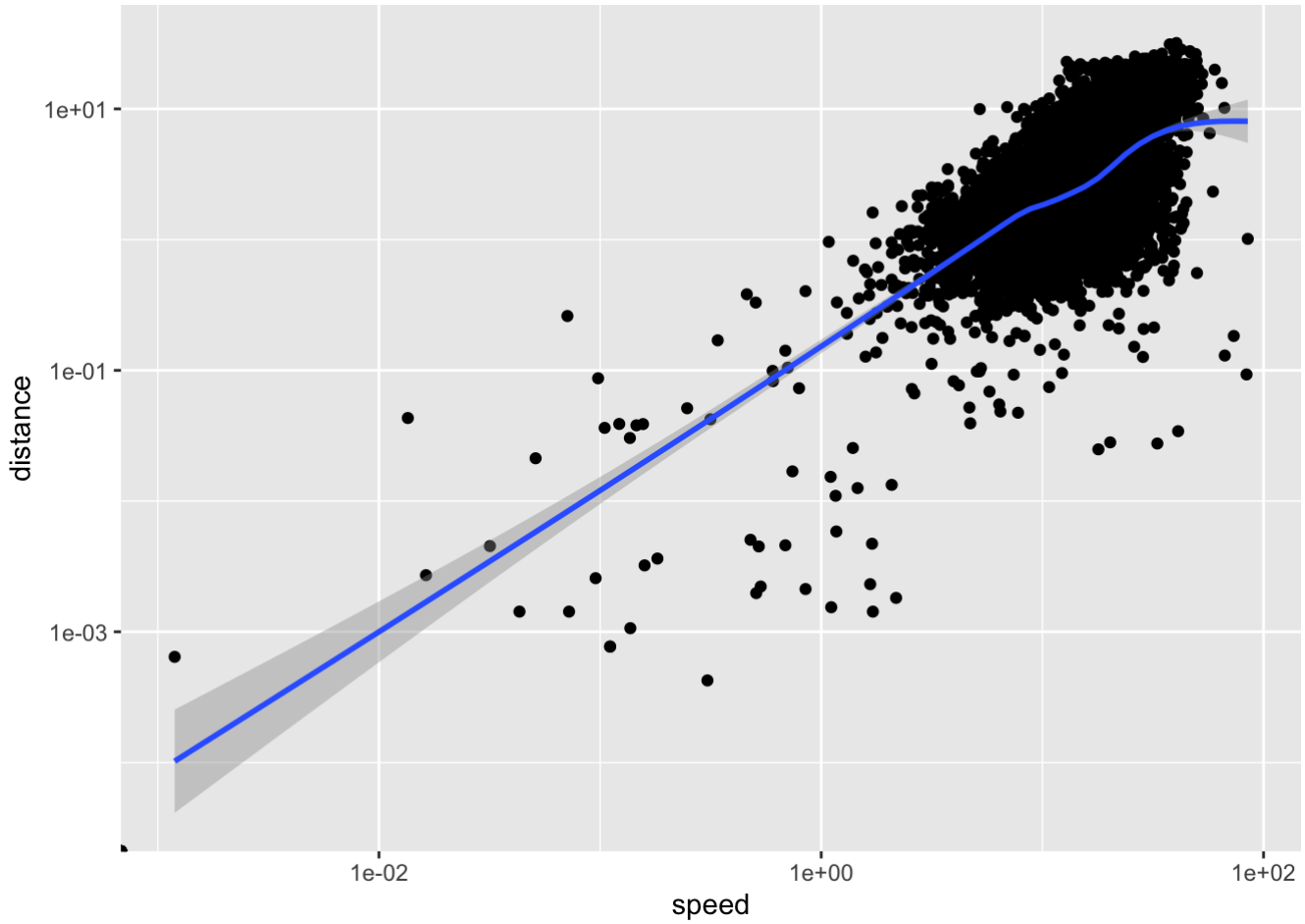
```
## Warning: Transformation introduced infinite values in continuous y-axis
```

```
## Warning: Transformation introduced infinite values in continuous x-axis
```

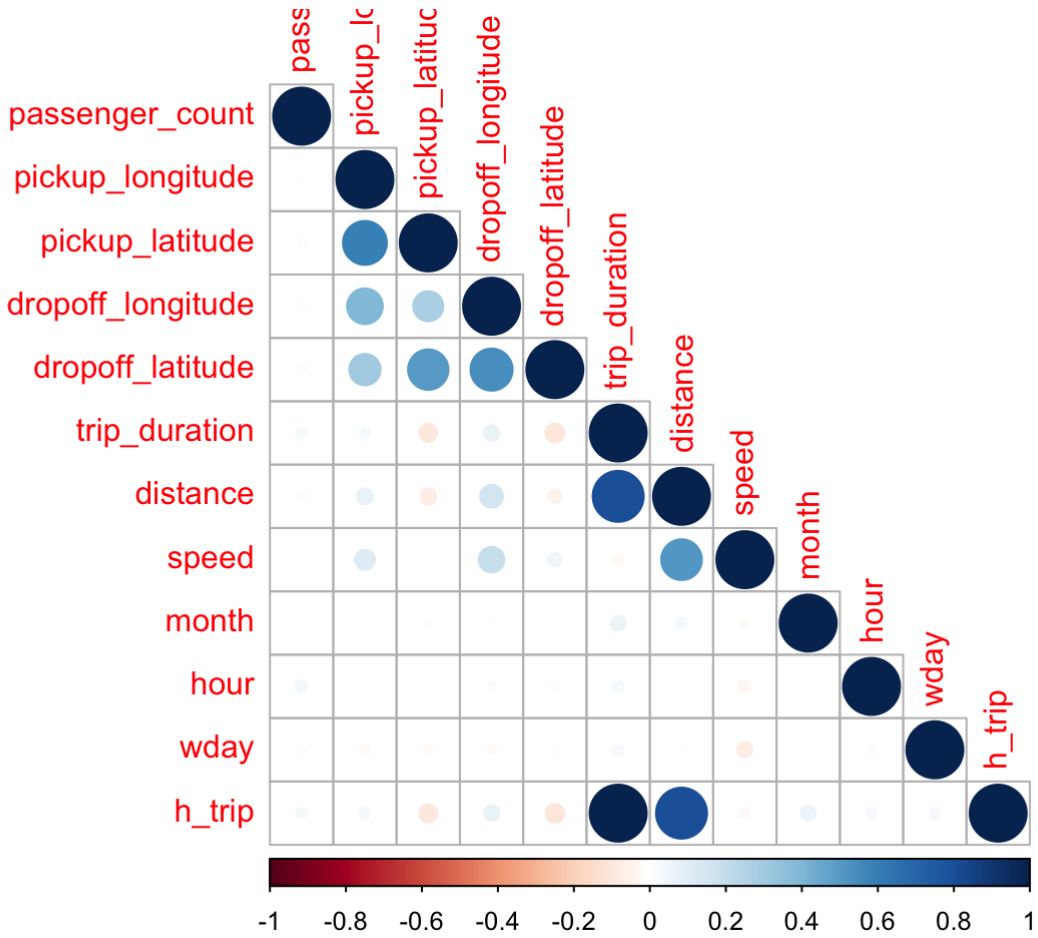
```
## Warning: Transformation introduced infinite values in continuous y-axis
```

```
## `geom_smooth()` using method = 'gam' and formula 'y ~ s(x, bs = "cs")'
```

```
## Warning: Removed 34 rows containing non-finite values (stat_smooth).
```




```
test_cor <- test1 %>%
  select(where(is.numeric)) %>%
  cor(method = "spearman") %>%
  corrplot(type = "lower")
```



行驶

小时数和行驶距离距离呈现强相关关系，上下车时间呈现相关关系，上下车纬度呈现相关关系。

#结论 1.纽约是一个繁忙的城市，出租车业务不论哪一个月份，哪一个星期，哪一天没有大的差异；

2.纽约是一个不夜城。仅仅在每日凌晨2:00-5:00乘客人数较少，但并不是没有，其他每日时间段打车人数都很多，区别不大；

3纽约是一个拥堵的城市，平均车速15km/h，一年中车速集中在0-50km/h，可以说不论什么地方什么时间，要想让车速超过50km/h是一件不容易的事；

4.短程车的乘客占比极高，行程以5km内居多，这与纽约拥堵的路况也是分不开的，从场景上看打车的范围距离都很近，可能是上下班更换交通工具等情况； 5.多人（大于4人）出行具有一定的市场，6个月内呈现小范围波动的增长趋势。

6.异常数据。 一月底二月初，基本无人打车，谷歌显示因为暴风雪天气。