共享单车租用频次分析

周世祥

2020/5/23

案例简介

从 2016 年开始，国内共享单车突然火爆。摩拜，ofo，至少 25 个新的共享单车品牌入驻很多大城市，Kaggle 中有一个关于共享单车的数据集。

美国华盛顿共享单车的租赁量。数据集变量少，简单易懂。

变量介绍

knitr包美化表格
使用knitr包kable()函数，表格输出结果并不会随着屏幕大小而出现原始表格的情况。

```
library(knitr)
kable(data)
```

变量名	变量含义
datatime	日期时间
season	季节
holiday	是否为假期
workingday	是否为工作日
weather	天气
temp	温度
humidity	湿度
windspeed	风速
count	频次

数据准备

做好数据清洗，时间格式，分析汇总，绘图等常用的程序包。

1、导入分析所需程序包

```
library(Rmisc) # multiplot()
```

```
## Warning: package 'Rmisc' was built under R version 4.0.5
```

```
## Loading required package: lattice
```

```
## Loading required package: plyr
```

```
## Warning: package 'plyr' was built under R version 4.0.5
```

```
library(tidyverse) # ggplot()
```

```
## Warning: package 'tidyverse' was built under R version 4.0.5
```

```
## Registered S3 methods overwritten by 'tibble':
##   method     from
##   format.tbl pillar
##   print.tbl  pillar
```

```
## -- Attaching packages --------------------------------------- tidyverse 1.3.1 --
```

```
## v ggplot2 3.3.3     v purrr   0.3.4
## v tibble  3.0.3     v dplyr   1.0.6
## v tidyr   1.1.3     v stringr 1.4.0
## v readr   1.4.0     v forcats 0.5.1
```

```
## Warning: package 'ggplot2' was built under R version 4.0.5
```

```
## Warning: package 'tidyr' was built under R version 4.0.5
```

```
## Warning: package 'readr' was built under R version 4.0.5
```

```
## Warning: package 'dplyr' was built under R version 4.0.5
```

```
## Warning: package 'forcats' was built under R version 4.0.5
```

```
## -- Conflicts ------------------------------------------ tidyverse_conflicts() --
## x dplyr::arrange()   masks plyr::arrange()
## x purrr::compact()   masks plyr::compact()
## x dplyr::count()     masks plyr::count()
## x dplyr::failwith()  masks plyr::failwith()
## x dplyr::filter()    masks stats::filter()
## x dplyr::id()        masks plyr::id()
## x dplyr::lag()       masks stats::lag()
## x dplyr::mutate()    masks plyr::mutate()
```

```
## x dplyr::rename()    masks plyr::rename()
## x dplyr::summarise() masks plyr::summarise()
## x dplyr::summarize() masks plyr::summarize()
```

```
library(data.table)
```

```
## Warning: package 'data.table' was built under R version 4.0.3
```

```
##
## Attaching package: 'data.table'
```

```
## The following objects are masked from 'package:dplyr':
##
##     between, first, last
```

```
## The following object is masked from 'package:purrr':
##
##     transpose
```

```
library(corrplot)#corrplot.mixed()
```

```
## Warning: package 'corrplot' was built under R version 4.0.5
```

```
## corrplot 0.88 loaded
```

```
options(scipen=20)# 避免绘图时使用科学计数法表示某个数值
```

查看数据集的基本结构

根据 str 对导入的数据集结构进行简单探索。除了租用时间点是因子型，其他都是数值型。对 datetime 变量进行差分重塑，对 weather/season 两个变量进行数据的重编码。

```
bike <- fread("./train.csv") #data.table 包中的函数
```

```
str(bike)
```

```
## Classes 'data.table' and 'data.frame':   10886 obs. of  12 variables:
##  $ datetime  : chr  "2011/1/1 0:00" "2011/1/1 1:00" "2011/1/1 2:00" "2011/1/1 3:00" ...
##  $ season    : int  1 1 1 1 1 1 1 1 1 ...
##  $ holiday   : int  0 0 0 0 0 0 0 0 0 ...
##  $ workingday: int  0 0 0 0 0 0 0 0 0 ...
##  $ weather   : int  1 1 1 1 1 2 1 1 1 ...
##  $ temp      : num  9.84 9.02 9.02 9.84 9.84 ...
##  $ atemp     : num  14.4 13.6 13.6 14.4 14.4 ...
##  $ humidity  : int  81 80 80 75 75 75 80 86 75 76 ...
```

```
##  $ windspeed : num  0 0 0 0 0 ...
##  $ casual    : int  3 8 5 3 0 0 2 1 1 8 ...
##  $ registered: int  13 32 27 10 1 1 0 2 7 6 ...
##  $ count     : int  16 40 32 13 1 1 2 3 8 14 ...
##  - attr(*, ".internal.selfref")=<externalptr>
```

查看到各个变量的最小数，最大数，中位数，均值，分位数等。发现最小的租赁次数是 1 次。

```
summary(bike)
```

```
##    datetime             season         holiday          workingday
##  Length:10886       Min.   :1.000   Min.   :0.00000   Min.   :0.0000
##  Class :character   1st Qu.:2.000   1st Qu.:0.00000   1st Qu.:0.0000
##  Mode  :character   Median :3.000   Median :0.00000   Median :1.0000
##                     Mean   :2.507   Mean   :0.02857   Mean   :0.6809
##                     3rd Qu.:4.000   3rd Qu.:0.00000   3rd Qu.:1.0000
##                     Max.   :4.000   Max.   :1.00000   Max.   :1.0000
##     weather           temp           atemp          humidity
##  Min.   :1.000   Min.   : 0.82   Min.   : 0.76   Min.   :  0.00
##  1st Qu.:1.000   1st Qu.:13.94   1st Qu.:16.66   1st Qu.: 47.00
##  Median :1.000   Median :20.50   Median :24.24   Median : 62.00
##  Mean   :1.418   Mean   :20.23   Mean   :23.66   Mean   : 61.89
##  3rd Qu.:2.000   3rd Qu.:26.24   3rd Qu.:31.06   3rd Qu.: 77.00
##  Max.   :4.000   Max.   :41.00   Max.   :45.45   Max.   :100.00
##    windspeed          casual         registered        count
##  Min.   : 0.000   Min.   :  0.00   Min.   :  0.0   Min.   :  1.0
##  1st Qu.: 7.002   1st Qu.:  4.00   1st Qu.: 36.0   1st Qu.: 42.0
##  Median :12.998   Median : 17.00   Median :118.0   Median :145.0
##  Mean   :12.799   Mean   : 36.02   Mean   :155.6   Mean   :191.6
##  3rd Qu.:16.998   3rd Qu.: 49.00   3rd Qu.:222.0   3rd Qu.:284.0
##  Max.   :56.997   Max.   :367.00   Max.   :886.0   Max.   :977.0
```

数据重塑

查看 season 的取值

```
table(bike$season)
```

```
##
##    1    2    3    4
## 2686 2733 2733 2734
```

```
table(bike$weather)
```

```
##
##    1    2    3    4
## 7192 2834  859    1
```

修正取值

```
bike$season <- factor(bike$season, labels = c("Spring", "Summer", "Fall", "Winter"))
bike$weather <- factor(bike$weather, labels = c("Good", "Normal", "Bad", "Very Bad"))

table(bike$season)
```

```
##
## Spring Summer   Fall Winter
##   2686   2733   2733   2734
```

```
table(bike$weather)
```

```
##
##     Good   Normal      Bad Very Bad
##     7192     2834      859        1
```

将变量日期时间转换为时间日期对象，之后再使用 hour 函数将日期时间中的小时数提取出来。

```
library(lubridate)
```

```
## Warning: package 'lubridate' was built under R version 4.0.5
```

```
##
## Attaching package: 'lubridate'
```

```
## The following objects are masked from 'package:data.table':
##
##     hour, isoweek, mday, minute, month, quarter, second, wday, week,
##     yday, year
```

```
## The following objects are masked from 'package:base':
##
##     date, intersect, setdiff, union
```

```
bike$hour <- lubridate::hour(ymd_hm(bike$datetime))
```

剔除 casual 和 registered 两列。

```
bike <- bike[,-c(10, 11)]
head(bike)
```

```
##         datetime season holiday workingday weather temp  atemp humidity
## 1: 2011/1/1 0:00 Spring       0          0    Good 9.84 14.395       81
## 2: 2011/1/1 1:00 Spring       0          0    Good 9.02 13.635       80
## 3: 2011/1/1 2:00 Spring       0          0    Good 9.02 13.635       80
## 4: 2011/1/1 3:00 Spring       0          0    Good 9.84 14.395       75
## 5: 2011/1/1 4:00 Spring       0          0    Good 9.84 14.395       75
## 6: 2011/1/1 5:00 Spring       0          0  Normal 9.84 12.880       75
##    windspeed count hour
## 1:    0.0000    16    0
## 2:    0.0000    40    1
## 3:    0.0000    32    2
## 4:    0.0000    13    3
## 5:    0.0000     1    4
## 6:    6.0032     1    5
```

柱状图在数据分析中的简单应用

分析 24 小时，哪些时段处于单车租赁的高峰，低谷，运用 dplyr 包汇总分析函数结合 ggplot2 包的绘图函数，画条形图。

```
bike %>%
  group_by(hour) %>%
  summarise(mcount = mean(count)) %>%
  ggplot(aes(x = hour, y = mcount, fill = hour)) +
  geom_bar(stat = 'identity') +
  guides(fill = 'none') +
  theme_minimal()
```

看到上午 8 到 9 点，下午 17 到 19 点是高峰期，上下班，说明上班族共享的租赁数多。

再探索假期和工作日的评价租车频次。

```
p8 <- bike %>%
  group_by(holiday) %>%
  summarise(mcount = mean(count)) %>%
  ggplot(aes(x = factor(holiday), y = mcount, fill = factor(holiday))) +
  geom_bar(stat = 'identity') +
  guides(fill = 'none') +
```

```
  labs(x = 'holiday') +
  theme_minimal()


# 探索是否工作日的平均租车频次
p9 <- bike %>%
  group_by(workingday) %>%
  summarise(mcount = mean(count)) %>%
  ggplot(aes(x = factor(workingday), y = mcount, fill = factor(workingday))) +
  geom_bar(stat = 'identity') +
  guides(fill = 'none') +
  labs(x = 'workingday') +
  theme_minimal()

multiplot(p8, p9, cols = 2)
```

差距不大。

柱状和扇形图在数据分析中的应用

探索共享单车数据集可通过 season 变量内不同季节每小时租车次数的对比，来寻求不同季节租赁共享单车的每小时租车次数差异。

```
p2 <- bike %>%
  group_by(season) %>%
  summarise(mcount = mean(count)) %>%
  ggplot(aes(x = reorder(season, mcount), y = mcount, fill = season)) +
  geom_bar(stat = 'identity') +
  labs(x = 'senson', y = 'mcount') +
  guides(fill = 'none') +
  theme_minimal()

p3 <- bike %>%
  group_by(season) %>%
  summarise(mcount = mean(count)) %>%
  ggplot(aes(x = reorder(season, mcount), y = mcount, fill = season)) +
  geom_bar(stat = 'identity', width = 1) +
  coord_polar(theta = "y") +
  labs(x = 'senson', y = 'mcount') +
  guides(fill = 'none') +
  theme_minimal()
```

```
multiplot(p2, p3, cols = 2)
```

两种方式传达的意义是一样的。春天最少，秋天最多。

将可视化变量更改为天气情况，使用条形图和极坐标图。

```
p5 <- bike %>%
  group_by(weather) %>%
  summarise(mcount = mean(count)) %>%
  ggplot(aes(x = reorder(weather, mcount), y = mcount, fill = weather)) +
  geom_bar(stat = 'identity') +
  labs(x = 'weather') +
  guides(fill = 'none') +
  theme_minimal()



p6 <- bike %>%
  group_by(weather) %>%
  summarise(mcount = mean(count)) %>%
  ggplot(aes(x = reorder(weather, mcount), y = mcount, fill = weather)) +
  geom_bar(stat = 'identity', width = 1) +
  coord_polar(theta = "y") +
  labs(x = 'senson', y = 'mcount') +
  guides(fill = 'none') +
  theme_minimal()

multiplot(p5, p6, cols = 2)
```

折线图在数据分析中的应用

观察数据趋势，查看不同时段各个季节的租赁次数的趋势。

```
bike %>%
  group_by(season, hour) %>%
  summarise(mcount = mean(count)) %>%
  ggplot(aes(x = hour, y = mcount, group = season, shape= season, linetype = season)) +
  geom_line() +
  theme_bw() +
  geom_point()
```

```
## `summarise()` has grouped output by 'season'. You can override using the `.groups` argument.
```

将变量改为天气情况，以相同的折线图可视化该变量。

```
bike %>%
  group_by(weather, hour) %>%
  summarise(mcount = mean(count)) %>%
  ggplot(aes(x = hour, y = mcount, group = weather, shape= weather,linetype = weather)) +
  geom_line(aes(group = weather)) +
  theme_bw() +
  geom_point()
```

```
## `summarise()` has grouped output by 'weather'. You can override using the `.groups` argument.
```

探索不同工作日不同时间段的平均租车频次。

```
bike %>%
  group_by(holiday, hour) %>%
  summarise(mcount = mean(count)) %>%
  mutate(Holiday = as.factor(holiday)) %>%
  ggplot(aes(x = hour, y = mcount, group  = Holiday, shape = Holiday )) +
  geom_line(aes(group = factor(holiday))) +
  geom_point() +
  theme_bw()
```

```
## `summarise()` has grouped output by 'holiday'. You can override using the `.groups` argument.
```

可见高峰期在 13 点左右和 18 点左右，8 点左右的租车频次降低了很多。非假期 0 时租车高峰期就是 8 点左右和 18 点左右，即上班族的上下班高峰期。

相关系数图综合分析

变量之间的相关性，帮助用户确认下一步的分析方向。

用 baseR 包中的 cor 函数，可视化用 corrplot.mixed。

```
cor(bike[,c(6:9, 10)]) %>%
  corrplot.mixed()
```

发现 temp 和 atemp 的相关系数到达惊人的 0.98. 最后一行发现，频次与温度呈弱的正相关，与湿度 humidity 呈较弱的负相关。与风速几乎不相关。温度，湿度，风速不会对租车频次产生较大的影响。

参考文献

刘健等，R 数据科学实战工具详解与案例分析，机械工业出版社，2019 年 7 月