EDA2021 综合大作业

191001228 刘秋岐

2022/1/15

1.（30分）MelbCV.csv是墨尔本人行道监控数据的一个子集，用探索一个新的数据集的一般策略（第三讲ppt）对其进行探索性分析，并利用统计描述或图形回答如下问题： 1.1该数据集有多少记录？多少变量？变量名称是什么？它们有意义吗？每个变量是什么类型？每个变量有多少个唯一值？什么值出现的频率最高，多久出现一次？有缺失值吗？如果有，这种情况发生的频率有多高？

library(readr)  
MelbCV<-read\_csv("MelbCV.csv",show\_col\_types = TRUE)#读入数据，15个变量，720条记录

## Rows: 720 Columns: 15

## -- Column specification --------------------------------------------------------  
## Delimiter: ","  
## chr (1): Date  
## dbl (14): Year, Month, Mdate, Weekday\_End, Day, Hour, Town\_Hall-West, Collin...

##   
## i Use `spec()` to retrieve the full column specification for this data.  
## i Specify the column types or set `show\_col\_types = FALSE` to quiet this message.

spec(MelbCV)#查看变量类型

## cols(  
## Date = col\_character(),  
## Year = col\_double(),  
## Month = col\_double(),  
## Mdate = col\_double(),  
## Weekday\_End = col\_double(),  
## Day = col\_double(),  
## Hour = col\_double(),  
## `Town\_Hall-West` = col\_double(),  
## `Collins Place-South` = col\_double(),  
## `Australia on Collins` = col\_double(),  
## `Bourke Street Mall-South` = col\_double(),  
## `Bourke Street Mall-North` = col\_double(),  
## `Melbourne Central` = col\_double(),  
## `Flagstaff Station` = col\_double(),  
## `State Library` = col\_double()  
## )

library(tidyverse)

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

## v ggplot2 3.3.5 v dplyr 1.0.7  
## v tibble 3.1.6 v stringr 1.4.0  
## v tidyr 1.1.4 v forcats 0.5.1  
## v purrr 0.3.4

## -- Conflicts ------------------------------------------ tidyverse\_conflicts() --  
## x dplyr::filter() masks stats::filter()  
## x dplyr::lag() masks stats::lag()

MelbCV<-as.tibble(MelbCV)

## Warning: `as.tibble()` was deprecated in tibble 2.0.0.  
## Please use `as\_tibble()` instead.  
## The signature and semantics have changed, see `?as\_tibble`.  
## This warning is displayed once every 8 hours.  
## Call `lifecycle::last\_lifecycle\_warnings()` to see where this warning was generated.

MelbCV.unique<-apply(MelbCV,2,FUN = function(x){length(unique(x))})  
MelbCV.unique#每个变量有多少唯一值

## Date Year Month   
## 30 1 1   
## Mdate Weekday\_End Day   
## 30 2 7   
## Hour Town\_Hall-West Collins Place-South   
## 24 613 485   
## Australia on Collins Bourke Street Mall-South Bourke Street Mall-North   
## 498 552 548   
## Melbourne Central Flagstaff Station State Library   
## 598 392 308

getmode <- function(v) {  
 uniqv <- data.frame(table(v))  
 i<-which.max(uniqv$Freq)  
 as.numeric(as.vector(uniqv$v[i]))  
}  
MelbCV.mode<-apply(MelbCV[,8:15],2,FUN = function(x){getmode(x)})  
MelbCV.mode#出现频次最高的值

## Town\_Hall-West Collins Place-South Australia on Collins   
## 35 10 16   
## Bourke Street Mall-South Bourke Street Mall-North Melbourne Central   
## 12 4 32   
## Flagstaff Station State Library   
## 10 24

#MelbCV.mode是每列出现频次最高的值即众数，其中Date,year,Month,Mdate,Hour中的各个唯一值出现频次相同如下,Day列中出现频次最高的有1和7;Hour中的出现频次最高的值每24个记录其中就会出现一次;Date,year,Month,每一条记录其都会出现其众数;  
MelbCV$Date%>%table

## .  
## 9/1/2012 9/10/2012 9/11/2012 9/12/2012 9/13/2012 9/14/2012 9/15/2012 9/16/2012   
## 24 24 24 24 24 24 24 24   
## 9/17/2012 9/18/2012 9/19/2012 9/2/2012 9/20/2012 9/21/2012 9/22/2012 9/23/2012   
## 24 24 24 24 24 24 24 24   
## 9/24/2012 9/25/2012 9/26/2012 9/27/2012 9/28/2012 9/29/2012 9/3/2012 9/30/2012   
## 24 24 24 24 24 24 24 24   
## 9/4/2012 9/5/2012 9/6/2012 9/7/2012 9/8/2012 9/9/2012   
## 24 24 24 24 24 24

MelbCV$Year%>%table

## .  
## 2012   
## 720

MelbCV$Month%>%table

## .  
## 9   
## 720

MelbCV$Mdate%>%table

## .  
## 1 2 3 4 5 6 7 8 9 10 11 12 13 14 15 16 17 18 19 20 21 22 23 24 25 26   
## 24 24 24 24 24 24 24 24 24 24 24 24 24 24 24 24 24 24 24 24 24 24 24 24 24 24   
## 27 28 29 30   
## 24 24 24 24

MelbCV$Day%>%table

## .  
## 1 2 3 4 5 6 7   
## 120 96 96 96 96 96 120

MelbCV$Hour%>%table

## .  
## 0 1 2 3 4 5 6 7 8 9 10 11 12 13 14 15 16 17 18 19 20 21 22 23   
## 30 30 30 30 30 30 30 30 30 30 30 30 30 30 30 30 30 30 30 30 30 30 30 30

MelbCV$Weekday\_End%>%table

## .  
## 10 20   
## 480 240

#众数出现的位置  
which(MelbCV$`Town\_Hall-West`==MelbCV.mode[1])

## [1] 411 436 461 579 605

which(MelbCV$`Collins Place-South`==MelbCV.mode[2])

## [1] 31 50 100 125 242 316 413 436 459 554 557 701

which(MelbCV$`Bourke Street Mall-North`==MelbCV.mode[5])

## [1] 220 293 389 437 557 604

which(MelbCV$`Australia on Collins`==MelbCV.mode[3])

## [1] 102 122 126 148 174 268 316 390 414 558

which(MelbCV$`Bourke Street Mall-South`==MelbCV.mode[4])

## [1] 54 74 149 150 242 293 317 412 485

which(MelbCV$`Melbourne Central`==MelbCV.mode[6])

## [1] 52 124 436 653

which(MelbCV$`Flagstaff Station`==MelbCV.mode[7])

## [1] 5 28 49 52 146 175 198 265 553 602 604 626 650 652 699

which(MelbCV$`State Library`==MelbCV.mode[8])

## [1] 54 461 462 606 628

MelbCV%>%summary#发现存在缺失值在`State Library`列，共有357个缺失值

## Date Year Month Mdate Weekday\_End   
## Length:720 Min. :2012 Min. :9 Min. : 1.0 Min. :10.00   
## Class :character 1st Qu.:2012 1st Qu.:9 1st Qu.: 8.0 1st Qu.:10.00   
## Mode :character Median :2012 Median :9 Median :15.5 Median :10.00   
## Mean :2012 Mean :9 Mean :15.5 Mean :13.33   
## 3rd Qu.:2012 3rd Qu.:9 3rd Qu.:23.0 3rd Qu.:20.00   
## Max. :2012 Max. :9 Max. :30.0 Max. :20.00   
##   
## Day Hour Town\_Hall-West Collins Place-South  
## Min. :1 Min. : 0.00 Min. : 13.0 Min. : 1.00   
## 1st Qu.:2 1st Qu.: 5.75 1st Qu.: 239.8 1st Qu.: 57.75   
## Median :4 Median :11.50 Median :1166.0 Median : 275.00   
## Mean :4 Mean :11.50 Mean :1380.1 Mean : 455.98   
## 3rd Qu.:6 3rd Qu.:17.25 3rd Qu.:2448.2 3rd Qu.: 686.75   
## Max. :7 Max. :23.00 Max. :4295.0 Max. :2445.00   
##   
## Australia on Collins Bourke Street Mall-South Bourke Street Mall-North  
## Min. : 0 Min. : 0.00 Min. : 1.00   
## 1st Qu.: 70 1st Qu.: 83.75 1st Qu.: 76.75   
## Median : 333 Median : 447.00 Median : 515.00   
## Mean : 683 Mean :1040.90 Mean :1045.48   
## 3rd Qu.:1210 3rd Qu.:1961.00 3rd Qu.:1980.75   
## Max. :2944 Max. :4372.00 Max. :4285.00   
##   
## Melbourne Central Flagstaff Station State Library   
## Min. : 14.0 Min. : 0.0 Min. : 5.0   
## 1st Qu.: 235.8 1st Qu.: 24.0 1st Qu.: 93.5   
## Median :1089.5 Median : 78.0 Median : 676.0   
## Mean :1264.5 Mean : 689.2 Mean : 800.4   
## 3rd Qu.:2259.8 3rd Qu.: 965.5 3rd Qu.:1455.5   
## Max. :3255.0 Max. :5561.0 Max. :2395.0   
## NA's :357

sum(is.na(MelbCV$`State Library`))#缺失值个数

## [1] 357

naprop<-sum(is.na(MelbCV$`State Library`))/nrow(MelbCV)  
naprop#出现缺失值的频率

## [1] 0.4958333

答：由上可知，该数据集有720条记录，15个变量。变量的名称、意义、类型、唯一值个数、出现的频率最高的值（众数）如下表：

| colname | 含义 | 类型 | 唯一值个数 | 众数 |
| --- | --- | --- | --- | --- |
| Date | 日期 | character | 30 | 所有值出现频次相同 |
| Year | 年份 | double | 1 | 所有值出现频次相同,2012 |
| Month | 月份 | double | 1 | 所有值出现频次相同,9 |
| Mdate | 日 | double | 30 | 所有值出现频次相同 |
| Weekday\_End | 表示周末或工作日，10代表工作日，20代表周末 | double | 2 | 所有值出现频次相同,10,20 |
| Day | 表示一周里的第几天，周日是第一天 | double | 7 | 1,7 |
| Hour | 表示一天中的第几个小时 | double | 24 | 所有值出现频次相同 |
| Town\_Hall-West | 在Town\_Hall-West的人行道通过人数 | double | 613 | 35 |
| Collins Place-South | 在Collins Place-South的人行道通过人数 | double | 485 | 10 |
| Australia on Collins | 在Australia on Collins的人行道通过人数 | double | 498 | 16 |
| Bourke Street Mall-South | 在Bourke Street Mall-South的人行道通过人数 | double | 552 | 12 |
| Bourke Street Mall-North | 在Bourke Street Mall-North的人行道通过人数 | double | 548 | 4 |
| Melbourne Central | 在Melbourne Central的人行道通过人数 | double | 598 | 32 |
| Flagstaff Station | 在Flagstaff Station的人行道通过人数 | double | 392 | 10 |
| State Library | 在Flagstaff Station的人行道通过人数 | double | 308 | 24 |

其中Date,year,Month,Mdate,Hour中的各个唯一值出现频次相同如下,Day列中出现频次最高的有1和7;Hour中的出现频次最高的值每24个记录其中就会出现一次;Date,year,Month,每一条记录其都会出现其众数，后8列众数的出现没发现有什么规律；有缺失值，缺失值都出现在State Library列，有357个，这种情况发生的频率约是0.4958

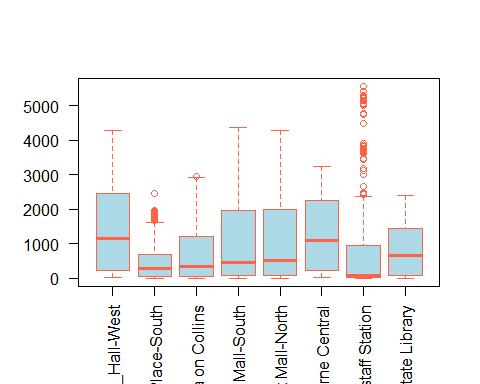
1.2根据1.1的初步探索，你知道了数据集的基本情况，接下来要做感兴趣的探索，通过对变量的观察及描述统计，判断该数据集是否有变量可以忽略，如果有是什么？其中一个变量Weekday\_End，其有多少唯一取值的，根据数据集相关变量及常识说明取值的含义； 答：已经有了年月日的数据，Date就可以忽略了；且这一整个数据集都是12年9月的数据年份、月份都是相同的，Month和Year可以忽略；由1.1可知Weekday\_End有两个唯一值10和20，其中10代表工作日，20代表周末

1.3分别利用描述统计量（5数）和图（选择适当的图）描述各条人行横道监控数据的特征；

MelbCV[,8:15]%>%summary

## Town\_Hall-West Collins Place-South Australia on Collins  
## Min. : 13.0 Min. : 1.00 Min. : 0   
## 1st Qu.: 239.8 1st Qu.: 57.75 1st Qu.: 70   
## Median :1166.0 Median : 275.00 Median : 333   
## Mean :1380.1 Mean : 455.98 Mean : 683   
## 3rd Qu.:2448.2 3rd Qu.: 686.75 3rd Qu.:1210   
## Max. :4295.0 Max. :2445.00 Max. :2944   
##   
## Bourke Street Mall-South Bourke Street Mall-North Melbourne Central  
## Min. : 0.00 Min. : 1.00 Min. : 14.0   
## 1st Qu.: 83.75 1st Qu.: 76.75 1st Qu.: 235.8   
## Median : 447.00 Median : 515.00 Median :1089.5   
## Mean :1040.90 Mean :1045.48 Mean :1264.5   
## 3rd Qu.:1961.00 3rd Qu.:1980.75 3rd Qu.:2259.8   
## Max. :4372.00 Max. :4285.00 Max. :3255.0   
##   
## Flagstaff Station State Library   
## Min. : 0.0 Min. : 5.0   
## 1st Qu.: 24.0 1st Qu.: 93.5   
## Median : 78.0 Median : 676.0   
## Mean : 689.2 Mean : 800.4   
## 3rd Qu.: 965.5 3rd Qu.:1455.5   
## Max. :5561.0 Max. :2395.0   
## NA's :357

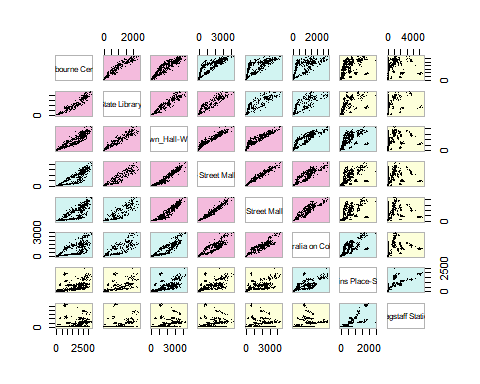
b.output<-boxplot(MelbCV[,8:15],col = "lightblue",border = "tomato",las=2)

 1.4用监控点数据绘制散点矩阵图（scatter matrix plot）,观察各条人行道监控数据间是否存在相关性，哪些变量间存在相关性？

library(gclus)

## 载入需要的程辑包：cluster

library(cluster)  
cpairs(MelbCV[,8:15],panel.colors=dmat.color(cor(MelbCV[,8:15],use = "na.or.complete")),order.single(cor(MelbCV[,8:15],use = "na.or.complete")),pch=".",gap=.8)

 答：如图所示，紫色为相关性最高，绿色其次，黄色最末；“Melbourne Central”和”State Library”、“Town\_Hall-West”存在相关性；“State Library”和”Town\_Hall-West”、“Bourke Street Mall-North”存在相关性；“Town\_Hall-West”和”Bourke Street Mall-North”、“Bourke Street Mall-South”存在相关性；“Bourke Street Mall-North”和”Bourke Street Mall-South”、“Australia on Collins”存在相关性；“Bourke Street Mall-South”和 “Australia on Collins”存在相关性。

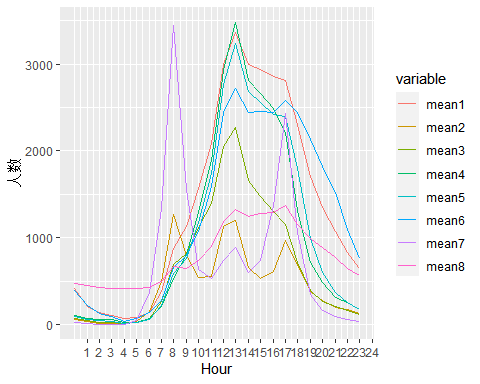
1.5依据统计数据回答各个监控点一天的哪些时段是行人通过的高峰？所有监控点是否有一致的人流高峰时段？若有，是哪个（些）时段？哪个路口人流量最大？哪个路口人流量最小？依据什么得出的结论？人流量的最大值出现在哪个监控点的哪天的哪个时段？

alter<-MelbCV  
alter[is.na(alter)]<-mean(MelbCV$`State Library`,na.rm=TRUE)#缺失值用平均值代替  
#用于做图  
t\_peak<-group\_by(alter,Hour)%>%summarise( mean1=mean(`Town\_Hall-West`),  
 mean2=mean(`Collins Place-South`),  
 mean3=mean(`Australia on Collins`),  
 mean4=mean(`Bourke Street Mall-South`),  
 mean5=mean(`Bourke Street Mall-North`),  
 mean6=mean(`Melbourne Central`),  
 mean7=mean(`Flagstaff Station`),  
 mean8=mean(`State Library`))  
  
library(reshape2)

##   
## 载入程辑包：'reshape2'

## The following object is masked from 'package:tidyr':  
##   
## smiths

long\_peak <- melt(t\_peak, id="Hour")  
ggplot(data=long\_peak,aes(x=Hour, y=value, group=variable, color=variable))+  
 geom\_line()+  
 ylab("人数")+  
 scale\_x\_continuous(breaks=seq(0:23))



#用于计算人流量  
t\_sum<-apply(alter[,8:15],2,sum)  
t\_sum

## Town\_Hall-West Collins Place-South Australia on Collins   
## 993690.0 328307.0 491753.0   
## Bourke Street Mall-South Bourke Street Mall-North Melbourne Central   
## 749446.0 752744.0 910455.0   
## Flagstaff Station State Library   
## 496204.0 576293.6

#人流量最大值  
values<-alter[,8:15]  
alter[which(values==max(values),arr.ind = TRUE)[1],]

## # A tibble: 1 x 15  
## Date Year Month Mdate Weekday\_End Day Hour `Town\_Hall-West`  
## <chr> <dbl> <dbl> <dbl> <dbl> <dbl> <dbl> <dbl>  
## 1 9/20/2012 2012 9 20 10 5 8 1288  
## # ... with 7 more variables: Collins Place-South <dbl>,  
## # Australia on Collins <dbl>, Bourke Street Mall-South <dbl>,  
## # Bourke Street Mall-North <dbl>, Melbourne Central <dbl>,  
## # Flagstaff Station <dbl>, State Library <dbl>

答：如上图一，各个监控点一天的13：00、8：00、13：00、13：00、13：00、13：00、8：00、17：00是行人通过的高峰；所有监控点有一致的人流高峰时段是8：00、13：00、17：00；Town\_Hall-West路口人流量最大，Collins Place-South路口人流量最小；人流量的最大值出现在2012年9月20日8：00的Flagstaff Station监测点。

1.6是否存在缺失值？出现在哪个（些）变量中，可否忽略？如不能，说明原因，并尝试对其进行插补，采用怎样的插补策略比较合适？ 答：存在缺失值，用平均值插补法进行插补,如上题1.5。

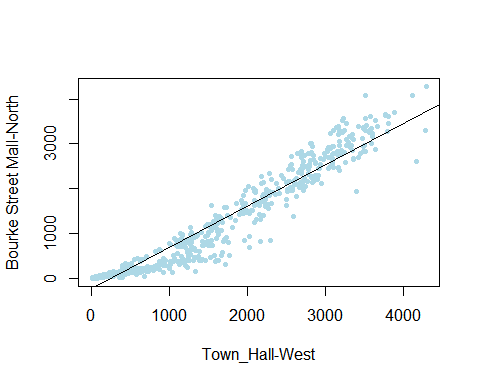
1.7对有明显相关性的变量尝试建立相应的统计模型，并通过作图检验模型。 答：“Town\_Hall-West”和”Bourke Street Mall-North”明显具有线性相关性

write\_csv(alter[,c(4:15)],"refine.csv")  
cor.test(alter$`Town\_Hall-West`,alter$`Bourke Street Mall-North`)

##   
## Pearson's product-moment correlation  
##   
## data: alter$`Town\_Hall-West` and alter$`Bourke Street Mall-North`  
## t = 102.04, df = 718, p-value < 2.2e-16  
## alternative hypothesis: true correlation is not equal to 0  
## 95 percent confidence interval:  
## 0.9621339 0.9716086  
## sample estimates:  
## cor   
## 0.967206

从运行结果可以得到相关系数r=0.967206，P值< 2.2e-16，表明”Town\_Hall-West”人数和”Bourke Street Mall-North”人数两者高度相关，存在显著的正相关关系。

lm<-lm(alter$`Bourke Street Mall-North`~alter$`Town\_Hall-West`)  
plot(alter$`Bourke Street Mall-North`~alter$`Town\_Hall-West`,xlab="Town\_Hall-West",ylab="Bourke Street Mall-North",col="lightblue",pch=20)  
abline(lm)



summary(lm)

##   
## Call:  
## lm(formula = alter$`Bourke Street Mall-North` ~ alter$`Town\_Hall-West`)  
##   
## Residuals:  
## Min 1Q Median 3Q Max   
## -1048.2 -146.5 47.1 176.3 1074.2   
##   
## Coefficients:  
## Estimate Std. Error t value Pr(>|t|)   
## (Intercept) -2.252e+02 1.630e+01 -13.81 <2e-16 \*\*\*  
## alter$`Town\_Hall-West` 9.207e-01 9.023e-03 102.04 <2e-16 \*\*\*  
## ---  
## Signif. codes: 0 '\*\*\*' 0.001 '\*\*' 0.01 '\*' 0.05 '.' 0.1 ' ' 1  
##   
## Residual standard error: 282.2 on 718 degrees of freedom  
## Multiple R-squared: 0.9355, Adjusted R-squared: 0.9354   
## F-statistic: 1.041e+04 on 1 and 718 DF, p-value: < 2.2e-16

从运行结果可以看出：它们的p 值均很小。故是非常显著的。相关系数0.9355拟合优度较高，表明在BourkeStreetMall-North与Town\_Hall-West的回归关系的数据中，由93.55%的数据可以由Bourke Street Mall-North与Town\_Hall-West的线性关系来解释，可见两者之间有较强的相关关系。 线性回归方程与回归系数的检验都是显著的，因此得到回归方程为

2.（40分）Bjmo.csv中存放了中国地面国际交换站气候资料日值数据集北京站点从2010年1月1日到2014年3月31日的日值数据，数据说明文档及数据格式说明文档见“中国地面国际交换站气候资料日值数据集\_数据格式说明文档.txt”，根据探索一个新的数据集的一般策略对其进行探索，并回答下列问题： 2.1选取除区站号和带有“气（汽）压”以外的变量存放到对象bjmoeda中，通过str（）函数、描述统计和适当的图查看bjmoeda数据基本特征，bjmoeda有多少变量？每个变量有多少个唯一值？什么值出现的频率最高，多久出现一次？有缺失值吗？如果有，这种情况发生的频率有多高？

bjmo<-read\_csv("bjmo.csv",show\_col\_types = FALSE)  
bjmoeda<-bjmo[,c(2:8,10,11,13,14,16,18:22)]  
str(bjmoeda)

## tibble [1,551 x 17] (S3: tbl\_df/tbl/data.frame)  
## $ 年 : num [1:1551] 2010 2010 2010 2010 2010 2010 2010 2010 2010 2010 ...  
## $ 月 : num [1:1551] 1 1 1 1 1 1 1 1 1 1 ...  
## $ 日 : num [1:1551] 1 2 3 4 5 6 7 8 9 10 ...  
## $ 20-20时降水量 : num [1:1551] 0 4 99 1 0 0 0 0 0 0 ...  
## $ 大型蒸发量 : num [1:1551] 32766 32766 32766 32766 32766 ...  
## $ 极大风速 : num [1:1551] 62 57 110 179 134 61 51 42 72 51 ...  
## $ 极大风速的风向: num [1:1551] 3 4 5 16 2 2 3 10 3 4 ...  
## $ 平均风速 : num [1:1551] 16 21 27 50 23 15 13 9 16 19 ...  
## $ 平均气温 : num [1:1551] -50 -45 -81 -103 -125 -112 -97 -98 -74 -70 ...  
## $ 平均相对湿度 : num [1:1551] 36 79 86 39 41 41 41 60 57 58 ...  
## $ 日照时数 : num [1:1551] 34 0 0 82 80 78 72 0 72 42 ...  
## $ 日最低气温 : num [1:1551] -102 -63 -99 -134 -156 -167 -135 -121 -134 -110 ...  
## $ 日最高气温 : num [1:1551] -16 -21 -44 -85 -75 -59 -44 -74 -28 -28 ...  
## $ 小型蒸发量 : num [1:1551] 12 2 0 13 10 7 8 4 7 6 ...  
## $ 最大风速 : num [1:1551] 34 34 57 77 60 33 27 22 43 33 ...  
## $ 最大风速的风向: num [1:1551] 3 4 5 16 1 3 2 9 1 3 ...  
## $ 最小相对湿度 : num [1:1551] 23 27 72 29 28 25 25 42 33 39 ...

bjmoeda.unique<-apply(bjmoeda,2,FUN = function(x){length(unique(x))})  
bjmoeda.unique#每个变量有多少唯一值

## 年 月 日 20-20时降水量 大型蒸发量   
## 5 12 31 136 87   
## 极大风速 极大风速的风向 平均风速 平均气温 平均相对湿度   
## 156 16 54 393 85   
## 日照时数 日最低气温 日最高气温 小型蒸发量 最大风速   
## 137 394 410 87 90   
## 最大风速的风向 最小相对湿度   
## 16 82

bjmoeda.mode<-apply(bjmoeda,2,FUN = function(x){getmode(x)})  
bjmoeda.mode#众数,没有发现明显的出现规律

## 年 月 日 20-20时降水量 大型蒸发量   
## 2012 1 1 0 32766   
## 极大风速 极大风速的风向 平均风速 平均气温 平均相对湿度   
## 56 10 16 277 65   
## 日照时数 日最低气温 日最高气温 小型蒸发量 最大风速   
## 0 -39 278 32766 34   
## 最大风速的风向 最小相对湿度   
## 10 11

summary(bjmoeda)#没有缺失值

## 年 月 日 20-20时降水量 大型蒸发量   
## Min. :2010 Min. : 1.00 Min. : 1.00 Min. : 0 Min. : 9   
## 1st Qu.:2011 1st Qu.: 3.00 1st Qu.: 8.00 1st Qu.: 0 1st Qu.: 40   
## Median :2012 Median : 6.00 Median :16.00 Median : 0 Median : 70   
## Mean :2012 Mean : 6.26 Mean :15.72 Mean : 2864 Mean :14938   
## 3rd Qu.:2013 3rd Qu.: 9.00 3rd Qu.:23.00 3rd Qu.: 1 3rd Qu.:32766   
## Max. :2014 Max. :12.00 Max. :31.00 Max. :32766 Max. :32766   
## 极大风速 极大风速的风向 平均风速 平均气温   
## Min. : 25.00 Min. : 1.000 Min. : 6.00 Min. : -125.0   
## 1st Qu.: 56.00 1st Qu.: 4.000 1st Qu.:16.00 1st Qu.: 15.0   
## Median : 73.00 Median : 9.000 Median :20.00 Median : 132.0   
## Mean : 80.63 Mean : 8.369 Mean :21.99 Mean : 145.2   
## 3rd Qu.:100.50 3rd Qu.:11.000 3rd Qu.:26.00 3rd Qu.: 234.0   
## Max. :228.00 Max. :16.000 Max. :66.00 Max. :32766.0   
## 平均相对湿度 日照时数 日最低气温 日最高气温   
## Min. : 9.00 Min. : 0.00 Min. :-167.00 Min. :-85.0   
## 1st Qu.:34.00 1st Qu.: 31.50 1st Qu.: -27.00 1st Qu.: 67.0   
## Median :52.00 Median : 77.00 Median : 76.00 Median :189.0   
## Mean :51.23 Mean : 65.92 Mean : 77.38 Mean :174.8   
## 3rd Qu.:67.00 3rd Qu.: 97.00 3rd Qu.: 186.00 3rd Qu.:280.0   
## Max. :97.00 Max. :141.00 Max. : 292.00 Max. :406.0   
## 小型蒸发量 最大风速 最大风速的风向 最小相对湿度   
## Min. : 0 Min. : 17.00 Min. : 1.000 Min. : 4.00   
## 1st Qu.: 24 1st Qu.: 35.50 1st Qu.: 4.000 1st Qu.:16.00   
## Median :32766 Median : 46.00 Median : 9.000 Median :26.00   
## Mean :18096 Mean : 49.06 Mean : 8.397 Mean :30.85   
## 3rd Qu.:32766 3rd Qu.: 61.00 3rd Qu.:11.000 3rd Qu.:43.00   
## Max. :32766 Max. :120.00 Max. :16.000 Max. :85.00

2.2阅读“气象数据集说明文档”，检查并处理缺失值，将缺失值替换成NA；检查除缺失值外的其他需要处理的数据编码，给出将其转换为正常值的策略；

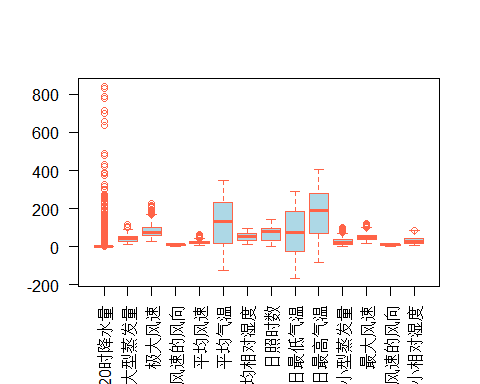
bjmoeda[bjmoeda==32766]<-NA  
bjmoeda[bjmoeda==32700]<-0

2.3用折线图和箱线图描述各个观测变量值的变化情况，计算各个变量的月平均值，最大值，最小值以及中值，找出5年间北京地区气候特征，最高温、最低温、平均温的极值分别出现在哪年哪月哪日，值是多少？

group\_bjmoeda<-group\_by(bjmoeda,年,月)  
summary(group\_bjmoeda)

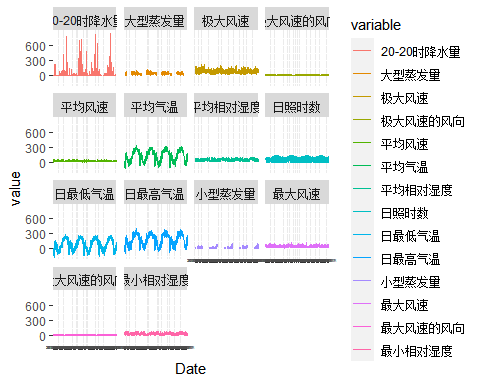
## 年 月 日 20-20时降水量   
## Min. :2010 Min. : 1.00 Min. : 1.00 Min. : 0.00   
## 1st Qu.:2011 1st Qu.: 3.00 1st Qu.: 8.00 1st Qu.: 0.00   
## Median :2012 Median : 6.00 Median :16.00 Median : 0.00   
## Mean :2012 Mean : 6.26 Mean :15.72 Mean : 16.85   
## 3rd Qu.:2013 3rd Qu.: 9.00 3rd Qu.:23.00 3rd Qu.: 0.00   
## Max. :2014 Max. :12.00 Max. :31.00 Max. :842.00   
## NA's :31   
## 大型蒸发量 极大风速 极大风速的风向 平均风速   
## Min. : 9.00 Min. : 25.00 Min. : 1.000 Min. : 6.00   
## 1st Qu.: 29.00 1st Qu.: 56.00 1st Qu.: 4.000 1st Qu.:16.00   
## Median : 41.00 Median : 73.00 Median : 9.000 Median :20.00   
## Mean : 42.82 Mean : 80.63 Mean : 8.369 Mean :21.99   
## 3rd Qu.: 54.00 3rd Qu.:100.50 3rd Qu.:11.000 3rd Qu.:26.00   
## Max. :117.00 Max. :228.00 Max. :16.000 Max. :66.00   
## NA's :706   
## 平均气温 平均相对湿度 日照时数 日最低气温   
## Min. :-125.0 Min. : 9.00 Min. : 0.00 Min. :-167.00   
## 1st Qu.: 15.0 1st Qu.:34.00 1st Qu.: 31.50 1st Qu.: -27.00   
## Median : 131.5 Median :52.00 Median : 77.00 Median : 76.00   
## Mean : 124.1 Mean :51.23 Mean : 65.92 Mean : 77.38   
## 3rd Qu.: 234.0 3rd Qu.:67.00 3rd Qu.: 97.00 3rd Qu.: 186.00   
## Max. : 345.0 Max. :97.00 Max. :141.00 Max. : 292.00   
## NA's :1   
## 日最高气温 小型蒸发量 最大风速 最大风速的风向   
## Min. :-85.0 Min. : 0.00 Min. : 17.00 Min. : 1.000   
## 1st Qu.: 67.0 1st Qu.: 13.00 1st Qu.: 35.50 1st Qu.: 4.000   
## Median :189.0 Median : 21.00 Median : 46.00 Median : 9.000   
## Mean :174.8 Mean : 26.58 Mean : 49.06 Mean : 8.397   
## 3rd Qu.:280.0 3rd Qu.: 35.00 3rd Qu.: 61.00 3rd Qu.:11.000   
## Max. :406.0 Max. :102.00 Max. :120.00 Max. :16.000   
## NA's :856   
## 最小相对湿度   
## Min. : 4.00   
## 1st Qu.:16.00   
## Median :26.00   
## Mean :30.85   
## 3rd Qu.:43.00   
## Max. :85.00   
##

boxplot(bjmoeda[,4:17],col = "lightblue",border = "tomato",las=2)



a<-paste(as.character(bjmoeda$年),as.character(bjmoeda$月),sep = "/")  
Date<-paste(a,as.character(bjmoeda$日),sep = "/")  
new\_bjmoeda<-mutate(bjmoeda,Date=Date)  
long\_bjmoeda <- melt(new\_bjmoeda[,4:18], id="Date")  
ggplot(data=long\_bjmoeda,aes(x=Date, y=value, group=variable, color=variable))+  
 geom\_line()+  
 scale\_x\_discrete(breaks = Date[seq(1,length(Date), by=29)])+  
 theme(axis.text.x=element\_text(vjust=10,size=3))+  
 facet\_wrap(variable~.)

## Warning: Removed 121 row(s) containing missing values (geom\_path).



bjmoeda[which(bjmoeda$平均气温==max(bjmoeda$平均气温,na.rm=TRUE)),]

## # A tibble: 1 x 17  
## 年 月 日 `20-20时降水量` 大型蒸发量 极大风速 极大风速的风向 平均风速  
## <dbl> <dbl> <dbl> <dbl> <dbl> <dbl> <dbl> <dbl>  
## 1 2010 7 6 0 100 100 8 26  
## # ... with 9 more variables: 平均气温 <dbl>, 平均相对湿度 <dbl>,  
## # 日照时数 <dbl>, 日最低气温 <dbl>, 日最高气温 <dbl>, 小型蒸发量 <dbl>,  
## # 最大风速 <dbl>, 最大风速的风向 <dbl>, 最小相对湿度 <dbl>

bjmoeda[which(bjmoeda$日最低气温==max(bjmoeda$日最低气温,na.rm=TRUE)),]

## # A tibble: 1 x 17  
## 年 月 日 `20-20时降水量` 大型蒸发量 极大风速 极大风速的风向 平均风速  
## <dbl> <dbl> <dbl> <dbl> <dbl> <dbl> <dbl> <dbl>  
## 1 2010 7 31 0 56 130 2 23  
## # ... with 9 more variables: 平均气温 <dbl>, 平均相对湿度 <dbl>,  
## # 日照时数 <dbl>, 日最低气温 <dbl>, 日最高气温 <dbl>, 小型蒸发量 <dbl>,  
## # 最大风速 <dbl>, 最大风速的风向 <dbl>, 最小相对湿度 <dbl>

bjmoeda[which(bjmoeda$日最高气温==max(bjmoeda$日最高气温,na.rm=TRUE)),]

## # A tibble: 1 x 17  
## 年 月 日 `20-20时降水量` 大型蒸发量 极大风速 极大风速的风向 平均风速  
## <dbl> <dbl> <dbl> <dbl> <dbl> <dbl> <dbl> <dbl>  
## 1 2010 7 5 0 86 97 16 25  
## # ... with 9 more variables: 平均气温 <dbl>, 平均相对湿度 <dbl>,  
## # 日照时数 <dbl>, 日最低气温 <dbl>, 日最高气温 <dbl>, 小型蒸发量 <dbl>,  
## # 最大风速 <dbl>, 最大风速的风向 <dbl>, 最小相对湿度 <dbl>

bjmoeda[which(bjmoeda$平均气温==min(bjmoeda$平均气温,na.rm=TRUE)),]

## # A tibble: 1 x 17  
## 年 月 日 `20-20时降水量` 大型蒸发量 极大风速 极大风速的风向 平均风速  
## <dbl> <dbl> <dbl> <dbl> <dbl> <dbl> <dbl> <dbl>  
## 1 2010 1 5 0 NA 134 2 23  
## # ... with 9 more variables: 平均气温 <dbl>, 平均相对湿度 <dbl>,  
## # 日照时数 <dbl>, 日最低气温 <dbl>, 日最高气温 <dbl>, 小型蒸发量 <dbl>,  
## # 最大风速 <dbl>, 最大风速的风向 <dbl>, 最小相对湿度 <dbl>

bjmoeda[which(bjmoeda$日最低气温==min(bjmoeda$日最低气温,na.rm=TRUE)),]

## # A tibble: 1 x 17  
## 年 月 日 `20-20时降水量` 大型蒸发量 极大风速 极大风速的风向 平均风速  
## <dbl> <dbl> <dbl> <dbl> <dbl> <dbl> <dbl> <dbl>  
## 1 2010 1 6 0 NA 61 2 15  
## # ... with 9 more variables: 平均气温 <dbl>, 平均相对湿度 <dbl>,  
## # 日照时数 <dbl>, 日最低气温 <dbl>, 日最高气温 <dbl>, 小型蒸发量 <dbl>,  
## # 最大风速 <dbl>, 最大风速的风向 <dbl>, 最小相对湿度 <dbl>

bjmoeda[which(bjmoeda$日最高气温==min(bjmoeda$日最高气温,na.rm=TRUE)),]

## # A tibble: 1 x 17  
## 年 月 日 `20-20时降水量` 大型蒸发量 极大风速 极大风速的风向 平均风速  
## <dbl> <dbl> <dbl> <dbl> <dbl> <dbl> <dbl> <dbl>  
## 1 2010 1 4 1 NA 179 16 50  
## # ... with 9 more variables: 平均气温 <dbl>, 平均相对湿度 <dbl>,  
## # 日照时数 <dbl>, 日最低气温 <dbl>, 日最高气温 <dbl>, 小型蒸发量 <dbl>,  
## # 最大风速 <dbl>, 最大风速的风向 <dbl>, 最小相对湿度 <dbl>

答：各个变量的月平均值，最大值，最小值以及中值如上方第一个输出；5年间北京地区气候特征，日最高温、最低温、平均温的极大值分别出现在2010年7月5日、2010年7月31日、2010年7月6日，值分别为406、292、345；5年间北京地区气候特征，日最高温、最低温、平均温的极小值分别出现在2010年1月4日、2010年1月6日、2010年1月5日，值分别为-85、-167、-125；

2.4这5年间北京的大风（10m/s以上）天气有多少天？最大风速是多少？探索风向和风速的关系，是否存在相关性？若存在，相关性如何？大风极端天气出现在什么季节？

length(bjmoeda[which(bjmoeda$极大风速>100),])#单位是0.1m/s

## [1] 17

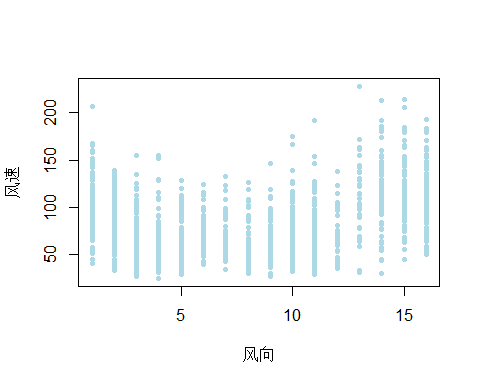
max(bjmoeda$极大风速)

## [1] 228

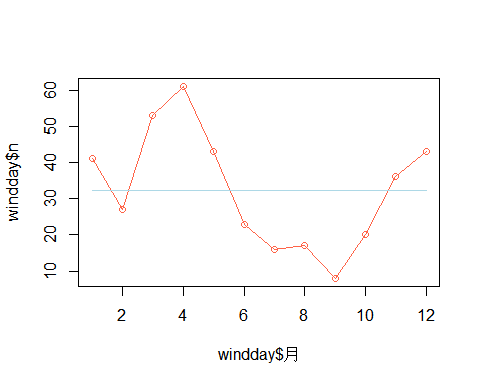
cor(bjmoeda$极大风速,bjmoeda$极大风速的风向)

## [1] 0.2745365

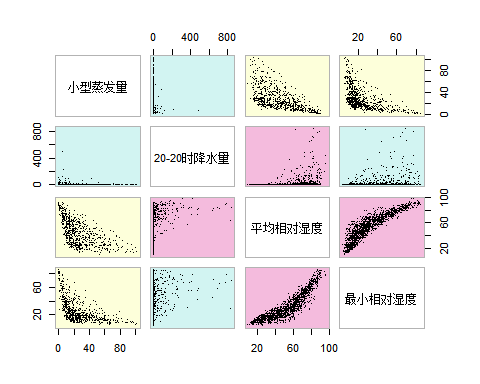
plot(bjmoeda$极大风速~bjmoeda$极大风速的风向,xlab="风向",ylab="风速",col="lightblue",pch=20)



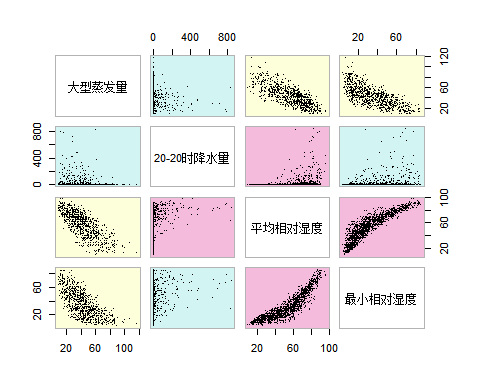
windday<-bjmoeda[which(bjmoeda$极大风速>100),]%>%group\_by(月)%>%summarise(n=n())  
plot(windday$月,windday$n,type="o",col="tomato")  
lines(windday$月,rep(mean(windday$n),length(windday$月)),col="lightblue")

 答：5年间北京的大风（10m/s以上）天气有17天，最大风速是22.8m/s;探索风向和风速存在相关性,相关性系数为0.2745365>0，为正相关；如图，大风极端天气出现在1、35、1112月的次数较多，35月为春季122月为冬季。 2.5探索降水量和蒸发量以及湿度之间的关系；用适当的图描述。

cpairs(bjmoeda[,c(4,10,14,17)],panel.colors=dmat.color(cor(bjmoeda[,c(4,10,14,17)],use = "na.or.complete")),order.single(cor(bjmoeda[,c(4,10,14,17)],use = "na.or.complete")),pch=".",gap=.8)



cpairs(bjmoeda[,c(4,5,10,17)],panel.colors=dmat.color(cor(bjmoeda[,c(4,5,10,17)],use = "na.or.complete")),order.single(cor(bjmoeda[,c(4,5,10,17)],use = "na.or.complete")),pch=".",gap=.8)

 答：如图所示，蒸发量与湿度之间存在负相关关系，蒸发量与降水量之间存在负相关关系，降水量与湿度间存在正相关关系

3.（30分）垃圾邮件数据分析，数据存放在spam.csv中，数据描述见附录。探索并尝试分析该数据。 3.1用探索一个新的数据集的一般策略对其进行探索，用描述统计方法对数据结构进行概述；

spam<-read\_csv("spam.csv",show\_col\_types = FALSE)  
spec(spam)#变量名称和类型

## cols(  
## isuid = col\_double(),  
## id = col\_double(),  
## `day of week` = col\_character(),  
## `time of day` = col\_double(),  
## size.kb = col\_double(),  
## box = col\_character(),  
## domain = col\_character(),  
## local = col\_character(),  
## digits = col\_double(),  
## name = col\_character(),  
## cappct = col\_double(),  
## special = col\_double(),  
## credit = col\_character(),  
## sucker = col\_character(),  
## porn = col\_character(),  
## chain = col\_character(),  
## username = col\_character(),  
## `large text` = col\_character(),  
## spampct = col\_double(),  
## category = col\_character(),  
## spam = col\_character()  
## )

spam.unique<-apply(spam,2,FUN = function(x){length(unique(x))})  
spam.unique#每个变量有多少唯一值

## isuid id day of week time of day size.kb box   
## 19 392 7 24 107 2   
## domain local digits name cappct special   
## 44 2 18 3 461 22   
## credit sucker porn chain username large text   
## 2 2 2 2 2 2   
## spampct category spam   
## 97 4 2

summary(spam)

## isuid id day of week time of day   
## Min. : 1.000 Min. : 1.0 Length:2171 Min. : 0.00   
## 1st Qu.: 4.000 1st Qu.: 33.0 Class :character 1st Qu.: 9.00   
## Median : 9.000 Median : 62.0 Mode :character Median :12.00   
## Mean : 9.234 Mean : 201.2 Mean :12.26   
## 3rd Qu.:14.000 3rd Qu.: 108.0 3rd Qu.:16.00   
## Max. :19.000 Max. :3470.0 Max. :23.00   
##   
## size.kb box domain local   
## Min. : 0.00 Length:2171 Length:2171 Length:2171   
## 1st Qu.: 2.00 Class :character Class :character Class :character   
## Median : 4.00 Mode :character Mode :character Mode :character   
## Mean : 16.49   
## 3rd Qu.: 7.00   
## Max. :1337.00   
##   
## digits name cappct special   
## Min. : 0.000 Length:2171 Min. :0.0000 Min. : 0.000   
## 1st Qu.: 0.000 Class :character 1st Qu.:0.0600 1st Qu.: 0.000   
## Median : 0.000 Mode :character Median :0.1280 Median : 1.000   
## Mean : 0.591 Mean :0.1584 Mean : 1.397   
## 3rd Qu.: 0.000 3rd Qu.:0.2000 3rd Qu.: 2.000   
## Max. :23.000 Max. :1.0000 Max. :35.000   
##   
## credit sucker porn chain   
## Length:2171 Length:2171 Length:2171 Length:2171   
## Class :character Class :character Class :character Class :character   
## Mode :character Mode :character Mode :character Mode :character   
##   
##   
##   
##   
## username large text spampct category   
## Length:2171 Length:2171 Min. : 0.00 Length:2171   
## Class :character Class :character 1st Qu.:11.00 Class :character   
## Mode :character Mode :character Median :47.50 Mode :character   
## Mean :44.63   
## 3rd Qu.:76.00   
## Max. :99.00   
## NA's :1353   
## spam   
## Length:2171   
## Class :character   
## Mode :character   
##   
##   
##   
##

nona\_s<-na.omit(spam)  
spam.mode<-apply(nona\_s[,c(1,2,4,5,9,11,12,19)],2,FUN = function(x){getmode(x)})  
spam.mode#出现频次最高的值

## isuid id time of day size.kb digits cappct   
## 2 42 14 3 0 0   
## special spampct   
## 0 0

spam$`day of week`%>%table

## .  
## Fri Mon Sat Sun Thu Tue Wed   
## 316 322 136 178 398 360 461

spam$box%>%table

## .  
## no yes   
## 1066 1105

spam$domain%>%table

## .  
## at bak be by c ca ch cn co com cy cz de edu fr gov   
## 2 1 1 1 1 9 3 8 1 807 1 1 45 1037 4 10   
## hfc hkd hu il info is it jp kr kw lt lv my net nz org   
## 1 1 1 1 13 2 7 1 1 1 1 1 1 118 2 26   
## ph pin pl pt rdp ro se sk tsp tw uk us   
## 1 1 2 1 1 2 1 1 1 1 42 7

spam$local%>%table

## .  
## no yes   
## 1221 950

spam$name%>%table

## .  
## empty name single   
## 295 1611 265

spam$credit%>%table

## .  
## no yes   
## 2081 90

spam$sucker%>%table

## .  
## no yes   
## 1945 226

spam$porn%>%table

## .  
## no yes   
## 2144 27

spam$chain%>%table

## .  
## no yes   
## 2113 58

spam$username%>%table

## .  
## no yes   
## 2080 91

spam$`large text`%>%table

## .  
## no yes   
## 1758 413

spam$category%>%table

## .  
## com list news ord   
## 635 690 100 746

spam$spam%>%table

## .  
## no yes   
## 1461 710

summary(spam)

## isuid id day of week time of day   
## Min. : 1.000 Min. : 1.0 Length:2171 Min. : 0.00   
## 1st Qu.: 4.000 1st Qu.: 33.0 Class :character 1st Qu.: 9.00   
## Median : 9.000 Median : 62.0 Mode :character Median :12.00   
## Mean : 9.234 Mean : 201.2 Mean :12.26   
## 3rd Qu.:14.000 3rd Qu.: 108.0 3rd Qu.:16.00   
## Max. :19.000 Max. :3470.0 Max. :23.00   
##   
## size.kb box domain local   
## Min. : 0.00 Length:2171 Length:2171 Length:2171   
## 1st Qu.: 2.00 Class :character Class :character Class :character   
## Median : 4.00 Mode :character Mode :character Mode :character   
## Mean : 16.49   
## 3rd Qu.: 7.00   
## Max. :1337.00   
##   
## digits name cappct special   
## Min. : 0.000 Length:2171 Min. :0.0000 Min. : 0.000   
## 1st Qu.: 0.000 Class :character 1st Qu.:0.0600 1st Qu.: 0.000   
## Median : 0.000 Mode :character Median :0.1280 Median : 1.000   
## Mean : 0.591 Mean :0.1584 Mean : 1.397   
## 3rd Qu.: 0.000 3rd Qu.:0.2000 3rd Qu.: 2.000   
## Max. :23.000 Max. :1.0000 Max. :35.000   
##   
## credit sucker porn chain   
## Length:2171 Length:2171 Length:2171 Length:2171   
## Class :character Class :character Class :character Class :character   
## Mode :character Mode :character Mode :character Mode :character   
##   
##   
##   
##   
## username large text spampct category   
## Length:2171 Length:2171 Min. : 0.00 Length:2171   
## Class :character Class :character 1st Qu.:11.00 Class :character   
## Mode :character Mode :character Median :47.50 Mode :character   
## Mean :44.63   
## 3rd Qu.:76.00   
## Max. :99.00   
## NA's :1353   
## spam   
## Length:2171   
## Class :character   
## Mode :character   
##   
##   
##   
##

length(unique(which(is.na(spam),arr.ind = TRUE)[,1]))

## [1] 1353

length(unique(which(is.na(spam),arr.ind = TRUE)[,1]))/nrow(spam)

## [1] 0.6232151

答：描述性统计信息见上方，数据字典如下表： 3.2 建立一个新变量 domain.reduced 用以减少域名的domain类别为： “edu” , “com” , “gov” , “org” , “net” , 和 “other.”

repacedomain<-data.frame(spam$domain)  
domain.reduced<-apply(repacedomain,1,function(x){ifelse((x=="edu"||x=="com"||x=="gov"||x=="org"||x=="net"),x,"other.")})  
new\_spam<-mutate(spam,domain.reduced=domain.reduced)

3.3 将 spam 作为分类变量, 用解释变量：day of week, time of day, size.kb, box, domain.reduced, local, digits, name, capct, special, credit, sucker, porn, chain, username, 和 large text, 构建一个随机森林分类器，令 mtry = 2.

rf\_param<-new\_spam[,c(3:6,8:17,21,22)]  
colnames(rf\_param)[1]<-"DayOfWeek"  
colnames(rf\_param)[2]<-"TimeOfDay"  
library(randomForest)

## randomForest 4.6-14

## Type rfNews() to see new features/changes/bug fixes.

##   
## 载入程辑包：'randomForest'

## The following object is masked from 'package:dplyr':  
##   
## combine

## The following object is masked from 'package:ggplot2':  
##   
## margin

set.seed(1234)  
fit.forest<-randomForest(factor(spam)~.,rf\_param,importance=TRUE,mtry=2)

3.4 给出各个变量重要性排序结果；

sort(fit.forest$importance[,3])

## chain porn special username DayOfWeek   
## 0.000494093 0.001617788 0.003498310 0.003587755 0.003630741   
## cappct credit TimeOfDay name size.kb   
## 0.005361679 0.005938708 0.008756713 0.010335924 0.012440278   
## digits sucker local domain.reduced box   
## 0.013773915 0.027883971 0.044121553 0.060750760 0.095589174

答：下面是重要性由低到高排序的结果：chain、porn、special、username、DayOfWeek、 cappct、 credit、TimeOfDay、name、size.kb、 digits、 sucker、 local domain.reduced、 box

3.5 有多少非垃圾邮件被错判成垃圾邮件 spam?

fit.forest#答：90个

##   
## Call:  
## randomForest(formula = factor(spam) ~ ., data = rf\_param, importance = TRUE, mtry = 2)   
## Type of random forest: classification  
## Number of trees: 500  
## No. of variables tried at each split: 2  
##   
## OOB estimate of error rate: 6.31%  
## Confusion matrix:  
## no yes class.error  
## no 1371 90 0.06160164  
## yes 47 663 0.06619718

3.6作出预测类别对真实类别的散点图, 作出按随机森林返回的变量重要性排序作出解释变量的平行坐标图（a parallel coordinate plot）在ggobi中刷出非垃圾邮件被错分成垃圾邮件的案例，标出这些邮件的信息（如，全都来自local box，字节少等等），然后观察垃圾邮件被正确分类的数据，它们有什么特殊之处？

library(ggplot2)  
library(DescribeDisplay)  
library(showtext)

## 载入需要的程辑包：sysfonts

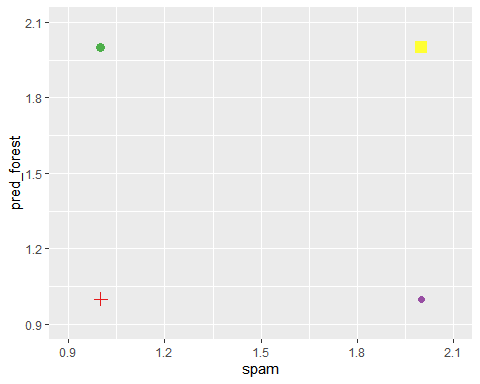
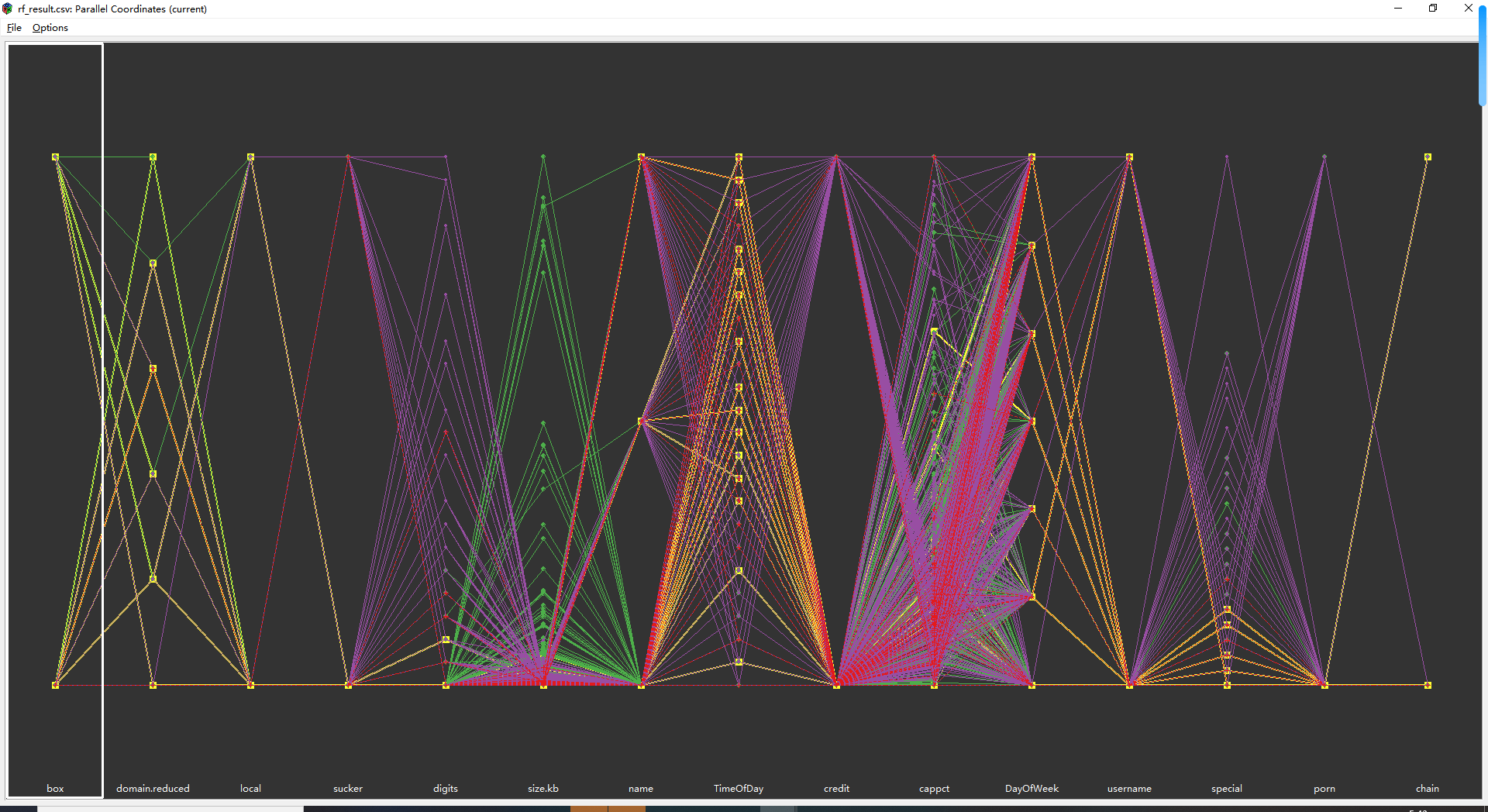
## 载入需要的程辑包：showtextdb

library(sysfonts)  
library(showtextdb)  
pred\_forest<-predict(fit.forest,rf\_param)  
rf\_result<-cbind(rf\_param,pred\_forest)  
write\_csv(rf\_result,"rf\_result.csv")  
d4<-dd\_load("../p\_t.r")  
p4<-ggplot(d4)  
print(p4)

## Warning in grid.Call(C\_stringMetric, as.graphicsAnnot(x$label)): Windows字体数据  
## 库里没有这样的字体系列

## Warning in grid.Call(C\_textBounds, as.graphicsAnnot(x$label), x$x, x$y, :  
## Windows字体数据库里没有这样的字体系列  
  
## Warning in grid.Call(C\_textBounds, as.graphicsAnnot(x$label), x$x, x$y, :  
## Windows字体数据库里没有这样的字体系列  
  
## Warning in grid.Call(C\_textBounds, as.graphicsAnnot(x$label), x$x, x$y, :  
## Windows字体数据库里没有这样的字体系列

## Warning in grid.Call.graphics(C\_text, as.graphicsAnnot(x$label), x$x, x$y, :  
## Windows字体数据库里没有这样的字体系列

答：非垃圾邮件被错分成垃圾邮件的案例信息：发件人并不在位于收件人的收件箱或发件箱中、域名为”com”、发件 email 是本地地址、主题行包含一个单词赚取，免费，保存、发件人姓名的数字个数较多、电子邮件的大小较小、如果主题行包括抵押,出售,批准,信贷之一、主题行中的大写字母较多、如果主题行不包含 pass, forward, help； 垃圾邮件被正确分类的数据的特点：发件 email 多数不是本地地址、主题行没有包含一个单词赚取，免费，保存、发件人姓名的数字个数较少、主题行没有包括抵押、出售、批准、信贷之一、主题行中的大写字母较少、大部分主题不包含收件人的姓名或登录名、主题中的特殊字符（非字母数字字符）数较少、主题行里没有 “ nude, sex, enlarge, improve”之一。 3.7 检查 Spam (真实类别) 和 Spam.Prob (可能被当成垃圾邮件的). 有多少不是垃圾邮件的被认定超过50% 可能是垃圾邮件（spam）?

Spam<-rf\_param$spam  
votes<-as.tibble(fit.forest$votes)[,c(1,2)]  
Spam.Prob<-apply(votes, 1, FUN = function(x){ifelse(x[2]>x[1],"yes","no")})  
matrixre<-as.tibble(cbind(Spam,Spam.Prob))  
matrixre%>%group\_by(Spam,Spam.Prob)%>%summarise(n=n())#答：有89个

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

## # A tibble: 4 x 3  
## # Groups: Spam [2]  
## Spam Spam.Prob n  
## <chr> <chr> <int>  
## 1 no no 1372  
## 2 no yes 89  
## 3 yes no 47  
## 4 yes yes 663

3.8 检查非垃圾邮件被随机森林分成垃圾邮件的案例的概率排名（probability rating ）.用一段文字描述具有高概率被当成垃圾邮件且被随机森林认为非常像垃圾邮件的案例。

a<-which(Spam=="no")  
b<-which(Spam.Prob=="yes")  
c<-intersect(a,b)  
prop<-apply(votes, 1, FUN = function(x){ifelse(x[2]>x[1],x[2],x[1])})[c]  
prop<-cbind(prop,c)  
prop[order(prop[,1]),]

## prop c  
## [1,] 0.5057471 346  
## [2,] 0.5101010 575  
## [3,] 0.5104167 133  
## [4,] 0.5182927 117  
## [5,] 0.5217391 907  
## [6,] 0.5222222 108  
## [7,] 0.5254237 538  
## [8,] 0.5279188 80  
## [9,] 0.5325444 1740  
## [10,] 0.5329670 306  
## [11,] 0.5376344 77  
## [12,] 0.5401070 1844  
## [13,] 0.5433526 47  
## [14,] 0.5492228 728  
## [15,] 0.5500000 906  
## [16,] 0.5567568 692  
## [17,] 0.5645161 155  
## [18,] 0.5707317 506  
## [19,] 0.5714286 1153  
## [20,] 0.5752688 734  
## [21,] 0.5820106 1195  
## [22,] 0.5833333 1360  
## [23,] 0.5852273 163  
## [24,] 0.5969388 753  
## [25,] 0.5977654 1214  
## [26,] 0.5989011 107  
## [27,] 0.5989011 546  
## [28,] 0.6000000 1189  
## [29,] 0.6033520 55  
## [30,] 0.6100629 1892  
## [31,] 0.6130952 78  
## [32,] 0.6149425 1209  
## [33,] 0.6169154 547  
## [34,] 0.6223404 121  
## [35,] 0.6268657 748  
## [36,] 0.6287129 76  
## [37,] 0.6304348 1846  
## [38,] 0.6368421 11  
## [39,] 0.6404494 102  
## [40,] 0.6464088 415  
## [41,] 0.6470588 852  
## [42,] 0.6551724 2167  
## [43,] 0.6560847 1271  
## [44,] 0.6596859 82  
## [45,] 0.6648936 50  
## [46,] 0.6770833 38  
## [47,] 0.6804124 104  
## [48,] 0.6806283 1  
## [49,] 0.6861702 918  
## [50,] 0.6956522 1344  
## [51,] 0.6994536 2004  
## [52,] 0.7050000 853  
## [53,] 0.7052632 1940  
## [54,] 0.7078652 1732  
## [55,] 0.7087912 879  
## [56,] 0.7173913 1474  
## [57,] 0.7195767 912  
## [58,] 0.7288136 1342  
## [59,] 0.7346939 97  
## [60,] 0.7393617 501  
## [61,] 0.7405405 103  
## [62,] 0.7417582 101  
## [63,] 0.7425150 49  
## [64,] 0.7595628 1898  
## [65,] 0.7664671 916  
## [66,] 0.7708333 895  
## [67,] 0.7790055 1414  
## [68,] 0.7828283 576  
## [69,] 0.7883598 1361  
## [70,] 0.7976190 1558  
## [71,] 0.8042328 1375  
## [72,] 0.8148148 1839  
## [73,] 0.8177083 1390  
## [74,] 0.8285714 66  
## [75,] 0.8313253 2005  
## [76,] 0.8315789 884  
## [77,] 0.8390805 1580  
## [78,] 0.8461538 1308  
## [79,] 0.8518519 1978  
## [80,] 0.8693467 1977  
## [81,] 0.8709677 1930  
## [82,] 0.8750000 1114  
## [83,] 0.8829787 2006  
## [84,] 0.8963731 899  
## [85,] 0.8969072 1479  
## [86,] 0.9141414 1301  
## [87,] 0.9505495 1358  
## [88,] 0.9540816 1347  
## [89,] 0.9838710 1416

rf\_param[prop[which.max(prop[,1]),2],]

## # A tibble: 1 x 16  
## DayOfWeek TimeOfDay size.kb box local digits name cappct special credit  
## <chr> <dbl> <dbl> <chr> <chr> <dbl> <chr> <dbl> <dbl> <chr>   
## 1 Wed 12 7 no no 0 single 0.1 1 no   
## # ... with 6 more variables: sucker <chr>, porn <chr>, chain <chr>,  
## # username <chr>, spam <chr>, domain.reduced <chr>

答：在周三12点发送的7kb大小的邮件只有一个名字主题行中的大写字母有0.1%，1个特殊字符，没有包括抵押、出售、批准、信贷之一，主题行包含一个单词赚取，免费，保存;域名为com的非垃圾邮件有98%的概率被识别为垃圾邮件。 3.9哪个用户的非垃圾邮件最有可能被当成垃圾邮件？

rf\_error<-cbind(prop[,1],spam[c,])  
rank\_user<-rf\_error%>%group\_by(isuid)%>%summarise(mean=mean(`prop[, 1]`))  
rank\_user[which.max(rank\_user$mean),1]

## # A tibble: 1 x 1  
## isuid  
## <dbl>  
## 1 9

3.10 根据你对数据的分析，你认为哪些变量在判断一个电子邮件是否垃圾邮件时最重要？ 答：我认为name、size.kb、 digits、 sucker、 local domain.reduced、 box这些变量比较重要。