R语言基础

NA: for missing or undefined data，有这个数字，可是获得不了，比如我的头发（真的-很难数嘛？）

NULL: for empty object (e.g. null / empty lists)，没有的数字，压根找不到，比如海龟根本没有头发这个概念

NaN: for results that cannot be reasonably defined

.libPaths() # get library location； library() # see all packages installed； search() # see packages currently loaded

获取帮助：help.start()；# help about function foo：help(foo)/?foo；exmaple()；args()

list all functions containing string foo：apropos("foo")；# show an example of function foo：example(foo)

builtins()列出所有的内置函数

%% in R = % in Python；%/% inR = // in Python, 不然会返回float; TRUE/T in R = True in Python

%in%是in的意思，但是要把list转化成vec。unlist()。再使用

data type：integer(5L)/continuous/categorical (norminal/ordinal)/ text

matrix里面只能有一个datatype，dt和df没有这个限制

options('scipen',100)显示100条向量值

定义函数：x=function (x) { return (T)}

探索性数据分析的心法

1、商业问题是什么

2、我需要知道数据的什么，来帮助理解并回答商业问题，找到商业机会

3、具体用R怎么实现

探索性数据分析的目的：

1、将数据与商业问题结合。数据充足吗？合适吗？比如没有预测性数值，类别变量，冗余

2、探测数据的问题。数据质量，异常值的监测

基本语法

c("a",1)会返回'a','1'，因为char比numeric更高级，而vec会保证元素的类型全部一样

c(1,2,3)*c(1,2,3)/c(1,2,3)^2=c(1,4,9)

class(c(TRUE))返回的是logical, class(dt)返回datatable; length()返回变量数量，length("TTT")返回1

str() check classification of viriables，检查-数据框中有哪些数据，包括类型和数值

summary()可以用于检查数据错误：给出数值型变量的数值summary，min，max，median，mean；给出字符串向量的长度和class；factor向量会给出各个因子的count

向量，矩阵，df，dt初始化：c()，matrix(data=NA,nrow,ncol)，data.frame("col":vec)，data.table(vec)

向量添加元素：vec <- append(vec,element)；vec=c(a,a,a)会得到一个flat的向量vec

重命名列：names(df1) <-"col"/setnames(df$col,old=c(xxx),new="xxx")/colnames(dt)=c('1','2','diff')

删除列：col=NULL/df[-c(1,2)]

把两列或者两个别的什么东西粘在一起：col=paste(col1,col2, sep='-')；paste(1,"a",T)

独特的值：unique(), nrow()

行列求和：colSums()/rowSums()

ifelse语句：ifelse(expression, 0, 1), a=1; b=if(a==2) 'a' else 'b'

a=1; if(a==2) b='a' else b='b'

#返回对应索引的值：switch(3, “apple”, “orange”, “pear”, “pineapple”) / switch(“beta”, “alpha”=“Big Rock”, “beta”=“Meteorite”, “gamma”=“Red Stones”)

切片索引：by address df[c(3:7),seq(1,ncol(df),2)]

by value：df[(df>=10) & (df<=20)]

赋值：df[2:4,1]=c(3,9,7), df[c(3,5), 2:4]=rbind(c(1,3,5), c(2,4,6), c(7,9,11))，df[1]=df[,1]

批量赋值：df[df<0]=-999。所有满足条件的值都被赋一个值

基本语法 (cont)

R语言一直在原数据上更改，而python未-必，这就是python会报错要求你去用iloc的原因

df[,2][df[,2]<0]=-777；可以先选择列再选择行

df[df[,1]<0 & df[,2]>0,]=cbind(9,8)

数据预处理

数据清洗的目的一是business object，二是-technical requirements

新建一列，清洗完之后和原列作比较

主要目的是防止清洗出错

缺失值的处理（删除/填充）

找到缺失的原因，这能决定如何填充缺失值

分类值预测：整列的mode，subgroup的mode，cart，逻辑回归

连续值预测：随机缺失的值可以用mean填充；系统性缺失的值可以用subgroup的mean，或者cart/逻辑回归；离散化连续值的列，当成分类值处理

删除一整行只是最后的选择

logistic/linear模型会自动忽略缺失值的行

is.na(dt)会显示所有的格子，which是不是空值

sum(is.na())只会得到一个值

which(is.na(data))能得到空值在第几行, which(h$counts==max(h$counts))

na.omit(ins.dt)

错误值

将错误值替换成正确的值

预测错误值，让他们变的更正确

把它们变成na，交给cart去预测

删掉一整行

数据不一致/数据重复

找到数据不一致的原因，尽量从源头解决问题

数据重复的定义，往往取决于主键

data exploration R

#rm(list=ls())，ls()列出所有的变量

读数据：read.table("csv", skip=7, header=T, sep=",", row.names="id",nrows=4)；

read.csv("csv", stringsAsFactors = TRUE)。stringsAsFactors把字符串转化为因子变量	fread("csv",, na.strings = c("NA", "missing", "N/A", -99, "", "m", "M", "na", "."))

写数据：write.csv(df,path, row.names=F); write.table(df,path)

rbind(df,df2), cbind(df.df2), names(df)[3:4]=c('a','b')

查看导入的数据集：head(df),view(df), 图形交互

数据维度：dim(df)	数据列的细节，如有多少缺失值，各个类别有几个数：summary(df)
查看子集：subset(df, is.na(df$col, select))。select选中了数据集中的某列	train <- sample.split(Y = mtcars$mpg, SplitRatio = 0.7) trainset <- subset(mtcars, train == T) testset <- subset(mtcars, train == F)

比较两个向量或者df是否相同：identical(df-1,df2)

为数据集增加噪音：jitter(df$col)，jitter只能给连续变量加噪音，logical和类别变量不行

#生成sequence，重复数字，反转列表：seq(10,20,3)，rep(10,5)，rev(v),sort(v)升序,unique(v),

vector(c, n) – returns vector with all values c of size n	names(vec)=c('a','b','c') 可以给vec的元素取名字

data exploration R (cont)

matrix(c, nrow=5, ncol=3) – returns a 5□3 matrix with all values c

data.frame(v1, v2, v3…) – returns a data frame made up of column vectors v1, v2, v3,

所以要转换类别：as.numeric(df$col)	as.integer()=int()in python
对每一列都执行某样操作：sapply(df, func, na.rm=TRUE)。Possible functions used in sapply include mean, sd, var, min, max, median, range, and quantile	sapply(v, function(x) if (is.na(x)) 999 else x)
lapply(my_list, function(x) x == element))	lapply和sapply的区别是返回结构不同，lapply返回list，sapply返回向量和矩阵
#见堆叠柱状图：table(deparse.level=2)/prop.table(col1,col2,margin)能把两列数据组合成透视表	deparse.level=2会展示所有行和列的名字
所有的factor：levels(factorcol, ordered=T, levels=c(xxx), labels=c(xxx))	labels把向量里的数值映射到一个新的空间，它和levels的区别是levels只涉及input，labels涉及到output

relevel(x, ref)将ref因子放在x的第一个

切分左开右闭区间：limits=c(-1，2，3，4，5)

生成新的一列，值是区间：dt[, new_col := cut(col, breaks=limits, include.lowest=T)]

data exploration R (cont)

cor计算相关系数，the degree of the consistensy of the trend of the relationship	解读时了一说，一个上升时，另一个倾向于xxx，但因果关系我们并不清楚
cor(dataframe)可以计算相关性矩阵，"corrplot"；corrplot(cor(mtcars), type = "upper")	cov计算协方差，-与cor（相关系数不同）

ceiling(2.5)=3

随机数

var(), sd()（标准差）, skewness(),kurtosis()

quantile(data,c(0.025,0.975))

sample(1:80,80,replace=F)；第一个是参数，第二个是size，第三个是能否重复

dbinom(x,n,p)，pbinom。x次成功from n次，每次概率为p。x可以是1：10，成功1：10次

dpois(x,lambda), ppois(x,lambda)

dnorm(x,mean,sd)，pnorm(x,mean,sd)

runif(size, low, high), rnorm(size, mean, sd 而不是方差), rbinom(size, times, possibility)

set.seed(seed) fixes the random result of ramdom function

curve(dnorm(x,mean=0,sd=1),col="red",from=-3,to=4,xlim=c(-3,4),ylim=c(0,1)); curve(dnorm(x,mean=1,sd=1),col="blue",-add=T)

t.test(x, y = NULL, alternative = c("two.sided", "less", "greater"), mu = 0, paired = FALSE, var.equal = FALSE, conf.level = 0.95)

如果x也有值，y也有值，那就是计算x-y的置信区间

prop.test(x, n, p = NULL, alternative = c("two.sided", "less", "greater"), conf.level = 0.95, correct = TRUE)

这里x可以是向量，两次trial的成功次数；n是总次数，同理。算出来就是做差值

ggplot2

data层：ggplot(data,aes(x=colname,y=colname, fill=factorcolname))

图像层：geom_point()

布局层：facet_grid(.~fl)。.~fl是指分组的colname

将factor变量映射到颜色：scae_fill_manual(values=c("0"="dark blue", "1"="orange")

细节层：labs(title, xlab,ylab)

visualization with R

par(mfrow=c(2,2))，两行两列的子图

plot(xaxt="n",yaxt="n");抹掉所有坐标轴

axis(1,at=seq(0.5,length(),1),labels=names(),tick=F,col="red")1是x轴，at决定位置，labels决定具体显示，包括内容和间距，tick控制有没有坐标线，col决定标线颜色

label太多，可调整字体方向：par(las=0,1,2,3)；0=parallel, 1=all horizontal, 2=all perpendicular to axis, 3=all vertical

图片太大，可调整图与周边margin：par(mar=c(5,4,4,2)+0.1)，margin的顺序是下，左，上，右；单位是line

允许画图画到外面去：par(xpd=T)。

添加legend：legend("topright", inset=c(0,0), fill=c("red","grey"), legend=rownames(counts), border="grey", cex=0.6)

第一个是大体位置，inset是具体位置，fill是对对应的颜色，legend是图例名字，border是边框颜色，

散点图和箱线图的区别是：散点图可以告诉你样本量的大小（比如50岁左右的人有保险的比没有的多），而箱线图不行

plot(density(df$col), xlab,ylab,main)

hist(df$col, ylim=c(0,220), breaks=c(-10,0,10,20), labels=T, col="light blue") labels会给每个柱子加上数字；默认每个区间是左开右闭的

visualization with R (cont)

boxplot(df$col ~ df$catecol)；use $stats查看两个箱线图计算出的几个critical数据，从上到下依次递增的数据

箱线图的数据依次是：Q1-1.5IQR，Q1，median，Q3，Q3+1.5QIR。inter-quatile range简称IQR。在box-and-whisker method里，其余的都叫做异常值

柱状图，展示类别变量的分布：barplot(table(df$col),col=c("light blue","mistyrose","lightcyan","lavander"), horiz=T, cex.names=0.5)。

堆叠柱状图：data=table(df$ycol,df$xcol)。row index是ycol的各个数值，col index是xcol的各个值。barplot(data, col=c("red","grey"))

百分比柱状图(主要是数据预处理的不同)：prop.table(df$ycol,df$xcol, margin=1/2)。margin=1是横着计算百分比，2是竖着计算百分比

散点图：plot(df$xcol1,df$ycol2)； 在散点基础上加一根smooth curve。scatter.smooth(df$col1,df$col2, col="grey")；col决定散点的颜色

散点曲线矩阵图，查看各个变量间的关系，分辨哪个最先分析：pairs(~ col1+age+sex++...., panel=panel.smooth, span=0.75, data=df)。panel.smooth是加上平滑曲线，span越大，线性程度越高

png(),jpeg()+dev.off()能存储图片

画完图，sys.sleep(0.05)

绘制空白图

plot(x = c(22, 28), y = c(1, 1000), type = "n", xlab = "", ylab = "") # set up a blank plot with specified ranges

data.table

data.table中，j参数里，：=的结果within在dt中，=在out新建一个dt

创建data.table：dt[, .(.N, colname=sum(col==2), prop.uninsured=sum(col==2)/.N), keyby=colname]

keyby和by的区别是：keyby会sorting分组的结果。可以有多个分组标准.(col1,col2)

data.table (cont)

.N是指列名是number，值是count

线性回归

多重共线性不影响预测，只影响模型解读。同时解读模型时要假设其他变量不变。不能反写成x的等式，因为这不是代数，是统计模型

assumption1：linear assosiation between y and x

assumption2：error has a normal distribution with mean 0

assumption3：errors与x互相独立，并且有常数的standard deviation

lm(y~x1+x2或者., data=data)；m4 <-step(m.full);赤潮信息准则

coef(model); confint(model)

abline(m1, col = "red")；identify(x = mtcars$wt, y = mtcars$mpg)

win.graph()；identify(x = x_data, y = y_data)

R suqare代表了模型的解释力。about xxx% of the data canbe explained bythe model。只要增加变量数，R方会一直上升，因此不能简单用R方来比较两个线性模型

adjusted R square：惩罚每一个被添加的变量。在多变量的前提下，用这个比r方要好

plot（lm函数的结果）来检验假设1，2，3

左上角残差图：test1，2。理想情况是y=0的一条红线。residuals(m5)

右上角qq图：test2.理想情况是沿着虚线

左下角经过标准化后的残差图：test3。理想情况是上下均匀地分布在一个矩形里，而不是随着x的增大而改变

右下角：展示influential outliers。有木有influence主要指去掉这个点对拟合曲线的影响有多大

单变量离群点很好辨认，超过两个变量散点图就不能用了，所以得用cook统计量，点落到在虚线外就是influential

线性回归 (cont)

处理离散类别变量，要注意用r转换成factor。然后哪怕是有序变量，数字本身也没有意义，不能当成连续变量来处理

k个类，就有k-1个变量，其中一个会是baseline，baseline自动是字母顺序表里最早的那个，然后也可以用relevel（ ）自定义

如何决定选择哪些变量：

1、专家，领域知识

2、统计知识：pvalue小于5%,前向，后向选择，双向选择，降维方法，CRT等

多重共线性：一个x能被其他x线性表出，意味着这个x的信息被其他的包含进去了。因此dummy variable要减去1

vif=1/（1-Ri^2）。^{Ri}2就是以这个x为因变量，-其他x为自变量回归得出的R2。vif(lm的return)

一版vif>5或者10，有dummy variable的模型一版gvif>2。使用vif模型from package car

预测未来值：predict.m5.test <- predict(m5, newdata = testset, type='response')response 返回y=1的概率值，

提取p值：summary(model)$coefficients[,4]

逻辑回归（预测分类变量）

glm(y~x, family=binomial, data=data)

1、建模预测分类变量

2、如何辨别高风险因子

3、双变量分类模型

4、odds（胜利/失败概率，chances），在代数中等于$e^{z,是个function，odds ratio代表每个bk的作用，e}$bk，是个常数

连续变量xk增加一个单位，胜率会怎么增加，会乘以e^bk

类别变量xk从baseline跳转到一个类，胜率-也会乘以e^bk

判断一个xk变化会怎么影响odds ratio，可以用置信区间，2.5%-97.5%的区间超过1且不包含1则大于1

5、multinomial（超过3个类别）

multinom() function from nnet Rpackage

cluster

library(cluster)

km2=kmeans(pts,centers=center)#初始的中心

clus1=pts[km2$cluster==1]

agnes(rivers),plot(that)