~\Downloads\R基础和基本统计.r

```r
#修改C盘临时文件夹下面中文的名称问题
#https://zhidao.baidu.com/question/748080860947976012.html
#安装包 options(warning=-1)
Sys.setenv(LANGUAGE= "en")
#https://rdrr.io/snippets/ 网络R
#初学者很好的学习材料
#https://tianyishi2001.github.io/r-and-tidyverse-book/what-is-R.html#youshi
#https://www.rdocumentation.org/
#搜R语言和来自CRAN，GitHub和Bioconductor上的近18000个包的所有的函数的说明和使用例。
install.packages("vegan")
#install.packages(c("vegan","hier.part"))
library(vegan)
#require(vegan);citation("vegan")
cca
?cca
example(cca)
?hier.part
#如果两个??,加函数还是搜不到，用google, bing,百度
#在当前R里面获取任何相关字节
getAnywhere(t.test)
#R官网的初学者材料
help.start()
help(package='vegan') #显示已经安装的包的描述和函数说明
RSiteSearch("pca") #在官方网站上联网搜索
Sys.getenv('R_HOME') #报告R主程序安装目录
(.packages())#查看工作空间已加载的包

#向量vector
a <- c(1, 2,5, 3, 6, -2, 4)      #  数值型向量
b <- c("one", "two", "three")    #  字符型向量
c <- c(TRUE, NA, FALSE)          #  逻辑型向量

#因子factor
province <- c("四川", "湖南", "江苏", "四川", "四川", "四川", "湖南", "江苏", "湖南", "江苏")
pf <- factor(province)    #  创建 province 的因子 pf
pf
#向量选择
a <- c(1, 2, 5, 3, 6, -2, 4)
a[3]
a[c(1, 3, 5)]
a[2:6]
a[a>3&a<5]
a[-1]
a[-1:-3]
a#产生向量
rep(2:5, 2) #  等价于  rep(2:5, times = 2)
rep(2:5, each=2)
rep(1:3, times = 4, each = 2)  #这个在自己编写差异表达分析的分组信息时可以用到
seq(1,10,0.1)
#想一下，如何保存这些系列呢，赋值！
#0.1-1，间隔0.1，重复1：10

#矩阵matrix必须是同一属性数据，如果不是，会自动调整为同一属性，vector类同
matrix(1:4, nrow = 2, ncol = 2)   #默认所有元素是按列向
(mat1<-matrix(1:12,6, byrow=T))
mat1
```

```r
mat2<-matrix(1:12,3,4, byrow=F)
mat2
x <- matrix(1:6, 2, 3)
x[2,2]
x[2,]
x[,2]
x[1:2,2:3]
## %*% does matrix multiplication
m <- t(x)     # Assign m the transposed m
x %*% t(x)    # %*% does matrix multiplication
m * m         # * 哈达马积(hadamard product)
#数组
a <- array(data=1:18,dim=c(3,3,2)) # 3d with dimensions 3x3x2
a <- array(1:18,c(3,3,2))          # the same array
a
#选取数字为17怎么选

#数据框data frame
patientID <- c(1, 2, 3, 4)
age <- c(25, 34, 28, 52)
diabetes <- c("Type1", "Type2", "Type1", "Type1")
status <- c("Poor", "Improved", "Excellent", "Poor")
patientdata <- data.frame(patientID, age, diabetes, status)
patientdata
patientdata[1:2,] #  第 1、2 行的所有元素
patientdata[,1:2]
patientdata$age    #“$”符合用于选取一个指定的变量
patientdata[,"age"] #试一下去逗号

patientdata[5,]=c(4,52,"Type1","Poor")
unique(patientdata) #去掉重复的行,常用于数据清洗
#但是不要乱加
a=unique(patientdata)
a$age #现在的a$age这个向量里面都是字符，而不是数值了，因为前面patientdata[5,]这一步操作相当于加的是一个字符型的向量。因为向量里面的元素类型应该是一致的。

# 列表list
#list是储物箱，list里面可以装list，可以无限延展。
g <- "My First List"
h <- c(25, 26, 18, 39)
j <- matrix(1:10, nrow = 5)
k <- c("one", "two", "three")
mylist <- list(title = g, ages = h, j, k,patientdata)
mylist[[3]] #提取list的第3个元素
mylist[3]   #也是提取的list的第3个元素，但提取出来后，仍是作为列表的第1个元素。少了一个套，相当于没有提出真正的元素。
#观察一下有什么不同吗？
#用class来判断一个对象的类型。 class()


#工作目录
getwd()
#  导入 text 和csv 文件，当前工作目录下
# mydata2 <- read.csv("oats.csv")
mydata2 <- read.csv("./learning/lai_jiangshan/data/oats.csv")
mydata2

# mydata3 <- read.table("oats.txt", header=TRUE)
mydata3 <- read.table("./learning/lai_jiangshan/data/oats.txt", header = TRUE)
mydata3
```

```r

mydata4 <- read.csv(file.choose(), header=T)
mydata4
class(mydata3)

#mydata4 <- read.csv(file.choose(), header=T)
#row.names=1
#读excel表格，两个方式任选一种
#install.packages("openxlsx")
library(openxlsx)
# mydata <- read.xlsx("oats.xlsx",sheet = 1)
mydata <- read.xlsx("./learning/lai_jiangshan/data/oats.xlsx", sheet = 1)
mydata
class(mydata)
#如果是openxlsx不行，尝试一下readxl函数
#install.packages("readxl")
#library(readxl)
#mydata <- read_excel("oats.xlsx",sheet = 1)
#mydata
#mydata=as.data.frame(mydata)
#class(mydata)
#http://www.milanor.net/blog/read-excel-files-from-r/ 读excel数据网站

#加盘符""，其他类型数据输入看ppt
#install.packages("foreign")
library(foreign)
x <-read.spss("sps.sav", to.data.frame=T) # Read file sps.sav
x
#尝试一下去掉to.data.frame=T，读入是列表
#https://cran.r-project.org/doc/manuals/r-release/R-data.html
#http://rsoftware.h.baike.com/article-1867733.html"
#练习读入数据你们自己的数据，发现问题

getwd()
#2.3 数据的存储
write.table(mydata2, "D:/fg.txt", row.names = F)
write.csv(mydata2,"D:/fg.csv", row.names = F)

library(openxlsx)#优势，可以保留列名重复
write.xlsx(mydata2,"D:/fg.xlsx")
?write.xlsx
save(mydata2, file = "D:/fg.Rdata") #练习一下多个文件,这个"file ="不能省略
save(mydata2, x, mydata, file = "D:/fg.Rdata") #save 可以存储多个文件,非常有用,
#而且这种格式“.Rdata”直接打开，不用read.table, read.csv之类的函数来读取；另一个好处是可以保存其他数据类型用于下一次直接打开，vevtor, matrix, array都可以
#而read.csv(), read.txt()只能读入数据框data frame。
#所以，如果不是为了给其他人数据，以后都用save()来存.Rdata数据，这个非常有用啊。
#载入.Rdata数据，用"load()"函数即可。

#fix()这个函数可以直接把某个数据调出进行修改,不过服务器上似乎用不了。
fix(mydata2)

ls() #显示当前工作目录下所有对象，包括：数据、函数等


#3 数据的排序（order)
head(mydata2)
# plyr 包的 arrange( )函数
library(plyr)
arrange(mydata2, yield)
```

```r
arrange(mydata2, Variety, -yield)
arrange(mydata2, desc(Variety), -yield)
#https://stackoverflow.com/questions/1296646/how-to-sort-a-dataframe-by-multiple-columns

#数据集的合并
#案例分析
# chinaIUCN<- read.csv("chinaIUCN.csv")
chinaIUCN <- read.csv("./learning/lai_jiangshan/data/chinaIUCN.csv")
head(chinaIUCN)

# focussp <- read.csv("shennongjia.csv") #如果文件中有中文，那么用记事本保存为"UTF-8"编码，再在read.csv()函数中加是“encoding = "UTF-8"”
focussp <- read.csv("./learning/lai_jiangshan/data/shennongjia-UTF-8.csv", encoding = "UTF-8")
head(focussp)
bb <- merge(chinaIUCN, focussp, by.x = "species", by.y ="name", all.y=T)
head(bb)
class(bb)
#练习，结果存为CSV和R的格式

#多个数据列是否可以
#http://rstudio-pubs-static.s3.amazonaws.com/13602_96265a9b3bac4cb1b214340770aa18a1.html

#数据结构查看
data() #查看R里面的自带数据集dim(iris)    #  数据集的维度，有多少行多少列？
names(iris)  #  数据有哪些列？
str(iris)    #  数据的结构如何？
iris[1:5,]  #  查看数据的前 5 行
head(iris)   #  查看数据的前 6 行
tail(iris)   #  查看数据的最后 6 行
class(iris)  # 查看类别

#巧用批处理，对所有行/列作同一种运算。
sapply(iris[,-5],mean,na.rm=T)
apply(iris[,-5],2,mean,na.rm=T) #margin=2对列
apply(iris[,-5],1,mean,na.rm=T) #margin=1对行

#tapply第一个参数为统计的对象，第二个参数为分组的因子，第三参数为统计函数
tapply(iris$Sepal.Length,iris$Species,var) #针对某一列的作一 种统计分析，可以按一个或多个因素分组，分组用list(),这一个或多个因素都会强行分类，
#如果某个因素组合下没有统计结果，就用“NA”表示。最终生成一个长行表。
aggregate(iris[,-5],list(iris$Species),sd) #可以对多列进行一种统计分析,分组用list()，只保留有统计结果的组合，但最终data.frame没有列名，最终生成长列表

#不同的列做不同的运算，即多列数据进行多种统计运算，这个最好，包罗前面的，同时生成的结果data.frame也有列名，代码量也小一些。
library(plyr)
ddply(iris,.
(Species),summarise,sum.a=sum(Sepal.Length),mean.c=mean(Sepal.Width),pl.sd=sd(Petal.Length))

#双分组因子
library(vegan)
data(dune.env)
head(dune.env)
ddply(dune.env,.(Management,Use),summarise,sum.a1=sum(A1))

#处理多变量的分组
#我是xxx,最近在整理大样地的数据。刚好读了您的博文《将样方多度数据转为“样方-物种“矩阵》，有个问题想请教一下老师。
#我现在的数据有两列，一列是样方号（site）,一列是物种名(species)， 怎么转换成三列数据样方号site，物种名spieces，物种多度abund。
#我的部分数据见附件。
```

```r
#dfx <- read.csv("样地原始数据.csv")
dfx <- read.csv("./learning/lai_jiangshan/data/样地原始数据.csv")
head(dfx)
#方法1，tapply()，单列单运算法，可以分组
zqdata <- tapply(dfx$abund,list(dfx$site,dfx$species),sum)
head(zqdata)
#把NA变为0
zqdata[is.na(zqdata)]=0
#ifelse()和“vegan”包中的decostand()两个函数可以转换原始数据为0、1数据。decostand()主要是用于数据转换的函数。
#其他可能用到的转换数据：ifelse(zqdata > 0, 1, zqdata)

#方法2，aggreate()，多列单运算法，可以分组
zqdata1 <- aggregate(dfx[ , 3], list(dfx$site, dfx$species), sum)
head(zqdata1)

#方法3，ddply()，多列多运算法，可以分组
zqdata2 <- ddply(dfx, .(site, species), summarise, sum.abund = sum(abund))
head(zqdata2)

#思考一下如何把0的地方变为NA?
zqdata[zqdata == 0] <- NA

############################
install.packages(c("adephylo","adespatial", "FD", "ggpubr"))
####################

save(zqdata,file="D:/ourresult.Rdata")
#练习存储你自己的数据和计算平均值
#options(warn=-1)
#描述统计
#https://www.statmethods.net/stats/descriptives.html
#install.packages("pastecs")
library(pastecs)
#Sepal.Width花萼宽度
x <- iris$Sepal.Width
stat.desc(x)
#x可以是矩阵
#标准误等于标准差/样本量的平方根/置信区间值为qt(0.975,150-1)*SE
0.43586628/sqrt(150)
qt(0.975,150-1)*0.03558833 #计算置信区间qt(0.975,n-1)*SE
#多个物种的情况
tapply(iris[,2],iris[,5],stat.desc)
#练习，样本量为100，平均值为5，标准差为0.6，求平均值95%的置信区间
#尝试一下，是否可以给矩阵matrix或数据框data.frame
#Hit <Return> to see next plot:  par(ask=F) #
#histogram
x1 <- iris$Sepal.Width
h<-hist(x1, breaks=10, col=2, xlab="",main="Histogram with Normal Curve")
hist(x1,main="Histogram of observed data",prob=T)##probability densities: h$counts/150/diff(h$breaks)
#Kernel Density Estimation
lines(density(x1),col=2,lwd=3,lty=2)

#par(ask=F)
plot(density(x1),col=1,lty=1,lwd=2)
#dorm后面这个x是个表达式
curve(dnorm(x,mean=3.057333,sd=0.4358663),add=T,col=4,lwd=3,lty=2)
#for normality test
qqnorm(x1)
```

```r
qqline(x1)

#线上发散，线下收敛
#W = 是观测与实际重叠的面积
#http://www.real-statistics.com/tests-normality-and-symmetry/statistical-tests-normality-symmetry/shapiro-wilk-test/
#样本量必须在3-5000之间
?shapiro.test
shapiro.test(x1)
shapiro.test(log(x1))
#17 popular distribution
?distribution
dnorm(0,mean=0,sd=1)
pnorm(1.96)
qnorm(0.975)
rnorm(10)
#dname( )   density or probability function
#pname( )   cumulative density function
#qname( )   quantile function
#rname( )   random deviates
#Probability Plots
#https://www.statmethods.net/advgraphs/probability.html
#分布拟合
#https://stats.stackexchange.com/questions/132652/how-to-determine-which-distribution-fits-my-data-best
#options(warn=-1)

x=iris$Petal.Width
#看一个定量向量是否属于某一个分布，先拟合求参数，再把参数代到ks.test里面
library(MASS)
#把x作为gamma分布进行拟合，得到参数
fitdistr(x,'gamma')
#用上一步得到拟合参数，作为真正的gamma分布，与实际的x向量值进行比较，确定x是否是gamma分布
ks.test(x,'pgamma',1.5578249,1.2989101)
fitdistr(x,'poisson')
ks.test(x,'ppois',1.19933333)

#练习，检验你们的数据是否属于正态分布

#########单样本T检验#########
#杨树的树高理论值为8m
height<-c(8.0, 7.9,7.9,8.1, 8.2, 8.3, 8.3, 8.5, 8.6, 8.8, 8.3, 8.5,8.7,8.7,
          8.7, 8.8,7.2, 8.0, 7.5, 7.6, 9.0, 8.7, 8.5, 8.8, 8.7, 8.6, 8.6,
          8.5, 8.2, 8.2, 8.0, 8.3)
shapiro.test(height)
#t=(x平均值-mu)/se
t.test(height, mu = 8, alternative = "two.sided")
#结果为: 2*（1-pt(4.5186, 31), 2*(1-pt(mu, df))
#t.test(height, mu = 8, alternative = "greater")
#t.test(height, mu = 8, alternative = "less")
sink("test.txt")
t.test(height, mu = 8, alternative = "two.sided")
sink()
#sink()函数一定是要闭合，跟括号一样,并且不会在屏幕上显示
#练习读入你们自己的数据，然后进行单样本T检验
#change rank to test威尔科克森秩检验（非参数）
# c("two.sided", "less", "greater")
wilcox.test(height, mu = 8)

############ chisq test ############
```

```r
freq = c(22,21,22,27,22,36)
prob = rep(1,6)/6
chisq.test(freq,p=prob)
#卡方检验都检验一切计算实测与预测值之间是否显著差异检验
#sum((freq-期望值)^2/期望值)
#1-pchisq(6.72,5)

#set union, intersection, (asymmetric!) difference, equality and membership on two vectors.
x <- 1:10
y <- 5:14
#并集
union(x, y)
#交集
intersect(x, y)
setdiff(x, y)# which in x, not in y
setdiff(y, x)
#ks.test()
######### 两个样本平均值T检验#########
#whether is mean of A equal to mean of B
A <- c(79.98, 80.04, 80.02, 80.04, 80.03, 80.03, 80.04, 79.97,
80.05, 80.03, 80.02, 80.00, 80.02)

B <-c(80.02,79.94, 79.98, 79.97, 79.97, 80.03, 79.95, 79.97)
#ks.test()
################################################################################
#t.test的正态分布+方差齐性检验
shapiro.test(A)
shapiro.test(B)
var.test(A,B)
dev.off()#删除当前的做图窗口
boxplot(A,B)
#零假设为差值=0，等价于95%置信区间是否互相包括对方的均值
t.test(A,B,var.equal=T)
#零假设设定为A-B=0.02
t.test(A,B,mu=0.02,var.equal=T)
#var.equal =T，加这个参数，统计结果更容易显著

#if no norm 威尔科克森秩检验
wilcox.test(A,B)

#如果不正态，用置换检验的比较即可
library(RVAideMemoire)
perm.t.test(A,B,nperm=999)
#成对T检验
#Testdata=read.csv("ravens.csv", header=T)
Testdata=read.csv("./learning/lai_jiangshan/data/ravens.csv", header=T)
Testdata
t.test(Testdata$after, Testdata$before, paired=T)
t.test(Testdata$after, Testdata$before)

#查看变量之间的相关性
#Practice Correlations
?cor
cor(iris$Sepal.Length, iris$Sepal.Width)
cor.test(iris$Sepal.Length, iris$Sepal.Width)

plot(iris$Sepal.Length, iris$Sepal.Width)
cor.test(iris$Sepal.Length, iris$Sepal.Width,method= "spearman")
cor(iris[ ,1:4]) #default pearson

```

```r
# 用相关系数算显著性
#对数据不要求正态分布和方差齐性
r=-0.1175698;n=150
#p值的算法
(1-pt(abs(r)*sqrt((n-2)/(1-r^2)),n-2))*2

#多变量的相关
library(psych)
corr.test(iris[ ,1:4])
cord <- corr.test(iris[ ,1:4],adjust="none")
cord$r #提取相关系数
cord$p #提取p值
(domG <- read.csv("./learning/lai_jiangshan/data/domG.csv",header=T))
(enzydomG <- read.csv("./learning/lai_jiangshan/data/enzydomG.csv"))
#计算酶与微生物的相关
library(psych)
pvalue <- corr.test(domG,enzydomG,adjust="none")
pvalue
round(pvalue$p,4)
library(corrplot)
dev.new()
corrplot(pvalue$r, method = "circle")
#?ccf #研究时间系列相关cross-correlation (互相关)
# pairs() is a function to plot a matrix of bivariate scatter plots.
# panelutils.R is a set of functions that add useful features to pairs():
# upper.panel=panel.cor: to print correlation coefficients in the upper panel,
# with significance levels;
# diag.panel=panel.hist: to plot histograms of the variables in the diagonal.
# Specify method for the choice of the correlation coefficient:
# by default, method = "pearson", other choices are "spearman" and "kendall".
# To get a plot in grey tones instead of colors, use no.col=TRUE and
# lower.panel=panel.smoothb.

source("./learning/lai_jiangshan/data/panelutils.R")
#去掉趋势线，请去掉 lower.panel参数
pairs(iris[,1:4], lower.panel=panel.smooth, upper.panel=panel.cor,
      diag.panel=panel.hist, main="Pearson Correlation Matrix",method="pearson")
#https://cran.r-project.org/web/packages/corrplot/vignettes/corrplot-intro.html
#http://uc-r.github.io/correlations
#install.packages("GGally")
library(GGally)
ggpairs(iris[,1:4])


#Partial Correlation 偏相关
# data
library(ppcor)
data=read.csv("./learning/lai_jiangshan/data/pcordata.csv")
data
# partial correlation between "x" and "y" given "v1"
#做剔除v1,v2影响后的x和y的相关
pcor.test(data$x,data$y,data$v1)

#attach()
a=resid(lm(x~v1,data))#回归获得残差
b=resid(lm(y~v1,data))#回归获得残差
cor.test(a,b)
cor.test(data$x,data$y)


```

```r
#
###图形输出
#ggplot2包下面的函数，只能保存ggplot2刚画出来的图
ggsave("./project/Temp/lai.pdf")

#三步：创建画布，画图，关闭画布
pdf(file="output.pdf")   #输出文件名为output 的pdf文件
 par(mfrow=c(2,2))
 plot(1:10)
 plot(1:10)
 plot(1:10)
 plot(1:10)
dev.off()   #关闭pdf 绘图设备 输出的图形请在工作路径下查找
#拼接pdf的网站
#https://smallpdf.com/cn/merge-pdf
#https://www.ilovepdf.com/zh-cn/merge_pdf

#三步：创建画布，画图，关闭画布
jpeg(filename="output.jpg")  #输出文件名为output 的jpg文件
plot(1:10)  #绘图
dev.off()   #关闭jpeg绘图设备 输出的图形请在工作路径下查找
dev.new() #如果Rstudio左边框里看不到图，就输这个命令
#输出高分辨率的图，用于发表
tiff(filename = "test.tif", width = 1200, height = 1200,units = "px",
pointsize = 3,bg = "white", res = 600)
?tiff
```