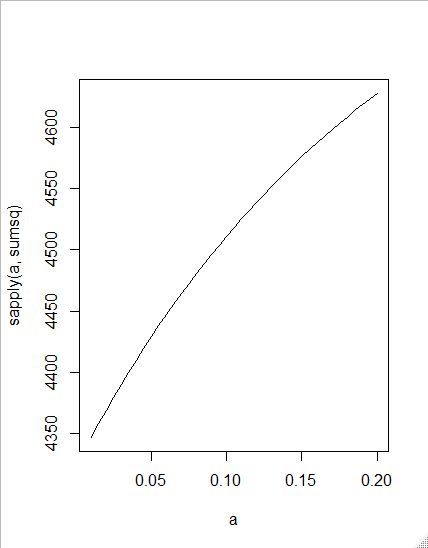
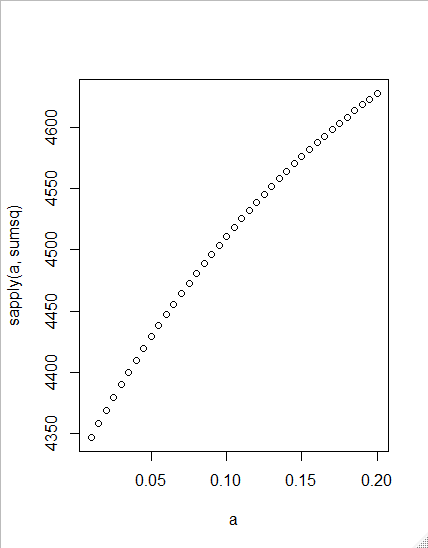
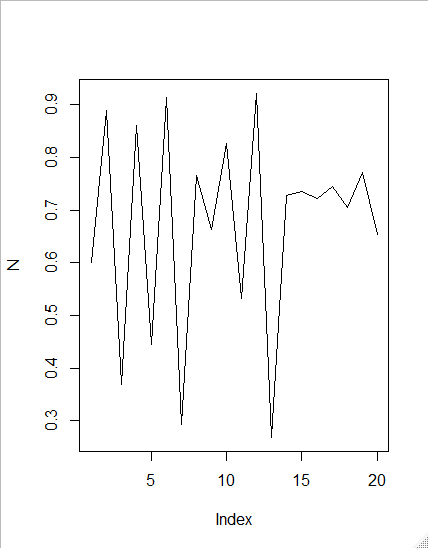
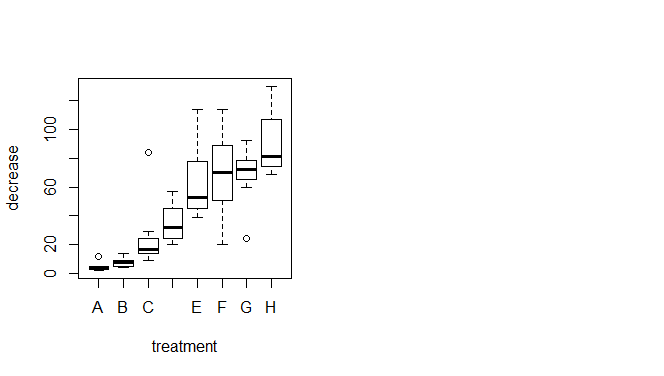
Nathan Wemmer









[1] 1 2 3 4 5 6 7

Code starts here:

> #sapdecay <- read.table("c:\\temp\\sapdecay.txt",header=T)

> sapdecay <- read.table("sapdecay.txt",header=T)

> attach(sapdecay)

The following objects are masked \_by\_ .GlobalEnv:

x, y

> names(sapdecay)

[1] "x" "y"

> sumsq <- function(a,xv=x,yv=y)

+ { yf <- exp(-a\*xv)

+ sum((yv-yf)^2) }

> # look at the function

> # below are the same because x, y are the default

> sumsq(1,x,y)

[1] 4861.861

Warning message:

In yv - yf :

longer object length is not a multiple of shorter object length

> sumsq(1)

[1] 4861.861

Warning message:

In yv - yf :

longer object length is not a multiple of shorter object length

> ########################################

> lm(log(y)~x)

Error in model.frame.default(formula = log(y) ~ x, drop.unused.levels = TRUE) :

variable lengths differ (found for 'x')

> a <- seq(0.01,0.2,.005)

> plot(a,sapply(a,sumsq),type="l")

There were 39 warnings (use warnings() to see them)

> plot(a,sapply(a,sumsq))

There were 39 warnings (use warnings() to see them)

> a[min(sapply(a,sumsq))==sapply(a,sumsq)]

[1] 0.01

There were 50 or more warnings (use warnings() to see the first 50)

> plot(x,y)

Error in xy.coords(x, y, xlabel, ylabel, log) :

'x' and 'y' lengths differ

> xv <- seq(0,50,0.1)

> lines(xv,exp(-0.055\*xv))

> fa <- function(a) sum((y-exp(-a\*x))^2)

> optimize(fa,c(0.01,0.1))

$minimum

[1] 0.01006598

$objective

[1] 4347.29

There were 16 warnings (use warnings() to see them)

> fb <- function(a) sum(abs(y-exp(-a\*x)))

> optimize(fb,c(0.01,0.1))

$minimum

[1] 0.01006598

$objective

[1] 276.9706

There were 16 warnings (use warnings() to see them)

> #data <- read.table("c:\\temp\\pgfull.txt",header=T)

> data <- read.table("pgfull.txt",header=F)

> attach(data)

> names(data)

[1] "V1" "V2" "V3" "V4" "V5" "V6" "V7" "V8"

[9] "V9" "V10" "V11" "V12" "V13" "V14" "V15" "V16"

[17] "V17" "V18" "V19" "V20" "V21" "V22" "V23" "V24"

[25] "V25" "V26" "V27" "V28" "V29" "V30" "V31" "V32"

[33] "V33" "V34" "V35" "V36" "V37" "V38" "V39" "V40"

[41] "V41" "V42" "V43" "V44" "V45" "V46" "V47" "V48"

[49] "V49" "V50" "V51" "V52" "V53" "V54" "V55" "V56"

[57] "V57" "V58" "V59"

> species <- data[,1:54]

> max.col(species)

[1] NA 22 22 22 1 32 32 22 1 22 22 22 1 22 22 1 1

[18] 22 22 22 4 2 2 51 2 1 1 22 22 1 1 2 5 1

[35] 4 2 2 1 4 22 22 22 4 2 2 25 25 2 2 5 25

[52] 32 1 22 22 2 2 1 1 51 2 2 27 2 2 2 2 35

[69] 51 51 1 2 2 1 1 32 32 1 1 1 1 1 1 14 1

[86] 2 1 1 2 2

Warning message:

In max.col(species) : NAs introduced by coercion

> names(species)[max.col(species)]

[1] NA "V22" "V22" "V22" "V1" "V32" "V32" "V22"

[9] "V1" "V22" "V22" "V22" "V1" "V22" "V22" "V1"

[17] "V1" "V22" "V22" "V22" "V4" "V2" "V2" "V51"

[25] "V2" "V1" "V1" "V22" "V22" "V1" "V1" "V2"

[33] "V5" "V1" "V4" "V2" "V2" "V1" "V4" "V22"

[41] "V22" "V22" "V4" "V2" "V2" "V25" "V25" "V2"

[49] "V2" "V5" "V25" "V32" "V1" "V22" "V22" "V2"

[57] "V2" "V1" "V1" "V51" "V2" "V2" "V27" "V2"

[65] "V2" "V2" "V2" "V35" "V51" "V51" "V1" "V2"

[73] "V2" "V1" "V1" "V32" "V32" "V1" "V1" "V1"

[81] "V1" "V1" "V1" "V14" "V1" "V2" "V1" "V1"

[89] "V2" "V2"

Warning message:

In max.col(species) : NAs introduced by coercion

> table(names(species)[max.col(species)])

V1 V14 V2 V22 V25 V27 V32 V35 V4 V5 V51

26 1 23 19 3 1 5 1 4 2 4

Warning message:

In max.col(species) : NAs introduced by coercion

> length(table(names(species)[max.col(species)]))

[1] 11

Warning message:

In max.col(species) : NAs introduced by coercion

> length(names(species))-length(table(names(species)[max.col(species)]))

[1] 43

Warning message:

In max.col(species) : NAs introduced by coercion

> max.col(-species)

[1] NA NA NA NA NA NA NA NA NA NA NA NA NA NA NA NA NA

[18] NA NA NA NA NA NA NA NA NA NA NA NA NA NA NA NA NA

[35] NA NA NA NA NA NA NA NA NA NA NA NA NA NA NA NA NA

[52] NA NA NA NA NA NA NA NA NA NA NA NA NA NA NA NA NA

[69] NA NA NA NA NA NA NA NA NA NA NA NA NA NA NA NA NA

[86] NA NA NA NA NA

There were 50 or more warnings (use warnings() to see the first 50)

> detach(data)

> #2.8.8 Restructuring a multi-dimensional array using aperm

> data <- array(1:24, 2:4)

> data

, , 1

[,1] [,2] [,3]

[1,] 1 3 5

[2,] 2 4 6

, , 2

[,1] [,2] [,3]

[1,] 7 9 11

[2,] 8 10 12

, , 3

[,1] [,2] [,3]

[1,] 13 15 17

[2,] 14 16 18

, , 4

[,1] [,2] [,3]

[1,] 19 21 23

[2,] 20 22 24

> dimnames(data)[[1]] <- list("male","female")

> dimnames(data)[[2]] <- list("young","mid","old")

> dimnames(data)[[3]] <- list("A","B","C","D")

> dimnames(data)

[[1]]

[1] "male" "female"

[[2]]

[1] "young" "mid" "old"

[[3]]

[1] "A" "B" "C" "D"

> new.data <- aperm(data,c(2,3,1))

> new.data

, , male

A B C D

young 1 7 13 19

mid 3 9 15 21

old 5 11 17 23

, , female

A B C D

young 2 8 14 20

mid 4 10 16 22

old 6 12 18 24

> #######################################################

> #2.9 Random numbers, sampling and shuffling

> #######################################################

> set.seed(375)

> runif(3)

[1] 0.9613669 0.6918535 0.7302684

> runif(3)

[1] 0.9228566 0.1603804 0.9642799

> runif(3)

[1] 0.52880907 0.08660864 0.29075809

> set.seed(375)

> runif(3)

[1] 0.9613669 0.6918535 0.7302684

> current<-.Random.seed

> runif(3)

[1] 0.9228566 0.1603804 0.9642799

> runif(3)

[1] 0.52880907 0.08660864 0.29075809

> runif(3)

[1] 0.02590182 0.85520652 0.31350305

> .Random.seed<-current

> runif(3)

[1] 0.9228566 0.1603804 0.9642799

> y <- c(8,3,5,7,6,6,8,9,2,3,9,4,10,4,11)

> sample(y)

[1] 8 4 3 10 9 7 6 3 8 6 5 4 9 2 11

> sample(y)

[1] 8 3 3 4 2 10 5 9 4 7 6 11 6 8 9

> sample(y,5)

[1] 3 2 9 10 6

> sample(y,5)

[1] 6 2 6 11 3

> sample(y,replace=T)

[1] 2 4 9 9 8 8 8 4 5 3 8 9 4 5 4

> sample(y,replace=T)

[1] 5 5 4 4 6 3 6 9 8 7 4 10 11 5 8

> p <- c(1, 2, 3, 4, 5, 5, 4, 3, 2, 1)

> x <- 1:10

> sapply(1:5,function(i) sample(x,4,prob=p))

[,1] [,2] [,3] [,4] [,5]

[1,] 9 5 5 7 4

[2,] 4 4 1 5 10

[3,] 6 8 6 2 8

[4,] 3 2 7 3 5

> #######################################################

> # 2.10 Loops and repeats

> #######################################################

> for (i in 1:5) print(i^2)

[1] 1

[1] 4

[1] 9

[1] 16

[1] 25

> j <- k <- 0

> for (i in 1:5) {

+ j <- j+1

+ k <- k+i\*j

+ print(i+j+k) }

[1] 3

[1] 9

[1] 20

[1] 38

[1] 65

> fac1 <- function(x) {

+ f <- 1

+ if (x<2) return (1)

+ for (i in 2:x) {

+ f <- f\*i}

+ f }

> sapply(0:5,fac1)

[1] 1 1 2 6 24 120

> fac2 <- function(x) {

+ f <- 1

+ t <- x

+ while(t>1) {

+ f <- f\*t

+ t <- t-1 }

+ return(f) }

> sapply(0:5,fac2)

[1] 1 1 2 6 24 120

> fac3 <- function(x) {

+ f <- 1

+ t <- x

+ repeat {

+ if (t<2) break

+ f <- f\*t

+ t <- t-1 }

+ return(f) }

> sapply(0:5,fac3)

[1] 1 1 2 6 24 120

> cumprod(1:5)

[1] 1 2 6 24 120

> fac4 <- function(x) max(cumprod(1:x))

> max(cumprod(1:0))

[1] 1

> sapply(0:5,fac4)

[1] 1 1 2 6 24 120

> fac5 <- function(x) gamma(x+1)

> sapply(0:5,fac5)

[1] 1 1 2 6 24 120

> sapply(0:5,factorial)

[1] 1 1 2 6 24 120

> binary <- function(x) {

+ i <- 0

+ string <- numeric(32)

+ while(x>0) {

+ string[32-i]< -x %% 2

+ x <- x%% 2

+ i <- i+1 }

+ first <- match(1,string)

+ string[first:32] }

> sapply(15:17,binary)

> fibonacci <- function(n) {

+ a <- 1

+ b <- 0

+ while(n>0)

+ {swap <- a

+ a <- a+b

+ b <- swap

+ n <- n-1 }

+ b }

> sapply(1:10,fibonacci)

[1] 1 1 2 3 5 8 13 21 34 55

> for (i in 1:length(y)) { if(y[i] < 0) y[i] <- 0 }

> y[y<0] <- 0

> z <- ifelse (y < 0, -1, 1)

> # data <- read.table("c:\\temp\\worms.txt",header=T)

> data <- read.table("worms.txt",header=T)

> attach(data)

> ifelse(Area>median(Area),"big","small")

[1] "big" "big" "small" "small" "big" "big"

[7] "big" "small" "small" "small" "small" "big"

[13] "big" "small" "big" "big" "small" "big"

[19] "small" "small"

> y <- log(rpois(20,1.5))

> y

[1] 0.0000000 0.0000000 0.0000000 -Inf 0.6931472

[6] 0.0000000 -Inf 0.6931472 0.6931472 0.6931472

[11] -Inf 0.6931472 0.0000000 0.0000000 1.3862944

[16] 0.0000000 1.0986123 0.0000000 0.6931472 0.6931472

> ifelse(y<0,NA,y)

[1] 0.0000000 0.0000000 0.0000000 NA 0.6931472

[6] 0.0000000 NA 0.6931472 0.6931472 0.6931472

[11] NA 0.6931472 0.0000000 0.0000000 1.3862944

[16] 0.0000000 1.0986123 0.0000000 0.6931472 0.6931472

> #################################################

> # 2.10.3 The slowness of loops

> #################################################

> x <- runif(10000000)

> system.time(max(x))

user system elapsed

0.02 0.00 0.02

> pc <- proc.time()

> cmax <- x[1]

> for (i in 2:10000000) {

+ if(x[i]>cmax) cmax <- x[i] }

> proc.time()-pc

user system elapsed

0.52 0.00 0.55

> # 2.10.4 Do not 'grow' data sets by concatenation or recursive function calls

> test1 <- function(){

+ y <- 1:100000

+ }

> test2 <- function(){

+ y <- numeric(100000)

+ for (i in 1:100000) y[i] <- i

+ }

> test3 <- function(){

+ y <- NULL

+ for (i in 1:100000) y <- c(y,i)

+ }

> proc.time()

user system elapsed

139.92 1.68 664.45

> system.time(test1())

user system elapsed

0 0 0

> system.time(test2())

user system elapsed

0 0 0

>

> binary <- function(x) {

+ i <- 0

+ string <- numeric(32)

+ while(x>0) {

+ string[32-i]< -x %% 2

+ x <- x%% 2

+ i <- i+1 }

+ first <- match(1,string)

+ string[first:32] }

> i <- 0

> string <- numeric(32)

> while(x>0) {

+ string[32-i]< -x %% 2

+ x <- x%% 2

+ i <- i+1 }

Lost warning messages

Warning messages:

1: In while (x > 0) { :

the condition has length > 1 and only the first element will be used

2: In while (x > 0) { :

the condition has length > 1 and only the first element will be used

3: In while (x > 0) { :

the condition has length > 1 and only the first element will be used

4: In while (x > 0) { :

the condition has length > 1 and only the first element will be used

5: In while (x > 0) { :

the condition has length > 1 and only the first element will be used

> first <- match(1,string)

> string[first:32] }

Error: unexpected '}' in " string[first:32] }"

> sapply(15:17,binary)

> sapply(15:17,binary)

> sapply(15:17,binary)

> for(i in 1:5) print(i^2)

[1] 1

[1] 4

[1] 9

[1] 16

[1] 25

> j<- k <- 0

> for (i in1:5) {}

Error: unexpected symbol in "for (i in1"

> for (i in 1:5) {}

> j <- k <- 0

> for (i in 1:5) {

+ j <- j+1

+ k <- k+i\*j

+ print(i+j+k) }

[1] 3

[1] 9

[1] 20

[1] 38

[1] 65

> fac1 <- function(x) {

+ f <- 1

+ if (x<2) return (1)

+ for (i in 2:x) {

+ f <- f\*i}

+ f }

> sapply(0:5,fac1)

[1] 1 1 2 6 24 120

> fac2 <- function(x) {

+ f <- 1

+ t <- x

+ while(t>1) {

+ f <- f\*t

+ t <- t-1 }

+ return(f) }

> sapply(0:5,fac2)

[1] 1 1 2 6 24 120

> fac3 <- function(x) {

+ f <- 1

+ t <- x

+ repeat {

+ if (t<2) break

+ f <- f\*t

+ t <- t-1 }

+ return(f) }

> sapply(0:5,fac3)

[1] 1 1 2 6 24 120

> cumprod(1:5)

[1] 1 2 6 24 120

> fac4 <- function(x) max(cumprod(1:x))

> max(cumprod(1:0))

[1] 1

> sapply(0:5,fac4)

[1] 1 1 2 6 24 120

> fac5 <- function(x) gamma(x+1)

> sapply(0:5,fac5)

[1] 1 1 2 6 24 120

> sapply(0:5,factorial)

[1] 1 1 2 6 24 120

> binary <- function(x) {

+ i <- 0

+ string <- numeric(32)

+ while(x>0) {

+ string[32-i]< -x %% 2

+ x <- x%% 2

+ i <- i+1 }

+ first <- match(1,string)

+ string[first:32] }

> sapply(15:17,binary)

>

>

> fibonacci <- function(n) {

+ a <- 1

+ b <- 0

+ while(n>0)

+ {swap <- a

+ a <- a+b

+ b <- swap

+ n <- n-1 }

+ b }

> sapply(1:10,fibonacci)

[1] 1 1 2 3 5 8 13 21 34 55

> for (i in 1:length(y)) { if(y[i] < 0) y[i] <- 0 }

> y[y<0] <- 0

> z <- ifelse (y < 0, -1, 1)

> # data <- read.table("c:\\temp\\worms.txt",header=T)

> data <- read.table("worms.txt",header=T)

> attach(data)

The following objects are masked from data (pos = 3):

Area, Damp, Field.Name, Slope, Soil.pH,

Vegetation, Worm.density

> ifelse(Area>median(Area),"big","small")

[1] "big" "big" "small" "small" "big" "big"

[7] "big" "small" "small" "small" "small" "big"

[13] "big" "small" "big" "big" "small" "big"

[19] "small" "small"

> y <- log(rpois(20,1.5))

> y

[1] 0.0000000 -Inf 0.0000000 0.0000000 0.0000000

[6] 0.6931472 0.0000000 0.6931472 -Inf 0.0000000

[11] 0.6931472 1.0986123 0.6931472 0.0000000 0.0000000

[16] -Inf 0.0000000 -Inf -Inf 0.0000000

> ifelse(y<0,NA,y)

[1] 0.0000000 NA 0.0000000 0.0000000 0.0000000

[6] 0.6931472 0.0000000 0.6931472 NA 0.0000000

[11] 0.6931472 1.0986123 0.6931472 0.0000000 0.0000000

[16] NA 0.0000000 NA NA 0.0000000

> #################################################

> # 2.10.3 The slowness of loops

> #################################################

> x <- runif(10000000)

> system.time(max(x))

user system elapsed

0.02 0.00 0.02

> pc <- proc.time()

> cmax <- x[1]

> for (i in 2:10000000) {

+ if(x[i]>cmax) cmax <- x[i] }

> proc.time()-pc

user system elapsed

0.51 0.00 2.45

> # 2.10.4 Do not 'grow' data sets by concatenation or recursive function calls

> test1 <- function(){

+ y <- 1:100000

+ }

> test2 <- function(){

+ y <- numeric(100000)

+ for (i in 1:100000) y[i] <- i

+ }

> test3 <- function(){

+ y <- NULL

+ for (i in 1:100000) y <- c(y,i)

+ }

> proc.time()

user system elapsed

418.04 35.51 1229.26

> system.time(test1())

user system elapsed

0 0 0

> system.time(test2())

user system elapsed

0.01 0.00 0.02

> system.time(test3())

user system elapsed

9.89 0.09 10.03

> # 2.10.5 Loops for producing time series

> next.year <- function(x) lambda \* x \* (1 - x)

> next.year(0.6)

Error in next.year(0.6) : object 'lambda' not found

> system.time(test3())

user system elapsed

9.80 0.00 9.83

> # 2.10.5 Loops for producing time series

> next.year <- function(x) lambda \* x \* (1 - x)

> lambda <- 3.7

> next.year(0.6)

[1] 0.888

> next.year(0.888)

[1] 0.3679872

> N <- numeric(20)

> N[1] <- 0.6

> for (t in 2:20) N[t] <- next.year(N[t-1])

> plot(N,type="l")

> apples <- c(4,4.5,4.2,5.1,3.9)

> oranges <- c(TRUE, TRUE, FALSE)

> chalk <- c("limestone", "marl","oolite", "CaC03")

> cheese <- c(3.2-4.5i,12.8+2.2i)

> data.frame(apples,oranges,chalk,cheese)

Error in data.frame(apples, oranges, chalk, cheese) :

arguments imply differing number of rows: 5, 3, 4, 2

> apples <- c(4,4.5,4.2,5.1,3.9)

> oranges <- c(TRUE, TRUE, FALSE)

> chalk <- c("limestone", "marl","oolite", "CaC03")

> cheese <- c(3.2-4.5i,12.8+2.2i)

> data.frame(apples,oranges,chalk,cheese)

Error in data.frame(apples, oranges, chalk, cheese) :

arguments imply differing number of rows: 5, 3, 4, 2

> # List can handle 4 different objects

> items <- list(apples,oranges,chalk,cheese)

> items

[[1]]

[1] 4.0 4.5 4.2 5.1 3.9

[[2]]

[1] TRUE TRUE FALSE

[[3]]

[1] "limestone" "marl" "oolite" "CaC03"

[[4]]

[1] 3.2-4.5i 12.8+2.2i

> items[[3]]

[1] "limestone" "marl" "oolite" "CaC03"

> items[[3]][3]

[1] "oolite"

> items[3]

[[1]]

[1] "limestone" "marl" "oolite" "CaC03"

> items[3][3]

[[1]]

NULL

> names(items)

NULL

> items <- list(first=apples,second=oranges,third=chalk,fourth=cheese)

> items$fourth

[1] 3.2-4.5i 12.8+2.2i

> # 2.11.1 Lists and lapply

> class(items)

[1] "list"

> mode(items)

[1] "list"

> is.numeric(items)

[1] FALSE

> is.list(items)

[1] TRUE

> length(items)

[1] 4

> lapply(items,length)

$first

[1] 5

$second

[1] 3

$third

[1] 4

$fourth

[1] 2

> lapply(items,class)

$first

[1] "numeric"

$second

[1] "logical"

$third

[1] "character"

$fourth

[1] "complex"

> lapply(items,mean)

$first

[1] 4.34

$second

[1] 0.6666667

$third

[1] NA

$fourth

[1] 8-1.15i

Warning message:

In mean.default(X[[i]], ...) :

argument is not numeric or logical: returning NA

> summary(items)

Length Class Mode

first 5 -none- numeric

second 3 -none- logical

third 4 -none- character

fourth 2 -none- complex

> str(items)

List of 4

$ first : num [1:5] 4 4.5 4.2 5.1 3.9

$ second: logi [1:3] TRUE TRUE FALSE

$ third : chr [1:4] "limestone" "marl" "oolite" "CaC03"

$ fourth: cplx [1:2] 3.2-4.5i 12.8+2.2i

> # 2.11.2 Manipulating and saving lists (skip it)

> # where is the data? I can not find it.

> data<-read.csv("c:\\temp\\pa.csv",row.names=1)

Error in file(file, "rt") : cannot open the connection

In addition: Warning message:

In file(file, "rt") :

cannot open file 'c:\temp\pa.csv': No such file or directory

> data

Field.Name Area Slope Vegetation Soil.pH Damp

1 Nashs.Field 3.6 11 Grassland 4.1 FALSE

2 Silwood.Bottom 5.1 2 Arable 5.2 FALSE

3 Nursery.Field 2.8 3 Grassland 4.3 FALSE

4 Rush.Meadow 2.4 5 Meadow 4.9 TRUE

5 Gunness.Thicket 3.8 0 Scrub 4.2 FALSE

6 Oak.Mead 3.1 2 Grassland 3.9 FALSE

7 Church.Field 3.5 3 Grassland 4.2 FALSE

8 Ashurst 2.1 0 Arable 4.8 FALSE

9 The.Orchard 1.9 0 Orchard 5.7 FALSE

10 Rookery.Slope 1.5 4 Grassland 5.0 TRUE

11 Garden.Wood 2.9 10 Scrub 5.2 FALSE

12 North.Gravel 3.3 1 Grassland 4.1 FALSE

13 South.Gravel 3.7 2 Grassland 4.0 FALSE

14 Observatory.Ridge 1.8 6 Grassland 3.8 FALSE

15 Pond.Field 4.1 0 Meadow 5.0 TRUE

16 Water.Meadow 3.9 0 Meadow 4.9 TRUE

17 Cheapside 2.2 8 Scrub 4.7 TRUE

18 Pound.Hill 4.4 2 Arable 4.5 FALSE

19 Gravel.Pit 2.9 1 Grassland 3.5 FALSE

20 Farm.Wood 0.8 10 Scrub 5.1 TRUE

Worm.density

1 4

2 7

3 2

4 5

5 6

6 2

7 3

8 4

9 9

10 7

11 8

12 1

13 2

14 0

15 6

16 8

17 4

18 5

19 1

20 3

> sapply(1:10,function(i) which(data[i,]>0))

[[1]]

[1] 2 3 5 7

[[2]]

[1] 2 3 5 7

[[3]]

[1] 2 3 5 7

[[4]]

[1] 2 3 5 6 7

[[5]]

[1] 2 5 7

[[6]]

[1] 2 3 5 7

[[7]]

[1] 2 3 5 7

[[8]]

[1] 2 5 7

[[9]]

[1] 2 5 7

[[10]]

[1] 2 3 5 6 7

There were 20 warnings (use warnings() to see them)

> spp<-sapply(1:10,function(i) which(data[i,]>0))

There were 20 warnings (use warnings() to see them)

> sapply(1:10, function(i)names(data)[spp[[i]]] )

[[1]]

[1] "Area" "Slope" "Soil.pH"

[4] "Worm.density"

[[2]]

[1] "Area" "Slope" "Soil.pH"

[4] "Worm.density"

[[3]]

[1] "Area" "Slope" "Soil.pH"

[4] "Worm.density"

[[4]]

[1] "Area" "Slope" "Soil.pH"

[4] "Damp" "Worm.density"

[[5]]

[1] "Area" "Soil.pH" "Worm.density"

[[6]]

[1] "Area" "Slope" "Soil.pH"

[4] "Worm.density"

[[7]]

[1] "Area" "Slope" "Soil.pH"

[4] "Worm.density"

[[8]]

[1] "Area" "Soil.pH" "Worm.density"

[[9]]

[1] "Area" "Soil.pH" "Worm.density"

[[10]]

[1] "Area" "Slope" "Soil.pH"

[4] "Damp" "Worm.density"

> sapply(1:9, function (j) rownames(data)[data[,j]>0] )

Error in `[.data.frame`(data, , j) : undefined columns selected

In addition: Warning messages:

1: In Ops.factor(data[, j], 0) : ‘>’ not meaningful for factors

2: In Ops.factor(data[, j], 0) :

C:\Users\Nathan\AppData\Local\Microsoft\Windows\INetCache\Content.MSO\5FEC01E7.tmp Show Traceback

C:\Users\Nathan\AppData\Local\Microsoft\Windows\INetCache\Content.MSO\4F1DAB8D.tmp Rerun with Debug

Error in `[.data.frame`(data, , j) : undefined columns selected

> spplists<-sapply(1:9, function (j) rownames(data)[data[,j]>0] )

Error in `[.data.frame`(data, , j) : undefined columns selected

In addition: Warning messages:

1: In Ops.factor(data[, j], 0) : ‘>’ not meaningful for factors

2: In Ops.factor(data[, j], 0) :

C:\Users\Nathan\AppData\Local\Microsoft\Windows\INetCache\Content.MSO\162E5223.tmp Show Traceback

C:\Users\Nathan\AppData\Local\Microsoft\Windows\INetCache\Content.MSO\66DA3329.tmp Rerun with Debug

Error in `[.data.frame`(data, , j) : undefined columns selected

> ######################################################

> # 2.12 Text, character strings and pattern matching

> ######################################################

> a <- "abc"

> b <- "123"

> as.numeric(a)

[1] NA

Warning message:

NAs introduced by coercion

> ######################################################

> # 2.12 Text, character strings and pattern matching

> ######################################################

> a <- "abc"

> b <- "123"

> as.numeric(a)

[1] NA

Warning message:

NAs introduced by coercion

> as.numeric(b)

[1] 123

> pets <- c("cat","dog","gerbil","terrapin")

> length(pets)

[1] 4

> nchar(pets)

[1] 3 3 6 8

> class(pets)

[1] "character"

> is.factor(pets)

[1] FALSE

> df <- data.frame(pets)

> is.factor(df$pets)

[1] TRUE

> letters

[1] "a" "b" "c" "d" "e" "f" "g" "h" "i" "j" "k" "l" "m"

[14] "n" "o" "p" "q" "r" "s" "t" "u" "v" "w" "x" "y" "z"

> LETTERS

[1] "A" "B" "C" "D" "E" "F" "G" "H" "I" "J" "K" "L" "M"

[14] "N" "O" "P" "Q" "R" "S" "T" "U" "V" "W" "X" "Y" "Z"

> which(letters=="n")

[1] 14

> noquote(letters)

[1] a b c d e f g h i j k l m n o p q r s t u v w x y z

> # 2.12.1 Pasting character strings together

> c(a,b)

[1] "abc" "123"

> paste(a,b,sep="")

[1] "abc123"

> paste(a,b)

[1] "abc 123"

> paste(a,b," a longer phrase containing blanks",sep="")

[1] "abc123 a longer phrase containing blanks"

> d <- c(a,b,"new")

> e <- paste(d,"a longer phrase containing blanks")

> e

[1] "abc a longer phrase containing blanks"

[2] "123 a longer phrase containing blanks"

[3] "new a longer phrase containing blanks"

> drive <- "c:"

> folder <- "temp"

> file <- "file"

> extension <- ".txt"

> paste(drive, folder, file, extension)

[1] "c: temp file .txt"

> paste(drive, "\\",folder, "\\",file, extension,sep="")

[1] "c:\\temp\\file.txt"

> # 2.12.2 Extracting parts of strings

> phrase <- "the quick brown fox jumps over the lazy dog"

> q <- character(20)

> for (i in 1:20) q[i] <- substr(phrase,1,i)

> q

[1] "t" "th"

[3] "the" "the "

[5] "the q" "the qu"

[7] "the qui" "the quic"

[9] "the quick" "the quick "

[11] "the quick b" "the quick br"

[13] "the quick bro" "the quick brow"

[15] "the quick brown" "the quick brown "

[17] "the quick brown f" "the quick brown fo"

[19] "the quick brown fox" "the quick brown fox "

> # 2.12.3 Counting things within strings

> nchar(phrase)

[1] 43

> strsplit(phrase,split=character(0))

[[1]]

[1] "t" "h" "e" " " "q" "u" "i" "c" "k" " " "b" "r" "o"

[14] "w" "n" " " "f" "o" "x" " " "j" "u" "m" "p" "s" " "

[27] "o" "v" "e" "r" " " "t" "h" "e" " " "l" "a" "z" "y"

[40] " " "d" "o" "g"

> table(strsplit(phrase,split=character(0)))

a b c d e f g h i j k l m n o p q r s t u v w x y z

8 1 1 1 1 3 1 1 2 1 1 1 1 1 1 4 1 1 2 1 2 2 1 1 1 1 1

> words <- 1+table(strsplit(phrase,split=character(0)))[1]

> words

9

> strsplit(phrase, " ")

[[1]]

[1] "the" "quick" "brown" "fox" "jumps" "over"

[7] "the" "lazy" "dog"

> table(lapply(strsplit(phrase, " "), nchar))

3 4 5

4 2 3

> strsplit(phrase,NULL)

[[1]]

[1] "t" "h" "e" " " "q" "u" "i" "c" "k" " " "b" "r" "o"

[14] "w" "n" " " "f" "o" "x" " " "j" "u" "m" "p" "s" " "

[27] "o" "v" "e" "r" " " "t" "h" "e" " " "l" "a" "z" "y"

[40] " " "d" "o" "g"

> lapply(strsplit(phrase,NULL),rev)

[[1]]

[1] "g" "o" "d" " " "y" "z" "a" "l" " " "e" "h" "t" " "

[14] "r" "e" "v" "o" " " "s" "p" "m" "u" "j" " " "x" "o"

[27] "f" " " "n" "w" "o" "r" "b" " " "k" "c" "i" "u" "q"

[40] " " "e" "h" "t"

> sapply(lapply(strsplit(phrase, NULL), rev), paste, collapse="")

[1] "god yzal eht revo spmuj xof nworb kciuq eht"

> strsplit(phrase,"the")

[[1]]

[1] ""

[2] " quick brown fox jumps over "

[3] " lazy dog"

> strsplit(phrase,"the")[[1]] [2]

[1] " quick brown fox jumps over "

> nchar(strsplit(phrase,"the")[[1]] [2])

[1] 28

> # 2.12.4 Upper- and lower-case text

> toupper(phrase)

[1] "THE QUICK BROWN FOX JUMPS OVER THE LAZY DOG"

> tolower(toupper(phrase))

[1] "the quick brown fox jumps over the lazy dog"

> first <- c(5,8,3,5,3,6,4,4,2,8,8,8,4,4,6)

> second <- c(8,6,4,2)

> match(first,second)

[1] NA 1 NA NA NA 2 3 3 4 1 1 1 3 3 2

> subjects <- c("A", "B", "G", "M", "N", "S", "T", "V", "Z")

> suitable.patients <- c("E", "G", "S", "U", "Z")

> match(subjects, suitable.patients)

[1] NA NA 2 NA NA 3 NA NA 5

> drug <- c("new", "conventional")

> drug[ifelse(is.na(match(subjects, suitable.patients)),2,1)]

[1] "conventional" "conventional" "new"

[4] "conventional" "conventional" "new"

[7] "conventional" "conventional" "new"

> #####################################################

> # 2.12.6 Pattern matching

> #####################################################

> #setwd("F:/JunYe\_Files/Jun\_2016Spring\_Course/Spring2016\_Advanced\_Computinbg/therbook")

> #wf <- read.table("worldfloras.txt",header=F)

> #wf <- read.table("c:\\temp\\worldfloras.txt",header=T)

> wf <- read.table("worldfloras.txt",header=F)

> attach(wf)

> names(wf)

[1] "V1" "V2" "V3" "V4" "V5" "V6" "V7"

> Country

Error: object 'Country' not found

> as.vector(Country[grep("R",as.character(Country))])

Error in as.vector(Country[grep("R", as.character(Country))]) :

object 'Country' not found

> as.vector(Country[grep("^R",as.character(Country))])

Error in as.vector(Country[grep("^R", as.character(Country))]) :

object 'Country' not found

> as.vector(Country[grep(" R",as.character(Country))])

Error in as.vector(Country[grep(" R", as.character(Country))]) :

object 'Country' not found

> #####################################################

> # 2.12.6 Pattern matching

> #####################################################

> #setwd("F:/JunYe\_Files/Jun\_2016Spring\_Course/Spring2016\_Advanced\_Computinbg/therbook")

> #wf <- read.table("worldfloras.txt",header=F)

> #wf <- read.table("c:\\temp\\worldfloras.txt",header=T)

> wf <- read.table("worldfloras.txt",header=F)

> attach(wf)

The following objects are masked from wf (pos = 3):

V1, V2, V3, V4, V5, V6, V7

> names(wf)

[1] "V1" "V2" "V3" "V4" "V5" "V6" "V7"

> Country

Error: object 'Country' not found

> as.vector(Country[grep("R",as.character(Country))])

Error in as.vector(Country[grep("R", as.character(Country))]) :

object 'Country' not found

V1, V2, V3, V4, V5, V6, V7

> #####################################################

> # 2.12.6 Pattern matching

> #####################################################

> #setwd("F:/JunYe\_Files/Jun\_2016Spring\_Course/Spring2016\_Advanced\_Computinbg/therbook")

> #wf <- read.table("worldfloras.txt",header=F)

> #wf <- read.table("c:\\temp\\worldfloras.txt",header=T)

> wf <- read.table("worldfloras.txt",header=T)

> attach(wf)

The following object is masked from data (pos = 6):

Area

The following object is masked from data (pos = 7):

Area

> names(wf)

[1] "Country" "Latitude" "Area" "Population"

[5] "Flora" "Endemism" "Continent"

> Country

[1] Afghanistan Albania

[3] Algeria Andorra

[5] Angola Antarctica

[7] Argentina Australia

[9] Austria Bahrain

[11] Balearic Islands Bangladesh

[13] Belgium Belize

[15] Benin Bhutan

[17] Bolivia Botswana

[19] Brazil Brunei

[21] Bulgaria Burkina Faso

[23] Burma Burundi

[25] Cameroon Canada

[27] Central African Republic Chad

[29] Chile China

[31] Colombia Congo

[33] Corsica Costa Rica

[35] Crete Cuba

[37] Cyprus Czechoslovakia

[39] Denmark Dominican Republic

[41] Ecuador Egypt

[43] El Salvador Ethiopia

[45] Finland France

[47] French Guiana Gabon

[49] Galapagos Gambia

[51] Germany East Germany West

[53] Ghana Greece

[55] Greenland Guatemala

[57] Guinea Guyana

[59] Haiti Hawaii

[61] Honduras Hong Kong

[63] Hungary Iceland

[65] India Indonesia

[67] Iran Iraq

[69] Ireland Israel

[71] Italy Ivory Coast

[73] Jamaica Japan

[75] Jordan Kampuchea

[77] Kenya Korea

[79] Kuwait Laos

[81] Lebanon Lesotho

[83] Liberia Libya

[85] Liechtenstein Luxembourg

[87] Madagascar Malawi

[89] Malaysia Mali

[91] Malta Mauritania

[93] Mauritius Mexico

[95] Mongolia Morocco

[97] Mozambique Namibia

[99] Nepal Netherlands

[101] New Caledonia New Zealand

[103] Nicaragua Niger

[105] Nigeria Norway

[107] Oman Pakistan

[109] Panama Papua New Guinea

[111] Paraguay Peru

[113] Philippines Poland

[115] Portugal Puerto Rico

[117] Qatar Reunion

[119] Romania Rwanda

[121] Sardinia Saudi Arabia

[123] Senegal Seychelles

[125] Sicily Sierra Leone

[127] Singapore Solomon Islands

[129] Somalia South Africa

[131] Spain Sri Lanka

[133] St.Helena Sudan

[135] Suriname Swaziland

[137] Sweden Switzerland

[139] Syria Taiwan

[141] Tanzania Thailand

[143] Togo Trinidad & Tobago

[145] Tristan da Cunha Tunisia

[147] Turkey Uganda

[149] United Kingdom Uruguay

[151] USA USSR

[153] Vanuatu Venezuela

[155] Viet Nam Yemen North

[157] Yemen South Yugoslavia

[159] Zaire Zambia

[161] Zimbabwe

161 Levels: Afghanistan Albania Algeria ... Zimbabwe

> as.vector(Country[grep("R",as.character(Country))])

[1] "Central African Republic" "Costa Rica"

[3] "Dominican Republic" "Puerto Rico"

[5] "Reunion" "Romania"

[7] "Rwanda" "USSR"

> as.vector(Country[grep("^R",as.character(Country))])

[1] "Reunion" "Romania" "Rwanda"

> as.vector(Country[grep(" R",as.character(Country))])

[1] "Central African Republic" "Costa Rica"

[3] "Dominican Republic" "Puerto Rico"

> as.vector(Country[grep(" ",as.character(Country))])

[1] "Balearic Islands"

[2] "Burkina Faso"

[3] "Central African Republic"

[4] "Costa Rica"

[5] "Dominican Republic"

[6] "El Salvador"

[7] "French Guiana"

[8] "Germany East"

[9] "Germany West"

[10] "Hong Kong"

[11] "Ivory Coast"

[12] "New Caledonia"

[13] "New Zealand"

[14] "Papua New Guinea"

[15] "Puerto Rico"

[16] "Saudi Arabia"

[17] "Sierra Leone"

[18] "Solomon Islands"

[19] "South Africa"

[20] "Sri Lanka"

[21] "Trinidad & Tobago"

[22] "Tristan da Cunha"

[23] "United Kingdom"

[24] "Viet Nam"

[25] "Yemen North"

[26] "Yemen South"

> as.vector(Country[grep("y$",as.character(Country))])

[1] "Hungary" "Italy" "Norway" "Paraguay"

[5] "Sicily" "Turkey" "Uruguay"

> as.vector(Country[grep("[C-E]",as.character(Country))])

[1] "Cameroon"

[2] "Canada"

[3] "Central African Republic"

[4] "Chad"

[5] "Chile"

[6] "China"

[7] "Colombia"

[8] "Congo"

[9] "Corsica"

[10] "Costa Rica"

[11] "Crete"

[12] "Cuba"

[13] "Cyprus"

[14] "Czechoslovakia"

[15] "Denmark"

[16] "Dominican Republic"

[17] "Ecuador"

[18] "Egypt"

[19] "El Salvador"

[20] "Ethiopia"

[21] "Germany East"

[22] "Ivory Coast"

[23] "New Caledonia"

[24] "Tristan da Cunha"

> as.vector(Country[grep("^[C-E]",as.character(Country))])

[1] "Cameroon"

[2] "Canada"

[3] "Central African Republic"

[4] "Chad"

[5] "Chile"

[6] "China"

[7] "Colombia"

[8] "Congo"

[9] "Corsica"

[10] "Costa Rica"

[11] "Crete"

[12] "Cuba"

[13] "Cyprus"

[14] "Czechoslovakia"

[15] "Denmark"

[16] "Dominican Republic"

[17] "Ecuador"

[18] "Egypt"

[19] "El Salvador"

[20] "Ethiopia"

> as.vector(Country[-grep("[a-t]$",as.character(Country))])

[1] "Hungary" "Italy" "Norway" "Paraguay"

[5] "Peru" "Sicily" "Turkey" "Uruguay"

[9] "USA" "USSR" "Vanuatu"

> as.vector(Country[-grep("[A-T a-t]$",as.character(Country))])

[1] "Hungary" "Italy" "Norway" "Paraguay"

[5] "Peru" "Sicily" "Turkey" "Uruguay"

[9] "Vanuatu"

> as.vector(Country[grep("^.y",as.character(Country))])

[1] "Cyprus" "Syria"

> as.vector(Country[grep("^..y",as.character(Country))])

[1] "Egypt" "Guyana" "Seychelles"

> as.vector(Country[grep("^.{5}y",as.character(Country))])

[1] "Norway" "Sicily" "Turkey"

> as.vector(Country[grep("^.{,4}$",as.character(Country))])

[1] "Benin" "Burma" "Chad" "Chile" "China" "Congo"

[7] "Crete" "Cuba" "Egypt" "Gabon" "Ghana" "Haiti"

[13] "India" "Iran" "Iraq" "Italy" "Japan" "Kenya"

[19] "Korea" "Laos" "Libya" "Mali" "Malta" "Nepal"

[25] "Niger" "Oman" "Peru" "Qatar" "Spain" "Sudan"

[31] "Syria" "Togo" "USA" "USSR" "Zaire"

> as.vector(Country[grep("^.{15,}$",as.character(Country))])

[1] "Balearic Islands" "Central African Republic"

[3] "Dominican Republic" "Papua New Guinea"

[5] "Solomon Islands" "Trinidad & Tobago"

[7] "Tristan da Cunha"

> text <- c("arm", "leg", "head", "foot", "hand", "hindleg", "elbow")

> # see the difference between sub and gsub

> gsub("h","H",text)

[1] "arm" "leg" "Head" "foot" "Hand"

[6] "Hindleg" "elbow"

> sub("o","O",text)

[1] "arm" "leg" "head" "fOot" "hand"

[6] "hindleg" "elbOw"

> gsub("o","O",text)

[1] "arm" "leg" "head" "fOOt" "hand"

[6] "hindleg" "elbOw"

> gsub("^.","O",text)

[1] "Orm" "Oeg" "Oead" "Ooot" "Oand"

[6] "Oindleg" "Olbow"

> gsub("(\\w\*)(\\w\*)", "\\U\\1\\L\\2",text, perl=TRUE)

[1] "ARM" "LEG" "HEAD" "FOOT" "HAND"

[6] "HINDLEG" "ELBOW"

> gsub("(\\w\*)", "\\U\\1",text, perl=TRUE)

[1] "ARM" "LEG" "HEAD" "FOOT" "HAND"

[6] "HINDLEG" "ELBOW"

> ############################################################

> # 2.12.9 Locations of a pattern within a vector using regexpr

> ############################################################

> text

[1] "arm" "leg" "head" "foot" "hand"

[6] "hindleg" "elbow"

> regexpr("o",text)

[1] -1 -1 -1 2 -1 -1 4

attr(,"match.length")

[1] -1 -1 -1 1 -1 -1 1

attr(,"index.type")

[1] "chars"

attr(,"useBytes")

[1] TRUE

> grep("o",text)

[1] 4 7

> text[grep("o",text)]

[1] "foot" "elbow"

> freq <- as.vector(unlist (lapply(gregexpr("o",text),length)))

> present <- ifelse(regexpr("o",text)<0,0,1)

> freq\*present

[1] 0 0 0 2 0 0 1

> charmatch("m", c("mean", "median", "mode"))

[1] 0

> charmatch("med", c("mean", "median", "mode"))

[1] 2

> #2.12.10 Using %in% and which

> stock <- c("car","van")

> requests <- c("truck","suv","van","sports","car","waggon","car")

> which(requests %in% stock)

[1] 3 5 7

> requests [which(requests %in% stock)]

[1] "van" "car" "car"

> stock[match(requests,stock)][!is.na(match(requests,stock))]

[1] "van" "car" "car"

> which(sapply(requests, "%in%", stock))

van car car

3 5 7

> # 2.12.11 More on pattern matching

> text <- c("arm","leg","head", "foot","hand", "hindleg", "elbow")

> grep("o{1}",text,value=T)

[1] "foot" "elbow"

> grep("o{2}",text,value=T)

[1] "foot"

> grep("o{3}",text,value=T)

character(0)

> grep("[[:alnum:]]{4, }",text,value=T)

[1] "head" "foot" "hand" "hindleg" "elbow"

> grep("[[:alnum:]]{5, }",text,value=T)

[1] "hindleg" "elbow"

> grep("[[:alnum:]]{6, }",text,value=T)

[1] "hindleg"

> grep("[[:alnum:]]{7, }",text,value=T)

[1] "hindleg"

> # 2.12.12 Perl regular expressions

> # 2.12.13 Stripping patterned text out of complex strings

> (entries <- c ("Trial 1 58 cervicornis (52 match)", "Trial 2 60

+ terrestris (51 matched)", "Trial 8 109 flavicollis (101 matches)"))

[1] "Trial 1 58 cervicornis (52 match)"

[2] "Trial 2 60\nterrestris (51 matched)"

[3] "Trial 8 109 flavicollis (101 matches)"

> gsub(" \*$", "", gsub("\\(.\*\\)$", "", entries))

[1] "Trial 1 58 cervicornis" "Trial 2 60\nterrestris"

[3] "Trial 8 109 flavicollis"

> pos <- regexpr("\\(.\*\\)$", entries)

> substring(entries, first=pos+1, last=pos+attr(pos,"match.length")-2)

[1] "52 match" "51 matched" "101 matches"

> pos

[1] 24 23 25

attr(,"match.length")

[1] 10 12 13

attr(,"index.type")

[1] "chars"

attr(,"useBytes")

[1] TRUE

> #############################################################

> # 2.13 Dates and times in R

> #############################################################

> Sys.time()

[1] "2019-09-04 17:46:44 EDT"

> as.numeric(Sys.time())

[1] 1567633605

> class(Sys.time())

[1] "POSIXct" "POSIXt"

> time.list <- as.POSIXlt(Sys.time())

> unlist(time.list)

sec min hour

"45.5264499187469" "46" "17"

mday mon year

"4" "8" "119"

wday yday isdst

"3" "246" "1"

zone gmtoff

"EDT" "-14400"

> # 2.13.1 Reading time data from files

> # data <- read.table("c:\\temp\\dates.txt",header=T)

> data <- read.table("dates.txt",header=F)

Error in scan(file = file, what = what, sep = sep, quote = quote, dec = dec, :

line 1 did not have 4 elements

> # 2.13.1 Reading time data from files

> # data <- read.table("c:\\temp\\dates.txt",header=T)

> data <- read.table("dates.txt",header=T)

Error in read.table("dates.txt", header = T) :

more columns than column names

> #############################################################

> # 2.13 Dates and times in R

> #############################################################

> Sys.time()

[1] "2019-09-04 17:47:13 EDT"

> as.numeric(Sys.time())

[1] 1567633634

> class(Sys.time())

[1] "POSIXct" "POSIXt"

> time.list <- as.POSIXlt(Sys.time())

> # 2.13.1 Reading time data from files

> # data <- read.table("c:\\temp\\dates.txt",header=T)

> data <- read.table("dates.txt",header=T)

Error in read.table("dates.txt", header = T) :

more columns than column names

> # 2.13.1 Reading time data from files

> # data <- read.table("c:\\temp\\dates.txt",header=T)

> data <- read.table("dates.txt",header=F)

Error in scan(file = file, what = what, sep = sep, quote = quote, dec = dec, :

line 1 did not have 4 elements

> # 2.13.1 Reading time data from files

> # data <- read.table("c:\\temp\\dates.txt",header=T)

> data <- read.table("dates.txt",header=T)

Error in read.table("dates.txt", header = T) :

more columns than column names

> # 2.13.1 Reading time data from files

> # data <- read.table("c:\\temp\\dates.txt",header=T)

> data <- read.table("dates.txt",header=F)

Error in scan(file = file, what = what, sep = sep, quote = quote, dec = dec, :

line 1 did not have 4 elements

> unlist(time.list)

sec min hour

"19.2497448921204" "47" "17"

mday mon year

"4" "8" "119"

wday yday isdst

"3" "246" "1"

zone gmtoff

"EDT" "-14400"

> # 2.13.1 Reading time data from files

> # data <- read.table("c:\\temp\\dates.txt",header=T)

> data <- read.table("dates.txt",header=F)

Error in scan(file = file, what = what, sep = sep, quote = quote, dec = dec, :

line 1 did not have 4 elements

> # 2.13.1 Reading time data from files

> # data <- read.table("c:\\temp\\dates.txt",header=T)

> data <- read.table("dates.txt",header=T)

Error in read.table("dates.txt", header = T) :

more columns than column names

> attach(data)

The following object is masked from wf (pos = 3):

Area

The following objects are masked from data (pos = 7):

Area, Damp, Field.Name, Slope, Soil.pH,

Vegetation, Worm.density

The following objects are masked from data (pos = 8):

Area, Damp, Field.Name, Slope, Soil.pH,

Vegetation, Worm.density

> head(data)

Field.Name Area Slope Vegetation Soil.pH Damp

1 Nashs.Field 3.6 11 Grassland 4.1 FALSE

2 Silwood.Bottom 5.1 2 Arable 5.2 FALSE

3 Nursery.Field 2.8 3 Grassland 4.3 FALSE

4 Rush.Meadow 2.4 5 Meadow 4.9 TRUE

5 Gunness.Thicket 3.8 0 Scrub 4.2 FALSE

6 Oak.Mead 3.1 2 Grassland 3.9 FALSE

Worm.density

1 4

2 7

3 2

4 5

5 6

6 2

> mode(date)

[1] "function"

> class(date)

[1] "function"

> # 2.13.2 The strptime function

> Rdate <- strptime(as.character(date),"%d/%m/%Y")

Error in as.character(date) :

cannot coerce type 'closure' to vector of type 'character'

> # 2.13.2 The strptime function

> Rdate <- strptime(as.character(date),"%d/%m/%Y")

Error in as.character(date) :

cannot coerce type 'closure' to vector of type 'character'

> # 2.13.1 Reading time data from files

> # data <- read.table("c:\\temp\\dates.txt",header=T)

> data <- read.table("dates.txt",header=T)

Error in read.table("dates.txt", header = T) :

more columns than column names

> attach(data)

The following objects are masked from data (pos = 3):

Area, Damp, Field.Name, Slope, Soil.pH,

Vegetation, Worm.density

The following object is masked from wf (pos = 4):

Area

The following objects are masked from data (pos = 8):

Area, Damp, Field.Name, Slope, Soil.pH,

Vegetation, Worm.density

The following objects are masked from data (pos = 9):

Area, Damp, Field.Name, Slope, Soil.pH,

Vegetation, Worm.density

> # 2.13.1 Reading time data from files

> # data <- read.table("c:\\temp\\dates.txt",header=T)

> data <- read.table("dates.txt",header=T)

Error in read.table("dates.txt", header = T) :

more columns than column names

> # 2.13.1 Reading time data from files

> # data <- read.table("c:\\temp\\dates.txt",header=T)

> data <- read.table("dates.txt",header=T)

Error in read.table("dates.txt", header = T) :

more columns than column names

> # 2.13.1 Reading time data from files

> data <- read.table("c:\\temp\\dates.txt",header=T)

Error in file(file, "rt") : cannot open the connection

In addition: Warning message:

In file(file, "rt") :

cannot open file 'c:\temp\dates.txt': No such file or directory

> # 2.13.1 Reading time data from files

> # data <- read.table("c:\\temp\\dates.txt",header=T)

> data <- read.table("dates.txt",header=T)

Error in read.table("dates.txt", header = T) :

more columns than column names

> # 2.13.1 Reading time data from files

> # data <- read.table("c:\\temp\\dates.txt",header=T)

> data <- read.table("dates.txt",header=T,quote = "\"'")

Error in read.table("dates.txt", header = T, quote = "\"'") :

more columns than column names

> # 2.13.1 Reading time data from files

> # data <- read.table("c:\\temp\\dates.txt",header=T)

> data <- read.table("dates.txt",header=F,quote = "\"'")

Error in scan(file = file, what = what, sep = sep, quote = quote, dec = dec, :

line 1 did not have 4 elements

> # 2.13.1 Reading time data from files

> # data <- read.table("c:\\temp\\dates.txt",header=T)

> data <- read.table("dates.txt",header=T)

Error in read.table("dates.txt", header = T) :

more columns than column names

> # 2.13.1 Reading time data from files

> # data <- read.table("c:\\temp\\dates.txt",header=T)

> data <- read.table("dates.txt",header=F)

Error in scan(file = file, what = what, sep = sep, quote = quote, dec = dec, :

line 1 did not have 4 elements

> # 2.13.1 Reading time data from files

> # data <- read.table("c:\\temp\\dates.txt",header=T)

> data <- read.table("dates.txt",header=T)

Error in read.table("dates.txt", header = T) :

more columns than column names

> # 2.13.1 Reading time data from files

> # data <- read.table("c:\\temp\\dates.txt",header=T)

> data2 <- read.table("dates.txt",header=T)

Error in read.table("dates.txt", header = T) :

more columns than column names

> #############################################################

> # 2.13 Dates and times in R

> #############################################################

> Sys.time()

[1] "2019-09-04 18:06:53 EDT"

> as.numeric(Sys.time())

[1] 1567634814

> class(Sys.time())

[1] "POSIXct" "POSIXt"

> time.list <- as.POSIXlt(Sys.time())

> unlist(time.list)

sec min hour

"55.5796000957489" "6" "18"

mday mon year

"4" "8" "119"

wday yday isdst

"3" "246" "1"

zone gmtoff

"EDT" "-14400"

> # 2.13.1 Reading time data from files

> # data <- read.table("c:\\temp\\dates.txt",header=T)

> data2 <- read.table("dates.txt",header=T)

Error in read.table("dates.txt", header = T) :

more columns than column names

> # 2.13.1 Reading time data from files

> # data <- read.table("c:\\temp\\dates.txt",header=T)

> data2 <- read.table("dates.txt",header=F)

Error in scan(file = file, what = what, sep = sep, quote = quote, dec = dec, :

line 1 did not have 4 elements

> # 2.13.1 Reading time data from files

> # data <- read.table("c:\\temp\\dates.txt",header=T)

> data2 <- read.table("dates.txt",header=T)

Error in read.table("dates.txt", header = T) :

more columns than column names

> attach(data2)

Error in attach(data2) : object 'data2' not found

> head(data2)

Error in head(data2) : object 'data2' not found

> # 2.13.1 Reading time data from files

> # data <- read.table("c:\\temp\\dates.txt",header=T)

> data2 <- read.table("dates.txt",header=T)

Error in read.table("dates.txt", header = T) :

more columns than column names

> # 2.13.1 Reading time data from files

> # data <- read.table("c:\\temp\\dates.txt",header=T)

> data2 <- read.table("dates.txt",header=T)

Error in read.table("dates.txt", header = T) :

more columns than column names

> # 2.13.1 Reading time data from files

> data2 <- read.table("dates.txt",header=T)

Error in read.table("dates.txt", header = T) :

more columns than column names

> data

Field.Name Area Slope Vegetation Soil.pH Damp

1 Nashs.Field 3.6 11 Grassland 4.1 FALSE

2 Silwood.Bottom 5.1 2 Arable 5.2 FALSE

3 Nursery.Field 2.8 3 Grassland 4.3 FALSE

4 Rush.Meadow 2.4 5 Meadow 4.9 TRUE

5 Gunness.Thicket 3.8 0 Scrub 4.2 FALSE

6 Oak.Mead 3.1 2 Grassland 3.9 FALSE

7 Church.Field 3.5 3 Grassland 4.2 FALSE

8 Ashurst 2.1 0 Arable 4.8 FALSE

9 The.Orchard 1.9 0 Orchard 5.7 FALSE

10 Rookery.Slope 1.5 4 Grassland 5.0 TRUE

11 Garden.Wood 2.9 10 Scrub 5.2 FALSE

12 North.Gravel 3.3 1 Grassland 4.1 FALSE

13 South.Gravel 3.7 2 Grassland 4.0 FALSE

14 Observatory.Ridge 1.8 6 Grassland 3.8 FALSE

15 Pond.Field 4.1 0 Meadow 5.0 TRUE

16 Water.Meadow 3.9 0 Meadow 4.9 TRUE

17 Cheapside 2.2 8 Scrub 4.7 TRUE

18 Pound.Hill 4.4 2 Arable 4.5 FALSE

19 Gravel.Pit 2.9 1 Grassland 3.5 FALSE

20 Farm.Wood 0.8 10 Scrub 5.1 TRUE

Worm.density

1 4

2 7

3 2

4 5

5 6

6 2

7 3

8 4

9 9

10 7

11 8

12 1

13 2

14 0

15 6

16 8

17 4

18 5

19 1

20 3

> # 2.13.1 Reading time data from files

> # data <- read.table("c:\\temp\\dates.txt",header=T)

> data2 <- read.table("dates.txt",header=T)

Error in read.table("dates.txt", header = T) :

more columns than column names

> # 2.13.1 Reading time data from files

> # data <- read.table("c:\\temp\\dates.txt",header=T)

> data2 <- read.table("dates.txt",header=T)

Error in read.table("dates.txt", header = T) :

more columns than column names

> # 2.13.1 Reading time data from files

> # data <- read.table("c:\\temp\\dates.txt",header=T)

> data2 <- read.table("dates.txt",header=T)

Error in read.table("dates.txt", header = T) :

more columns than column names

> # 2.13.1 Reading time data from files

> # data <- read.table("c:\\temp\\dates.txt",header=T)

> data2 <- read.table("dates.txt",header=T)

Error in read.table("dates.txt", header = T) :

more columns than column names

> # 2.13.1 Reading time data from files

> # data <- read.table("c:\\temp\\dates.txt",header=T)

> data2 <- read.table("dates.txt",header=T)

Error in read.table("dates.txt", header = T) :

more columns than column names

> # 2.13.1 Reading time data from files

> # data <- read.table("c:\\temp\\dates.txt",header=T)

> data2 <- read.table("dates.txt",header=F)

Error in scan(file = file, what = what, sep = sep, quote = quote, dec = dec, :

line 1 did not have 4 elements

> attach(data2)

Error in attach(data2) : object 'data2' not found

> head(data2)

Error in head(data2) : object 'data2' not found

> mode(date)

[1] "function"

> class(date)

[1] "function"

> # 2.13.2 The strptime function

> Rdate <- strptime(as.character(date),"%d/%m/%Y")

Error in as.character(date) :

cannot coerce type 'closure' to vector of type 'character'

> class(Rdate)

Error: object 'Rdate' not found

> data <- data.frame(data,Rdate)

Error in data.frame(data, Rdate) : object 'Rdate' not found

> head(data)

Field.Name Area Slope Vegetation Soil.pH Damp

1 Nashs.Field 3.6 11 Grassland 4.1 FALSE

2 Silwood.Bottom 5.1 2 Arable 5.2 FALSE

3 Nursery.Field 2.8 3 Grassland 4.3 FALSE

4 Rush.Meadow 2.4 5 Meadow 4.9 TRUE

5 Gunness.Thicket 3.8 0 Scrub 4.2 FALSE

6 Oak.Mead 3.1 2 Grassland 3.9 FALSE

Worm.density

1 4

2 7

3 2

4 5

5 6

6 2

> tapply(x,Rdate$wday,mean)

Error in tapply(x, Rdate$wday, mean) : object 'Rdate' not found

> y <- strptime("01/02/2014",format="%d/%m/%Y")

> weekdays(y)

[1] "Saturday"

> y$wday

[1] 6

> other.dates <- c("1jan99", "2jan05", "31mar04", "30jul05")

> strptime(other.dates, "%d%b%y")

[1] "1999-01-01 EST" "2005-01-02 EST" "2004-03-31 EST"

[4] "2005-07-30 EDT"

> yet.another.date <- c("2016 January 2 Mon","2017 February 6 Fri","2018

+ March 10 Tue")

> strptime(yet.another.date,"%Y %B %W %a")

[1] "2016-01-11 EST" "2017-02-10 EST" "2018-03-06 EST"

> yet.more.dates <- c("2016 2 Mon","2017 6 Fri","2018 10 Tue")

> strptime(yet.more.dates,"%Y %W %a")

[1] "2016-01-11 EST" "2017-02-10 EST" "2018-03-06 EST"

> # 2.13.3 The difftime function

> as.difftime(yet.more.dates,"%Y %W %a")

Time differences in days

[1] -1331.9583 -935.9583 -546.9583

> difftime("2014-02-06","2014-07-06")

Time difference of -149.9583 days

> round(difftime("2014-02-06","2014-07-06"),0)

Time difference of -150 days

> # 2.13.4 Calculations with dates and times

> y2 <- as.POSIXlt("2015-10-22")

> y1 <- as.POSIXlt("2018-10-22")

> y1-y2

Time difference of 1096 days

> #2.13.5 The difftime and as.difftime functions

> difftime("2015-10-21","2013-8-15")

Time difference of 797 days

> as.numeric(difftime("2015-10-21","2013-8-15"))

[1] 797

> t1 <- as.difftime("6:14:21")

> t2 <- as.difftime("5:12:32")

> t1-t2

Time difference of 1.030278 hours

> #times <- read.table("c:\\temp\\times.txt",header=T)

> times <- read.table("times.txt",header=F)

> attach(times)

The following objects are masked from wf (pos = 6):

V1, V2, V3, V4

The following objects are masked from wf (pos = 7):

V1, V2, V3, V4

The following objects are masked from wf (pos = 8):

V1, V2, V3, V4

> head(times)

V1 V2 V3 V4

1 hrs min sec experiment

2 2 23 6 A

3 3 16 17 A

4 3 2 56 A

5 2 45 0 A

6 3 4 42 A

> paste(hrs,min,sec,sep=":")

Error in paste(hrs, min, sec, sep = ":") : object 'hrs' not found

> duration <- as.difftime (paste(hrs,min,sec,sep=":"))

Error in paste(hrs, min, sec, sep = ":") : object 'hrs' not found

> tapply(duration,experiment,mean)

Error in tapply(duration, experiment, mean) :

object 'experiment' not found

> # 2.13.6 Generating sequences of dates

> seq(as.POSIXlt("2015-11-04"), as.POSIXlt("2015-11-15"), "1 day")

[1] "2015-11-04 EST" "2015-11-05 EST" "2015-11-06 EST"

[4] "2015-11-07 EST" "2015-11-08 EST" "2015-11-09 EST"

[7] "2015-11-10 EST" "2015-11-11 EST" "2015-11-12 EST"

[10] "2015-11-13 EST" "2015-11-14 EST" "2015-11-15 EST"

> seq(as.POSIXlt("2015-11-04"), as.POSIXlt("2016-04-05"), "2 weeks")

[1] "2015-11-04 00:00:00 EST" "2015-11-18 00:00:00 EST"

[3] "2015-12-02 00:00:00 EST" "2015-12-16 00:00:00 EST"

[5] "2015-12-30 00:00:00 EST" "2016-01-13 00:00:00 EST"

[7] "2016-01-27 00:00:00 EST" "2016-02-10 00:00:00 EST"

[9] "2016-02-24 00:00:00 EST" "2016-03-09 00:00:00 EST"

[11] "2016-03-23 01:00:00 EDT"

> seq(as.POSIXlt("2015-11-04"), as.POSIXlt("2018-10-04"), "3 months")

[1] "2015-11-04 EST" "2016-02-04 EST" "2016-05-04 EDT"

[4] "2016-08-04 EDT" "2016-11-04 EDT" "2017-02-04 EST"

[7] "2017-05-04 EDT" "2017-08-04 EDT" "2017-11-04 EDT"

[10] "2018-02-04 EST" "2018-05-04 EDT" "2018-08-04 EDT"

> seq(as.POSIXlt("2015-11-04"), as.POSIXlt("2026-02-04"), "year")

[1] "2015-11-04 EST" "2016-11-04 EDT" "2017-11-04 EDT"

[4] "2018-11-04 EDT" "2019-11-04 EST" "2020-11-04 EST"

[7] "2021-11-04 EDT" "2022-11-04 EDT" "2023-11-04 EDT"

[10] "2024-11-04 EST" "2025-11-04 EST"

> seq(as.POSIXlt("2015-11-04"), as.POSIXlt("2015-11-05"), 8955)

[1] "2015-11-04 00:00:00 EST" "2015-11-04 02:29:15 EST"

[3] "2015-11-04 04:58:30 EST" "2015-11-04 07:27:45 EST"

[5] "2015-11-04 09:57:00 EST" "2015-11-04 12:26:15 EST"

[7] "2015-11-04 14:55:30 EST" "2015-11-04 17:24:45 EST"

[9] "2015-11-04 19:54:00 EST" "2015-11-04 22:23:15 EST"

> seq(as.POSIXlt("2015-11-04"), by="month", length=10)

[1] "2015-11-04 EST" "2015-12-04 EST" "2016-01-04 EST"

[4] "2016-02-04 EST" "2016-03-04 EST" "2016-04-04 EDT"

[7] "2016-05-04 EDT" "2016-06-04 EDT" "2016-07-04 EDT"

[10] "2016-08-04 EDT"

> results <- runif(16)

> seq(as.POSIXlt("2015-11-04"), by="month", along=results )

[1] "2015-11-04 EST" "2015-12-04 EST" "2016-01-04 EST"

[4] "2016-02-04 EST" "2016-03-04 EST" "2016-04-04 EDT"

[7] "2016-05-04 EDT" "2016-06-04 EDT" "2016-07-04 EDT"

[10] "2016-08-04 EDT" "2016-09-04 EDT" "2016-10-04 EDT"

[13] "2016-11-04 EDT" "2016-12-04 EST" "2017-01-04 EST"

[16] "2017-02-04 EST"

> weekdays(seq(as.POSIXlt("2015-11-04"), by="month", along=results ))

[1] "Wednesday" "Friday" "Monday" "Thursday"

[5] "Friday" "Monday" "Wednesday" "Saturday"

[9] "Monday" "Thursday" "Sunday" "Tuesday"

[13] "Friday" "Sunday" "Wednesday" "Saturday"

> y <- as.Date(1:100,origin="2015-12-31")

> x <- as.POSIXlt(y)

> x[x$wday==1]

[1] "2016-01-04 UTC" "2016-01-11 UTC" "2016-01-18 UTC"

[4] "2016-01-25 UTC" "2016-02-01 UTC" "2016-02-08 UTC"

[7] "2016-02-15 UTC" "2016-02-22 UTC" "2016-02-29 UTC"

[10] "2016-03-07 UTC" "2016-03-14 UTC" "2016-03-21 UTC"

[13] "2016-03-28 UTC" "2016-04-04 UTC"

> y <- as.POSIXlt(as.Date(1:365,origin="2015-12-31"))

> data.frame(monday=y[y$wday==1],month=y$mo[y$wday==1])[1:12,]

monday month

1 2016-01-04 0

2 2016-01-11 0

3 2016-01-18 0

4 2016-01-25 0

5 2016-02-01 1

6 2016-02-08 1

7 2016-02-15 1

8 2016-02-22 1

9 2016-02-29 1

10 2016-03-07 2

11 2016-03-14 2

12 2016-03-21 2

> wanted <- !duplicated(y$mo[y$wday==1])

> y[y$wday==1][wanted]

[1] "2016-01-04 UTC" "2016-02-01 UTC" "2016-03-07 UTC"

[4] "2016-04-04 UTC" "2016-05-02 UTC" "2016-06-06 UTC"

[7] "2016-07-04 UTC" "2016-08-01 UTC" "2016-09-05 UTC"

[10] "2016-10-03 UTC" "2016-11-07 UTC" "2016-12-05 UTC"

> # 2.13.7 Calculating time differences between the rows of a dataframe

> class(duration)

Error: object 'duration' not found

> duration

Error: object 'duration' not found

> duration[1:15]-duration[2:16]

Error: object 'duration' not found

> length(duration[1:15]-duration[2:16])

Error: object 'duration' not found

> length(duration)

Error: object 'duration' not found

> diffs <- c(duration[1:15]-duration[2:16],NA)

Error: object 'duration' not found

> diffs

Error: object 'diffs' not found

> times$diffs <- diffs

Error: object 'diffs' not found

> times

V1 V2 V3 V4

1 hrs min sec experiment

2 2 23 6 A

3 3 16 17 A

4 3 2 56 A

5 2 45 0 A

6 3 4 42 A

7 2 56 25 A

8 3 12 28 A

9 1 57 12 A

10 2 22 22 B

11 1 42 7 B

12 2 31 17 B

13 3 15 16 B

14 2 28 4 B

15 1 55 34 B

16 2 17 7 B

17 1 48 48 B

> times$diffs[8] <- NA

Error in `$<-.data.frame`(`\*tmp\*`, diffs, value = c(NA, NA, NA, NA, NA, :

replacement has 8 rows, data has 17

> # 2.13.8 Regression using dates and times

> # data <- read.table("c:\\temp\\timereg.txt",header=T)

> data <- read.table("timereg.txt",header=F)

Error in file(file, "rt") : cannot open the connection

In addition: Warning message:

In file(file, "rt") :

cannot open file 'timereg.txt': No such file or directory

> attach(data)

The following objects are masked from data (pos = 4):

Area, Damp, Field.Name, Slope, Soil.pH,

Vegetation, Worm.density

The following objects are masked from data (pos = 5):

Area, Damp, Field.Name, Slope, Soil.pH,

Vegetation, Worm.density

The following object is masked from wf (pos = 6):

Area

The following objects are masked from data (pos = 10):

Area, Damp, Field.Name, Slope, Soil.pH,

Vegetation, Worm.density

The following objects are masked from data (pos = 11):

Area, Damp, Field.Name, Slope, Soil.pH,

Vegetation, Worm.density

> head(data)

Field.Name Area Slope Vegetation Soil.pH Damp

1 Nashs.Field 3.6 11 Grassland 4.1 FALSE

2 Silwood.Bottom 5.1 2 Arable 5.2 FALSE

3 Nursery.Field 2.8 3 Grassland 4.3 FALSE

4 Rush.Meadow 2.4 5 Meadow 4.9 TRUE

5 Gunness.Thicket 3.8 0 Scrub 4.2 FALSE

6 Oak.Mead 3.1 2 Grassland 3.9 FALSE

Worm.density

1 4

2 7

3 2

4 5

5 6

6 2

> dl <- strptime(date,"%d/%m/%Y")

Error in as.character(x) :

cannot coerce type 'closure' to vector of type 'character'

> class(dl)

Error: object 'dl' not found

> mode(dl)

Error in mode(dl) : object 'dl' not found

> windows(7,4)

> par(mfrow=c(1,2))

> # 2.13.8 Regression using dates and times

> # data <- read.table("c:\\temp\\timereg.txt",header=T)

> data <- read.table("timereg.txt",header=F)

Error in file(file, "rt") : cannot open the connection

In addition: Warning message:

In file(file, "rt") :

cannot open file 'timereg.txt': No such file or directory

> attach(data)

The following objects are masked from data (pos = 3):

Area, Damp, Field.Name, Slope, Soil.pH,

Vegetation, Worm.density

The following objects are masked from data (pos = 5):

Area, Damp, Field.Name, Slope, Soil.pH,

Vegetation, Worm.density

The following objects are masked from data (pos = 6):

Area, Damp, Field.Name, Slope, Soil.pH,

Vegetation, Worm.density

The following object is masked from wf (pos = 7):

Area

The following objects are masked from data (pos = 11):

Area, Damp, Field.Name, Slope, Soil.pH,

Vegetation, Worm.density

The following objects are masked from data (pos = 12):

Area, Damp, Field.Name, Slope, Soil.pH,

Vegetation, Worm.density

> head(data)

Field.Name Area Slope Vegetation Soil.pH Damp

1 Nashs.Field 3.6 11 Grassland 4.1 FALSE

2 Silwood.Bottom 5.1 2 Arable 5.2 FALSE

3 Nursery.Field 2.8 3 Grassland 4.3 FALSE

4 Rush.Meadow 2.4 5 Meadow 4.9 TRUE

5 Gunness.Thicket 3.8 0 Scrub 4.2 FALSE

6 Oak.Mead 3.1 2 Grassland 3.9 FALSE

Worm.density

1 4

2 7

3 2

4 5

5 6

6 2

> dl <- strptime(date,"%d/%m/%Y")

Error in as.character(x) :

cannot coerce type 'closure' to vector of type 'character'

> class(dl)

Error: object 'dl' not found

> mode(dl)

Error in mode(dl) : object 'dl' not found

> windows(7,4)

> par(mfrow=c(1,2))

> plot(dl,survivors,pch=16,xlab ="month")

Error in plot(dl, survivors, pch = 16, xlab = "month") :

object 'dl' not found

> plot(dl,log(survivors),pch=16,xlab ="month")

Error in plot(dl, log(survivors), pch = 16, xlab = "month") :

object 'dl' not found

> model <- lm(log(survivors)~dl,subset=(survivors>0))

Error in eval(predvars, data, env) : object 'survivors' not found

> dc <- as.POSIXct(dl)

Error in as.POSIXct(dl) : object 'dl' not found

> model <- lm(log(survivors)~dc,subset=(survivors>0))

Error in eval(predvars, data, env) : object 'survivors' not found

> abline(model)

Error in abline(model) : object 'model' not found

> summary(model)

Error in summary(model) : object 'model' not found

> -2.315E-07 \* 60 \* 60 \* 24 \* 30

[1] -0.600048

> 100\*exp(-0.600048 \* 2)

[1] 30.11653

> with(OrchardSprays,boxplot(decrease~treatment))

> library(MASS)

> with(bacteria,tapply((y=="n"),trt,sum))

placebo drug drug+

12 18 13

> with(mammals,plot(body,brain,log="xy"))

> # reg.data <- read.table("c:\\temp\\regression.txt",header=T)

> reg.data <- read.table("regression.txt",header=F)

> with (reg.data, {

+ model <- lm(growth~tannin)

+ summary(model) })

C:\Users\Nathan\AppData\Local\Microsoft\Windows\INetCache\Content.MSO\CA498EBE.tmp Show Traceback

C:\Users\Nathan\AppData\Local\Microsoft\Windows\INetCache\Content.MSO\4584E97C.tmp Rerun with Debug

Error in eval(predvars, data, env) : object 'growth' not found

> summary(lm(growth~tannin,data=reg.data))

Error in eval(predvars, data, env) : object 'growth' not found

> data()

Data sets in package ‘akima’:

akima Waveform Distortion Data for

Bivariate Interpolation

akima760 Sample data from Akima's Bicubic

Spline Interpolation code (TOMS

760)

franke Test datasets from Franke for

interpolation of scattered data

Data sets in package ‘boot’:

acme Monthly Excess Returns

aids Delay in AIDS Reporting in

England and Wales

aircondit Failures of Air-conditioning

Equipment

aircondit7 Failures of Air-conditioning

Equipment

amis Car Speeding and Warning Signs

aml Remission Times for Acute

Myelogenous Leukaemia

beaver Beaver Body Temperature Data

bigcity Population of U.S. Cities

brambles Spatial Location of Bramble Canes

breslow Smoking Deaths Among Doctors

calcium Calcium Uptake Data

cane Sugar-cane Disease Data

capability Simulated Manufacturing Process

Data

catsM Weight Data for Domestic Cats

cav Position of Muscle Caveolae

cd4 CD4 Counts for HIV-Positive

Patients

cd4.nested Nested Bootstrap of cd4 data

channing Channing House Data

city Population of U.S. Cities

claridge Genetic Links to Left-handedness

cloth Number of Flaws in Cloth

co.transfer Carbon Monoxide Transfer

coal Dates of Coal Mining Disasters

darwin Darwin's Plant Height Differences

dogs Cardiac Data for Domestic Dogs

downs.bc Incidence of Down's Syndrome in

British Columbia

ducks Behavioral and Plumage

Characteristics of Hybrid Ducks

fir Counts of Balsam-fir Seedlings

frets Head Dimensions in Brothers

grav Acceleration Due to Gravity

gravity Acceleration Due to Gravity

hirose Failure Time of PET Film

islay Jura Quartzite Azimuths on Islay

manaus Average Heights of the Rio Negro

river at Manaus

melanoma Survival from Malignant Melanoma

motor Data from a Simulated Motorcycle

Accident

neuro Neurophysiological Point Process

Data

nitrofen Toxicity of Nitrofen in Aquatic

Systems

nodal Nodal Involvement in Prostate

Cancer

nuclear Nuclear Power Station

Construction Data

paulsen Neurotransmission in Guinea Pig

Brains

poisons Animal Survival Times

polar Pole Positions of New Caledonian

Laterites

remission Cancer Remission and Cell

Activity

salinity Water Salinity and River

Discharge

survival Survival of Rats after Radiation

Doses

tau Tau Particle Decay Modes

tuna Tuna Sighting Data

urine Urine Analysis Data

wool Australian Relative Wool Prices

Data sets in package ‘cluster’:

agriculture European Union Agricultural

Workforces

animals Attributes of Animals

chorSub Subset of C-horizon of Kola Data

flower Flower Characteristics

plantTraits Plant Species Traits Data

pluton Isotopic Composition Plutonium

Batches

ruspini Ruspini Data

votes.repub Votes for Republican Candidate in

Presidential Elections

xclara Bivariate Data Set with 3

Clusters

Data sets in package ‘coda’:

line Simple linear regression example

Data sets in package ‘datasets’:

AirPassengers Monthly Airline Passenger Numbers

1949-1960

BJsales Sales Data with Leading Indicator

BJsales.lead (BJsales)

Sales Data with Leading Indicator

BOD Biochemical Oxygen Demand

CO2 Carbon Dioxide Uptake in Grass

Plants

ChickWeight Weight versus age of chicks on

different diets

DNase Elisa assay of DNase

EuStockMarkets Daily Closing Prices of Major

European Stock Indices, 1991-1998

Formaldehyde Determination of Formaldehyde

HairEyeColor Hair and Eye Color of Statistics

Students

Harman23.cor Harman Example 2.3

Harman74.cor Harman Example 7.4

Indometh Pharmacokinetics of Indomethacin

InsectSprays Effectiveness of Insect Sprays

JohnsonJohnson Quarterly Earnings per Johnson &

Johnson Share

LakeHuron Level of Lake Huron 1875-1972

LifeCycleSavings

Intercountry Life-Cycle Savings

Data

Loblolly Growth of Loblolly pine trees

Nile Flow of the River Nile

Orange Growth of Orange Trees

OrchardSprays Potency of Orchard Sprays

PlantGrowth Results from an Experiment on

Plant Growth

Puromycin Reaction Velocity of an Enzymatic

Reaction

Seatbelts Road Casualties in Great Britain

1969-84

Theoph Pharmacokinetics of Theophylline

Titanic Survival of passengers on the

Titanic

ToothGrowth The Effect of Vitamin C on Tooth

Growth in Guinea Pigs

UCBAdmissions Student Admissions at UC Berkeley

UKDriverDeaths Road Casualties in Great Britain

1969-84

UKgas UK Quarterly Gas Consumption

USAccDeaths Accidental Deaths in the US

1973-1978

USArrests Violent Crime Rates by US State

USJudgeRatings Lawyers' Ratings of State Judges

in the US Superior Court

USPersonalExpenditure

Personal Expenditure Data

UScitiesD Distances Between European Cities

and Between US Cities

VADeaths Death Rates in Virginia (1940)

WWWusage Internet Usage per Minute

WorldPhones The World's Telephones

ability.cov Ability and Intelligence Tests

airmiles Passenger Miles on Commercial US

Airlines, 1937-1960

airquality New York Air Quality Measurements

anscombe Anscombe's Quartet of 'Identical'

Simple Linear Regressions

attenu The Joyner-Boore Attenuation Data

attitude The Chatterjee-Price Attitude

Data

austres Quarterly Time Series of the

Number of Australian Residents

beaver1 (beavers)

Body Temperature Series of Two

Beavers

beaver2 (beavers)

Body Temperature Series of Two

Beavers

cars Speed and Stopping Distances of

Cars

chickwts Chicken Weights by Feed Type

co2 Mauna Loa Atmospheric CO2

Concentration

crimtab Student's 3000 Criminals Data

discoveries Yearly Numbers of Important

Discoveries

esoph Smoking, Alcohol and

(O)esophageal Cancer

euro Conversion Rates of Euro

Currencies

euro.cross (euro)

Conversion Rates of Euro

Currencies

eurodist Distances Between European Cities

and Between US Cities

faithful Old Faithful Geyser Data

fdeaths (UKLungDeaths)

Monthly Deaths from Lung Diseases

in the UK

freeny Freeny's Revenue Data

freeny.x (freeny)

Freeny's Revenue Data

freeny.y (freeny)

Freeny's Revenue Data

infert Infertility after Spontaneous and

Induced Abortion

iris Edgar Anderson's Iris Data

iris3 Edgar Anderson's Iris Data

islands Areas of the World's Major

Landmasses

ldeaths (UKLungDeaths)

Monthly Deaths from Lung Diseases

in the UK

lh Luteinizing Hormone in Blood

Samples

longley Longley's Economic Regression

Data

lynx Annual Canadian Lynx trappings

1821-1934

mdeaths (UKLungDeaths)

Monthly Deaths from Lung Diseases

in the UK

morley Michelson Speed of Light Data

mtcars Motor Trend Car Road Tests

nhtemp Average Yearly Temperatures in

New Haven

nottem Average Monthly Temperatures at

Nottingham, 1920-1939

npk Classical N, P, K Factorial

Experiment

occupationalStatus

Occupational Status of Fathers

and their Sons

precip Annual Precipitation in US Cities

presidents Quarterly Approval Ratings of US

Presidents

pressure Vapor Pressure of Mercury as a

Function of Temperature

quakes Locations of Earthquakes off Fiji

randu Random Numbers from Congruential

Generator RANDU

rivers Lengths of Major North American

Rivers

rock Measurements on Petroleum Rock

Samples

sleep Student's Sleep Data

stack.loss (stackloss)

Brownlee's Stack Loss Plant Data

stack.x (stackloss)

Brownlee's Stack Loss Plant Data

stackloss Brownlee's Stack Loss Plant Data

state.abb (state)

US State Facts and Figures

state.area (state)

US State Facts and Figures

state.center (state)

US State Facts and Figures

state.division (state)

US State Facts and Figures

state.name (state)

US State Facts and Figures

state.region (state)

US State Facts and Figures

state.x77 (state)

US State Facts and Figures

sunspot.month Monthly Sunspot Data, from 1749

to "Present"

sunspot.year Yearly Sunspot Data, 1700-1988

sunspots Monthly Sunspot Numbers,

1749-1983

swiss Swiss Fertility and Socioeconomic

Indicators (1888) Data

treering Yearly Treering Data, -6000-1979

trees Diameter, Height and Volume for

Black Cherry Trees

uspop Populations Recorded by the US

Census

volcano Topographic Information on

Auckland's Maunga Whau Volcano

warpbreaks The Number of Breaks in Yarn

during Weaving

women Average Heights and Weights for

American Women

Data sets in package ‘deldir’:

niProperties Northern Ireland properties.

seaweed seaweed

Data sets in package ‘deSolve’:

ccl4data Closed Chamber Study of CCl4

Metabolism by Rats.

Data sets in package ‘expm’:

matStig Stig's "infamous" Example Matrix

Data sets in package ‘gdata’:

MedUnits Table of conversions between

Intertional Standard (SI) and US

'Conventional' Units for common

medical measurements.

Data sets in package ‘gtools’:

ELISA Data from an ELISA assay

badDend Dataset That Crashes

Base:::Plot.Dendogram with 'Node

Stack Overflow'

Data sets in package ‘lattice’:

USMortality Mortality Rates in US by Cause

and Gender

USRegionalMortality

Mortality Rates in US by Cause

and Gender

barley Yield data from a Minnesota

barley trial

environmental Atmospheric environmental

conditions in New York City

ethanol Engine exhaust fumes from burning

ethanol

melanoma Melanoma skin cancer incidence

singer Heights of New York Choral

Society singers

Data sets in package ‘LearnBayes’:

achievement School achievement data

baseball.1964 Team records in the 1964 National

League baseball season

bermuda.grass Bermuda grass experiment data

birdextinct Bird measurements from British

islands

birthweight Birthweight regression study

breastcancer Survival experience of women with

breast cancer under treatment

calculus.grades

Calculus grades dataset

cancermortality

Cancer mortality data

chemotherapy Chemotherapy treatment effects on

ovarian cancer

darwin Darwin's data on plants

donner Donner survival study

election Florida election data

election.2008 Poll data from 2008 U.S.

Presidential Election

footballscores Game outcomes and point spreads

for American football

hearttransplants

Heart transplant mortality data

iowagpa Admissions data for an university

jeter2004 Hitting data for Derek Jeter

marathontimes Marathon running times

puffin Bird measurements from British

islands

schmidt Batting data for Mike Schmidt

sluggerdata Hitting statistics for ten great

baseball players

soccergoals Goals scored by professional

soccer team

stanfordheart Data from Stanford Heart

Transplanation Program

strikeout Baseball strikeout data

studentdata Student dataset

Data sets in package ‘lme4’:

Arabidopsis Arabidopsis

clipping/fertilization data

Dyestuff Yield of dyestuff by batch

Dyestuff2 Yield of dyestuff by batch

InstEval University Lecture/Instructor

Evaluations by Students at ETH

Pastes Paste strength by batch and cask

Penicillin Variation in penicillin testing

VerbAgg Verbal Aggression item responses

cake Breakage Angle of Chocolate Cakes

cbpp Contagious bovine pleuropneumonia

grouseticks Data on red grouse ticks from

Elston et al. 2001

grouseticks\_agg (grouseticks)

Data on red grouse ticks from

Elston et al. 2001

sleepstudy Reaction times in a sleep

deprivation study

Data sets in package ‘MASS’:

Aids2 Australian AIDS Survival Data

Animals Brain and Body Weights for 28

Species

Boston Housing Values in Suburbs of

Boston

Cars93 Data from 93 Cars on Sale in the

USA in 1993

Cushings Diagnostic Tests on Patients with

Cushing's Syndrome

DDT DDT in Kale

GAGurine Level of GAG in Urine of Children

Insurance Numbers of Car Insurance claims

Melanoma Survival from Malignant Melanoma

OME Tests of Auditory Perception in

Children with OME

Pima.te Diabetes in Pima Indian Women

Pima.tr Diabetes in Pima Indian Women

Pima.tr2 Diabetes in Pima Indian Women

Rabbit Blood Pressure in Rabbits

Rubber Accelerated Testing of Tyre

Rubber

SP500 Returns of the Standard and Poors

500

Sitka Growth Curves for Sitka Spruce

Trees in 1988

Sitka89 Growth Curves for Sitka Spruce

Trees in 1989

Skye AFM Compositions of Aphyric Skye

Lavas

Traffic Effect of Swedish Speed Limits on

Accidents

UScereal Nutritional and Marketing

Information on US Cereals

UScrime The Effect of Punishment Regimes

on Crime Rates

VA Veteran's Administration Lung

Cancer Trial

abbey Determinations of Nickel Content

accdeaths Accidental Deaths in the US

1973-1978

anorexia Anorexia Data on Weight Change

bacteria Presence of Bacteria after Drug

Treatments

beav1 Body Temperature Series of Beaver

1

beav2 Body Temperature Series of Beaver

2

biopsy Biopsy Data on Breast Cancer

Patients

birthwt Risk Factors Associated with Low

Infant Birth Weight

cabbages Data from a cabbage field trial

caith Colours of Eyes and Hair of

People in Caithness

cats Anatomical Data from Domestic

Cats

cement Heat Evolved by Setting Cements

chem Copper in Wholemeal Flour

coop Co-operative Trial in Analytical

Chemistry

cpus Performance of Computer CPUs

crabs Morphological Measurements on

Leptograpsus Crabs

deaths Monthly Deaths from Lung Diseases

in the UK

drivers Deaths of Car Drivers in Great

Britain 1969-84

eagles Foraging Ecology of Bald Eagles

epil Seizure Counts for Epileptics

farms Ecological Factors in Farm

Management

fgl Measurements of Forensic Glass

Fragments

forbes Forbes' Data on Boiling Points in

the Alps

galaxies Velocities for 82 Galaxies

gehan Remission Times of Leukaemia

Patients

genotype Rat Genotype Data

geyser Old Faithful Geyser Data

gilgais Line Transect of Soil in Gilgai

Territory

hills Record Times in Scottish Hill

Races

housing Frequency Table from a Copenhagen

Housing Conditions Survey

immer Yields from a Barley Field Trial

leuk Survival Times and White Blood

Counts for Leukaemia Patients

mammals Brain and Body Weights for 62

Species of Land Mammals

mcycle Data from a Simulated Motorcycle

Accident

menarche Age of Menarche in Warsaw

michelson Michelson's Speed of Light Data

minn38 Minnesota High School Graduates

of 1938

motors Accelerated Life Testing of

Motorettes

muscle Effect of Calcium Chloride on

Muscle Contraction in Rat Hearts

newcomb Newcomb's Measurements of the

Passage Time of Light

nlschools Eighth-Grade Pupils in the

Netherlands

npk Classical N, P, K Factorial

Experiment

npr1 US Naval Petroleum Reserve No. 1

data

oats Data from an Oats Field Trial

painters The Painter's Data of de Piles

petrol N. L. Prater's Petrol Refinery

Data

phones Belgium Phone Calls 1950-1973

quine Absenteeism from School in Rural

New South Wales

road Road Accident Deaths in US States

rotifer Numbers of Rotifers by Fluid

Density

ships Ships Damage Data

shoes Shoe wear data of Box, Hunter and

Hunter

shrimp Percentage of Shrimp in Shrimp

Cocktail

shuttle Space Shuttle Autolander Problem

snails Snail Mortality Data

steam The Saturated Steam Pressure Data

stormer The Stormer Viscometer Data

survey Student Survey Data

synth.te Synthetic Classification Problem

synth.tr Synthetic Classification Problem

topo Spatial Topographic Data

waders Counts of Waders at 15 Sites in

South Africa

whiteside House Insulation: Whiteside's

Data

wtloss Weight Loss Data from an Obese

Patient

Data sets in package ‘Matrix’:

CAex Albers' example Matrix with

"Difficult" Eigen Factorization

KNex Koenker-Ng Example Sparse Model

Matrix and Response Vector

USCounties USCounties Contiguity Matrix

Data sets in package ‘meta’:

Fleiss93 Aspirin after Myocardial

Infarction

Fleiss93cont Mental Health Treatment

Olkin95 Thrombolytic Therapy after Acute

Myocardial Infarction

amlodipine Amlodipine for Work Capacity

cisapride Cisapride in Non-Ulcer Dispepsia

lungcancer Smoking example

smoking Smoking example

woodyplants Elevated CO\_2 and total biomass

of woody plants

Data sets in package ‘metafor’:

dat.bangertdrowns2004

Studies on the Effectiveness of

Writing-to-Learn Interventions

dat.bcg Studies on the Effectiveness of

the BCG Vaccine Against

Tuberculosis

dat.begg1989 Studies on Bone-Marrow

Transplantation versus

Chemotherapy for the Treatment of

Leukemia

dat.berkey1998 Studies on Treatments for

Periodontal Disease

dat.bonett2010 Studies on the Reliability of the

CES-D Scale

dat.bourassa1996

Studies on the Association

between Handedness and

Eye-Dominance

dat.colditz1994

Studies on the Effectiveness of

the BCG Vaccine Against

Tuberculosis

dat.collins1985a

Studies on the Treatment of Upper

Gastrointestinal Bleeding by a

Histamine H2 Antagonist

dat.collins1985b

Studies on the Effects of

Diuretics in Pregnancy

dat.curtis1998 Studies on the Effects of

Elevated CO2 Levels on Woody

Plant Mass

dat.debruin2009

Studies on Standard Care Quality

and HAART-Adherence

dat.egger2001 Studies on the Effectiveness of

Intravenous Magnesium in Acute

Myocardial Infarction

dat.fine1993 Studies on Radiation Therapy with

or without Adjuvant Chemotherapy

in Patients with Malignant

Gliomas

dat.gibson2002 Studies on the Effectiveness of

Self-Management Education and

Regular Medical Review for Adults

with Asthma

dat.hackshaw1998

Studies on Lung Cancer Risk from

ETS Exposure

dat.hart1999 Studies on the Effectiveness of

Warfarin for Preventing Strokes

dat.hasselblad1998

Studies on the Effectiveness of

Counseling for Smoking Cessation

dat.hine1989 Studies on Prophylactic Use of

Lidocaine After a Heart Attack

dat.ishak2007 Studies on Deep-Brain Stimulation

dat.konstantopoulos2011

Studies on the Effects of

Modified School Calendars on

Student Achievement

dat.laopaiboon2015

Studies on the Effectiveness of

Azithromycin for Treating Lower

Respiratory Tract Infections

dat.lee2004 Studies on Acupoint P6

Stimulation for Preventing Nausea

dat.li2007 Studies on the Effectiveness of

Intravenous Magnesium in Acute

Myocardial Infarction

dat.linde2005 Studies on the Effectiveness of

St. John's Wort for Treating

Depression

dat.mcdaniel1994

Studies on the Validity of

Employment Interviews

dat.molloy2014 Studies on the Relationship

between Conscientiousness and

Medication Adherence

dat.nielweise2007

Studies on Anti-Infective-Treated

Central Venous Catheters for

Prevention of Catheter-Related

Bloodstream Infections

dat.nielweise2008

Studies on Anti-Infective-Treated

Central Venous Catheters for

Prevention of Catheter-Related

Bloodstream Infections

dat.normand1999

Studies on the Length of Hospital

Stay of Stroke Patients

dat.pagliaro1992

Studies on the Effectiveness of

Nonsurgical Treatments in

Cirrhosis

dat.pignon2000 Studies on the Effectiveness of

Locoregional Treatment plus

Chemotherapy for Head and Neck

Squamous-Cell Carcinoma

dat.pritz1997 Studies on the Effectiveness of

Hyperdynamic Therapy for Treating

Cerebral Vasospasm

dat.raudenbush1985

Studies on Assessing the Effects

of Teacher Expectations on Pupil

IQ

dat.riley2003 Studies on MYC-N as a Prognostic

Marker for Neuroblastoma

dat.senn2013 Studies on the Effectiveness of

Glucose-Lowering Agents

dat.yusuf1985 Studies of Beta Blockers During

and After Myocardial Infarction

Data sets in package ‘mgcv’:

columb Reduced version of Columbus OH

crime data

columb.polys Reduced version of Columbus OH

crime data

Data sets in package ‘nlme’:

Alfalfa Split-Plot Experiment on

Varieties of Alfalfa

Assay Bioassay on Cell Culture Plate

BodyWeight Rat weight over time for

different diets

Cefamandole Pharmacokinetics of Cefamandole

Dialyzer High-Flux Hemodialyzer

Earthquake Earthquake Intensity

Fatigue Cracks caused by metal fatigue

Gasoline Refinery yield of gasoline

Glucose Glucose levels over time

Glucose2 Glucose Levels Following Alcohol

Ingestion

Gun Methods for firing naval guns

IGF Radioimmunoassay of IGF-I Protein

Machines Productivity Scores for Machines

and Workers

MathAchSchool School demographic data for

MathAchieve

MathAchieve Mathematics achievement scores

Meat Tenderness of meat

Milk Protein content of cows' milk

Muscle Contraction of heart muscle

sections

Nitrendipene Assay of nitrendipene

Oats Split-plot Experiment on

Varieties of Oats

Orthodont Growth curve data on an

orthdontic measurement

Ovary Counts of Ovarian Follicles

Oxboys Heights of Boys in Oxford

Oxide Variability in Semiconductor

Manufacturing

PBG Effect of Phenylbiguanide on

Blood Pressure

Phenobarb Phenobarbitol Kinetics

Pixel X-ray pixel intensities over time

Quinidine Quinidine Kinetics

Rail Evaluation of Stress in Railway

Rails

RatPupWeight The weight of rat pups

Relaxin Assay for Relaxin

Remifentanil Pharmacokinetics of remifentanil

Soybean Growth of soybean plants

Spruce Growth of Spruce Trees

Tetracycline1 Pharmacokinetics of tetracycline

Tetracycline2 Pharmacokinetics of tetracycline

Wafer Modeling of Analog MOS Circuits

Wheat Yields by growing conditions

Wheat2 Wheat Yield Trials

bdf Language scores

ergoStool Ergometrics experiment with stool

types

Data sets in package ‘R2WinBUGS’:

schools 8 schools analysis

Data sets in package ‘rjags’:

LINE Linear regression example

Data sets in package ‘rpart’:

car.test.frame Automobile Data from 'Consumer

Reports' 1990

car90 Automobile Data from 'Consumer

Reports' 1990

cu.summary Automobile Data from 'Consumer

Reports' 1990

kyphosis Data on Children who have had

Corrective Spinal Surgery

solder Soldering of Components on

Printed-Circuit Boards

solder.balance (solder)

Soldering of Components on

Printed-Circuit Boards

stagec Stage C Prostate Cancer

Data sets in package ‘sp’:

Rlogo Rlogo jpeg image

gt (Rlogo) Rlogo jpeg image

meuse Meuse river data set

meuse.area River Meuse outline

meuse.grid Prediction Grid for Meuse Data

Set

meuse.grid\_ll Prediction Grid for Meuse Data

Set, geographical coordinates

meuse.riv River Meuse outline

Data sets in package ‘spatstat.data’:

Kovesi Colour Sequences with Uniform

Perceptual Contrast

amacrine Hughes' Amacrine Cell Data

anemones Beadlet Anemones Data

ants Harkness-Isham ants' nests data

ants.extra (ants)

Harkness-Isham ants' nests data

austates Australian States and Mainland

Territories

bdspots Breakdown Spots in

Microelectronic Materials

bei Tropical rain forest trees

bei.extra (bei)

Tropical rain forest trees

betacells Beta Ganglion Cells in Cat Retina

bramblecanes Hutchings' Bramble Canes data

bronzefilter Bronze gradient filter data

cells Biological Cells Point Pattern

cetaceans Point patterns of whale and

dolphin sightings.

cetaceans.extra (cetaceans)

Point patterns of whale and

dolphin sightings.

chicago Chicago Crime Data

chorley Chorley-Ribble Cancer Data

chorley.extra (chorley)

Chorley-Ribble Cancer Data

clmfires Castilla-La Mancha Forest Fires

clmfires.extra (clmfires)

Castilla-La Mancha Forest Fires

copper Berman-Huntington points and

lines data

demohyper Demonstration Example of

Hyperframe of Spatial Data

demopat Artificial Data Point Pattern

dendrite Dendritic Spines Data

finpines Pine saplings in Finland.

flu Influenza Virus Proteins

ganglia Beta Ganglion Cells in Cat

Retina, Old Version

gordon People in Gordon Square

gorillas Gorilla Nesting Sites

gorillas.extra (gorillas)

Gorilla Nesting Sites

hamster Aherne's hamster tumour data

heather Diggle's Heather Data

humberside Humberside Data on Childhood

Leukaemia and Lymphoma

humberside.convex (humberside)

Humberside Data on Childhood

Leukaemia and Lymphoma

hyytiala Scots pines and other trees at

Hyytiala

japanesepines Japanese Pines Point Pattern

lansing Lansing Woods Point Pattern

letterR Window in Shape of Letter R

longleaf Longleaf Pines Point Pattern

mucosa Cells in Gastric Mucosa

mucosa.subwin (mucosa)

Cells in Gastric Mucosa

murchison Murchison gold deposits

nbfires Point Patterns of New Brunswick

Forest Fires

nbfires.extra (nbfires)

Point Patterns of New Brunswick

Forest Fires

nbw.rect (nbfires)

Point Patterns of New Brunswick

Forest Fires

nbw.seg (nbfires)

Point Patterns of New Brunswick

Forest Fires

nztrees New Zealand Trees Point Pattern

osteo Osteocyte Lacunae Data:

Replicated Three-Dimensional

Point Patterns

paracou Kimboto trees at Paracou, French

Guiana

ponderosa Ponderosa Pine Tree Point Pattern

ponderosa.extra (ponderosa)

Ponderosa Pine Tree Point Pattern

pyramidal Pyramidal Neurons in Cingulate

Cortex

redwood California Redwoods Point Pattern

(Ripley's Subset)

redwood3 California Redwoods Point Pattern

(Ripley's Subset)

redwoodfull California Redwoods Point Pattern

(Entire Dataset)

redwoodfull.extra (redwoodfull)

California Redwoods Point Pattern

(Entire Dataset)

residualspaper Data and Code From JRSS

Discussion Paper on Residuals

shapley Galaxies in the Shapley

Supercluster

shapley.extra (shapley)

Galaxies in the Shapley

Supercluster

simba Simulated data from a two-group

experiment with replication

within each group.

simdat Simulated Point Pattern

simplenet Simple Example of Linear Network

spiders Spider Webs on Mortar Lines of a

Brick Wall

sporophores Sporophores Data

spruces Spruces Point Pattern

swedishpines Swedish Pines Point Pattern

urkiola Urkiola Woods Point Pattern

vesicles Vesicles Data

vesicles.extra (vesicles)

Vesicles Data

waka Trees in Waka national park

waterstriders Waterstriders data. Three

independent replications of a

point pattern formed by insects.

Data sets in package ‘spData’:

LO\_nb (house) Lucas county OH housing

SplashDams Data for Splash Dams in western

Oregon

afcon Spatial patterns of conflict in

Africa 1966-78

africa.rook.nb (afcon)

Spatial patterns of conflict in

Africa 1966-78

afxy (afcon) Spatial patterns of conflict in

Africa 1966-78

aggregating\_zones

Datasets to illustrate the

concept of spatial congruence

alaska Alaska multipolygon

auckland Marshall's infant mortality in

Auckland dataset

auckland.nb (auckland)

Marshall's infant mortality in

Auckland dataset

auckpolys (auckland)

Marshall's infant mortality in

Auckland dataset

baltimore House sales prices, Baltimore, MD

1978

bbs (columbus) Columbus OH spatial analysis data

set

boston.c (boston)

Corrected Boston Housing Data

boston.soi (boston)

Corrected Boston Housing Data

boston.utm (boston)

Corrected Boston Housing Data

coffee\_data World coffee production data

col.gal.nb (columbus)

Columbus OH spatial analysis data

set

columbus Columbus OH spatial analysis data

set

congruent Datasets to illustrate the

concept of spatial congruence

coords (columbus)

Columbus OH spatial analysis data

set

cycle\_hire Cycle hire points in London

cycle\_hire\_osm Cycle hire points in London from

OSM

dll (elect80) 1980 Presidential election

results

e80\_queen (elect80)

1980 Presidential election

results

eire.coords.utm (eire)

Eire data sets

eire.df (eire) Eire data sets

eire.nb (eire) Eire data sets

eire.polys.utm (eire)

Eire data sets

elect80 1980 Presidential election

results

elect80\_lw (elect80)

1980 Presidential election

results

elev Artificial elevation raster data

set

grain Artificial raster dataset

representing grain sizes

hawaii Hawaii multipolygon

hopkins Hopkins burnt savanna herb

remains

house Lucas county OH housing

huddersfield Prevalence of respiratory

symptoms

incongruent Datasets to illustrate the

concept of spatial congruence

jenks71 Illinois 1959 county gross farm

product value per acre

k4 (elect80) 1980 Presidential election

results

listw\_NY (nydata)

New York leukemia data

lnd The boroughs of London

nc.sids North Carolina SIDS data

ncCC89.nb (nc.sids)

North Carolina SIDS data

ncCR85.nb (nc.sids)

North Carolina SIDS data

nydata New York leukemia data

nz Regions in New Zealand

nz\_height High points in New Zealand

paper.nb (afcon)

Spatial patterns of conflict in

Africa 1966-78

polys (columbus)

Columbus OH spatial analysis data

set

seine Small river network in France

sidscents (nc.sids)

North Carolina SIDS data

sidspolys (nc.sids)

North Carolina SIDS data

state.vbm US State Visibility Based Map

trMat (house) Lucas county OH housing

urban\_agglomerations

Major urban areas worldwide

us\_states US states polygons

us\_states\_df the American Community Survey

(ACS) data

usa48.nb (used.cars)

US 1960 used car prices

used.cars US 1960 used car prices

wheat Mercer and Hall wheat yield data

world World country polygons

worldbank\_df World Bank data

x (getisord) Getis-Ord remote sensing example

data

xyz (getisord) Getis-Ord remote sensing example

data

y (getisord) Getis-Ord remote sensing example

data

Data sets in package ‘spdep’:

COL.OLD (oldcol)

Columbus OH spatial analysis data

set - old numbering

COL.nb (oldcol)

Columbus OH spatial analysis data

set - old numbering

bbs (columbus) Columbus OH spatial analysis data

set

col.gal.nb (columbus)

Columbus OH spatial analysis data

set

columbus Columbus OH spatial analysis data

set

coords (columbus)

Columbus OH spatial analysis data

set

eire.coords.utm (eire)

Eire data sets

eire.df (eire) Eire data sets

eire.nb (eire) Eire data sets

eire.polys.utm (eire)

Eire data sets

polys (columbus)

Columbus OH spatial analysis data

set

Data sets in package ‘survival’:

aml (leukemia) Acute Myelogenous Leukemia

survival data

bladder Bladder Cancer Recurrences

bladder1 (bladder)

Bladder Cancer Recurrences

bladder2 (bladder)

Bladder Cancer Recurrences

cancer NCCTG Lung Cancer Data

capacitor (reliability)

Reliability data sets

cgd Chronic Granulotamous Disease

data

cgd0 (cgd) Chronic Granulotomous Disease

data

colon Chemotherapy for Stage B/C colon

cancer

cracks (reliability)

Reliability data sets

diabetic Ddiabetic retinopathy

flchain Assay of serum free light chain

for 7874 subjects.

genfan (reliability)

Reliability data sets

heart Stanford Heart Transplant data

ifluid (reliability)

Reliability data sets

imotor (reliability)

Reliability data sets

jasa (heart) Stanford Heart Transplant data

jasa1 (heart) Stanford Heart Transplant data

kidney Kidney catheter data

leukemia Acute Myelogenous Leukemia

survival data

logan Data from the 1972-78 GSS data

used by Logan

lung NCCTG Lung Cancer Data

mgus Monoclonal gammopathy data

mgus1 (mgus) Monoclonal gammopathy data

mgus2 Monoclonal gammopathy data

myeloid Acute myeloid leukemia

nwtco Data from the National Wilm's

Tumor Study

ovarian Ovarian Cancer Survival Data

pbc Mayo Clinic Primary Biliary

Cirrhosis Data

pbcseq (pbc) Mayo Clinic Primary Biliary

Cirrhosis, sequential data

rats Rat treatment data from Mantel et

al

rats2 (rats) Rat data from Gail et al.

retinopathy Diabetic Retinopathy

rhDNase rhDNASE data set

solder Data from a soldering experiment

stanford2 More Stanford Heart Transplant

data

survexp.mn (survexp)

Census Data Sets for the Expected

Survival and Person Years

Functions

survexp.us (survexp)

Census Data Sets for the Expected

Survival and Person Years

Functions

survexp.usr (survexp)

Census Data Sets for the Expected

Survival and Person Years

Functions

tobin Tobin's Tobit data

transplant Liver transplant waiting list

turbine (reliability)

Reliability data sets

udca Data from a trial of

usrodeoxycholic acid

udca1 (udca) Data from a trial of

usrodeoxycholic acid

udca2 (udca) Data from a trial of

usrodeoxycholic acid

uspop2 Projected US Population

valveSeat (reliability)

Reliability data sets

veteran Veterans' Administration Lung

Cancer study