Corso di Statistica per Biotecnologi

Lezione2

Come importare dati nell'ambiente di lavoro di R

- Due esempi:
  - La vostra lista delle presenze
  - La lista dei cromosomi di genomi nucleari sequenziati e disponibili sulla banca dati GenBank
  - (http://www.ncbi.nlm.nih.gov/genomes/genlist.cgi?taxid=2759&type=8&name=Eukaryotae%20Complete%20Chromosomes)

Come importare dati nell'ambiente di lavoro di R

- La via più facile per importare i dati in R e' la funzione read.table
- read.table(file, header = FALSE, sep = "", quote = "\"", dec = ".", row.names, col.names, as.is = !stringsAsFactors, na.strings = "NA", colClasses = NA, nrows = -1, skip = 0, check.names = TRUE, fill = !blank.lines.skip, strip.white = FALSE, blank.lines.skip = TRUE, comment.char = "#", allowEscapes = FALSE, flush = FALSE, stringsAsFactors = default.stringsAsFactors(), fileEncoding = "", encoding = "unknown")
- File l'indirizzo del file da dove leggere i dati
- header = FALSE/TRUE il nome delle colonne e' scritto nella prima riga?
- sep = " " le colonne in una riga sono separate da che cosa
- quote = "\" ' " parti di testo da non importare
- dec = "." i numeri decimali sono delimitati con la virgola o con il punto
- strip.white = FALSE togliere eventuali spazi a molte e a valle di ogni valore

Come importare dati nell'ambiente di lavoro di R

- Se i dati sono da trascrivere a mano (primo esempio) e' meglio usare un foglio elettronico come Excell e poi esportare i dati come documento delimitato da tabulazione e poi sul nuovo file usare il comando read.table
- Se i dati provengono da internet (secondo esempio) e' meglio lavorare direttamente in un editor di testo (meglio notepad++ o textpad, va bene anche se pone qualche problema nel a capo di usare Word) e riportare ad un file di testo semplice delimitato da tabulazioni

Esempio 1

- studenti<-read.table('studenti2.txt',sep='\t', header=TRUE, quote='', strip.white=TRUE)
- dim(studenti)
- [1] 126 4
- Controllare che i numeri di righe e colonne corrispondano a quelli del foglio excell

```
summary(studenti)  #uno sguardo ai dati
is.factor(studenti$N)
is.factor(studenti$C)
is.factor(studenti$Portatile)
Levels((studenti$Nome)  # nomi scritti in maiuscolo e minisculo indistintamente
levels(studenti$Cognome)<-tolower(levels(studenti$Cognome))
levels(studenti$Nome)<-tolower(levels(studenti$Nome))  #errore causato da lettera
    accentata
levels(studenti$Nome)[73]<-"nicolo'
levels(studenti$Nome)<-tolower(levels(studenti$Nome))
```

Esempio 1

- Nomi e Cognomi più comuni

```
table(studenti$C)
table(table(studenti$C))
table(table(studenti$N))
sort(table(studenti$N), decreasing=TRUE)[1:8]
sort(table(studenti$C), decreasing=TRUE)[1:5]
```

- Domini di posta piu comuni

```
>studenti$Dominio<-studenti$email
>levels(studenti$Dominio)<-sub(".+@","",levels(studenti$Dominio))
```

La stringa di testo ".+@" e' un espressione regolare che indica uno o più (+) qualunque carattere (.) seguito da @

```
>sum(table(studenti$Dominio)
>length(studenti$Dominio)
```

Table non include nel calcolo gli NA

```
>100*sort(table(studenti$Dominio))/sum(table(studenti$Dominio))
```

Esempio 2

- Recarsi nel sito di GenBank
- Copiare ed incollare in un editor di testo
- Assicurarsi che la spaziatura sia una tabulazione
- Importare in R

```
genomi<-read.table("genomiNucleari.txt", sep='\t', header=TRUE)
> dim(genomi)
[1] 1095    9
>names(genomi)
[1] "organism"    "name"        "accession"   "length"      "proteins"
[6] "RNAs"        "genes"       "create_date" "update_date"
> summary(genomi)
…
```

Vedo length non e' considerato un numero perche c'e' la stringa 'nt' dopo il valore di numero

Esempio 2

Modifico le etichette del vettore di fattori

```
>L<-sub(' nt','',levels(genomi$length))
> summary(L)
   Length     Class      Mode
     1095 character character
```

Le trasformo in numeri

```
> L<-as.numeric(sub(' nt','',levels(genomi$length)))
> mode(L)
[1] "numeric"
> mean(L)
[1] 41305903
```

Trasformo il vettore originale da fattori a numerico

```
>L<-L[as.integer(genomi$length)]
```

Controllo la correttezza

```
> L[10]
[1] 14500692
> genomi$length[10]
[1] 14500692 nt
```

Esempio 2

- Sostituisco i dati nel data.frame

```
>genomi$length<-L
> summary(genomi)
```

- Rappresentazione grafica dei dati numerici

```
plot(genomi$length)
hist(genomi$length)
```

Essendoci valori molto grandi e molto piccoli e' piu agevole applicare una trasformzione logartimica per rappresentare i dati

```
hist(log(genomi$length,10))
plot(genomi$length, genomi$gene)
plot(genomi$length, genomi$gene, log='xy')
plot(log(genomi[,c('length','genes','proteins','RNAs')],10))
plot(genomi$length, genomi$gene, log='xy', col=1+(genomi$RNA>100))
```

Esempio 2

- Gestione dei dati temporali in R

```
>genomi$create_date[2]
[1] Jun 2 2003
111 Levels: Apr 11 2008  Apr 13 2007  Apr 18 2007  ... Sep 9 2004
>genomi$create_date[2]> genomi$create_date[4]
[1] NA
Warning message:
In Ops.factor(genomi$create_date[2], genomi$create_date[4]) :
  > senza senso per variabili factor
```

Bisogna leggere le date e trasformarle in un numero definito come numero di giorni da una data prefissata.

La funzione as.Date si occupa di questo

```
>as.Date("gen 1 1970","%b %d %Y")
>as.Date("1 1 1970","%m %d %Y")
>as.numeric(as.Date("gen 1 1970","%b %d %Y"))
```

Esempio 2

- Applicato ai nostri dati la funzione non svolge correttamente il suo lavoro

```
>as.Date(genomi$cr,"%b %d %Y")
```

- Perche le date sono scritte in inglese e il programma assume che siano scritte in italiano

```
Temp<-Sys.getlocale(category = "LC_TIME")
```

- Se usate windows dovete scrivere:

```
>Sys.setlocale("LC_TIME", "English_United States.1252")
```

Se usate Mac e linux dovete scrivere:

```
>Sys.setlocale("LC_TIME",  ”en_US.UTF-8")
>D<-as.Date(genomi$cr,"%b %d %Y")
>Sys.setlocale("LC_TIME", Temp)
>genomi$create_date<-D
>plot(genomi$length, genomi$cr)
```

Esempio 2

- I dati sono riportati per i singoli cromosomi come lavorare sui genomi interi? La funzione aggregate

```
>?aggregate
>G<-aggregate(genomi$length, by=list(genomi$organism), sum)
>D<-aggregate(genomi$cr, by=list(genomi$organism), mean)
>D
```

- La funzione "aggregate" fa perdere la rappresentazione di data ed dunque necessario ri-imporla

```
>plot(G[,2],as.Date(D[,2],origin="1970-01-01"))
```

Qual'e' il progetto genoma che e' che ha piu' differenza tra primi risultati e fine

```
>boxplot(split(as.numeric(genomi$cre),genomi$org),las=2)
```