Programação 2017/2018
Mestrado em Informática Médica / Curso de Especialização em Informática na Saúde
Nuno Moniz DCC - FCUP

Folha 2: Manipulação de Dados - Soluções

Importação de Dados

1. O site UCI ML Repository - Post-Operative Patient contém um conjunto de dados com a informação sobre para onde devem ser enviados pacientes na área de recobro pós-operatório: cuidados intensivos, área geral do hospital, ou para casa.

(a) Faça download do ficheiro post-operative.data.

(b) Importe o conjunto de dados para um data frame do R, garantindo que os valores desconhecidos são corretamente traduzidos para a nomenclatura do R.

```
path <- "/Users/rpribeiro/MyStuff/Teaching/PROG/1617/exercicios/folha2/"
pop <- read.csv(paste(path,"post-operative.data",sep=""),header=F,na.strings="?")
```

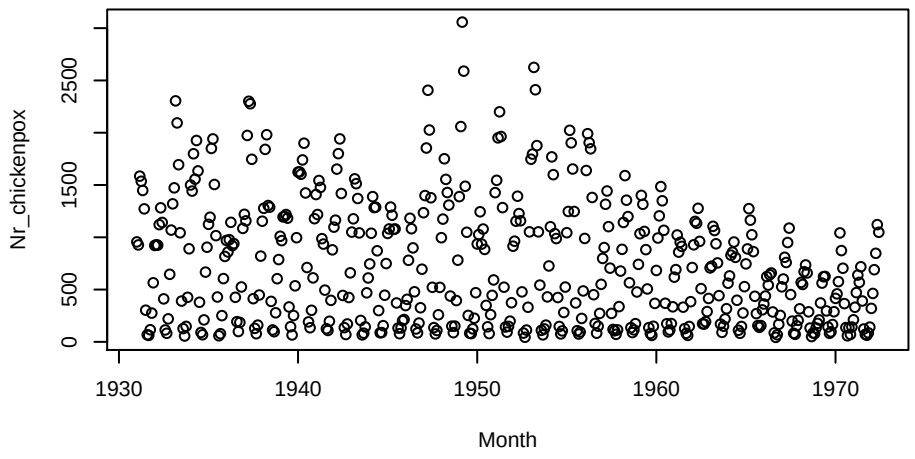
2. No ficheiro monthly-reported-number-of-chick.xlsx estão registados o número de casos de varicela reportados mensalmente em Nova York entre 1931 e 1972.

(a) Faça download do ficheiro e, usando a função `read_excel` da package `readxl` importe o conjunto de dados para um data frame do R.

```
library(readxl)
chick <- read_excel(paste(path,"monthly-reported-number-of-chick.xlsx",sep=""))
```

(b) Chame a função `plot` sobre o data frame criado.

```
plot(chick)
```

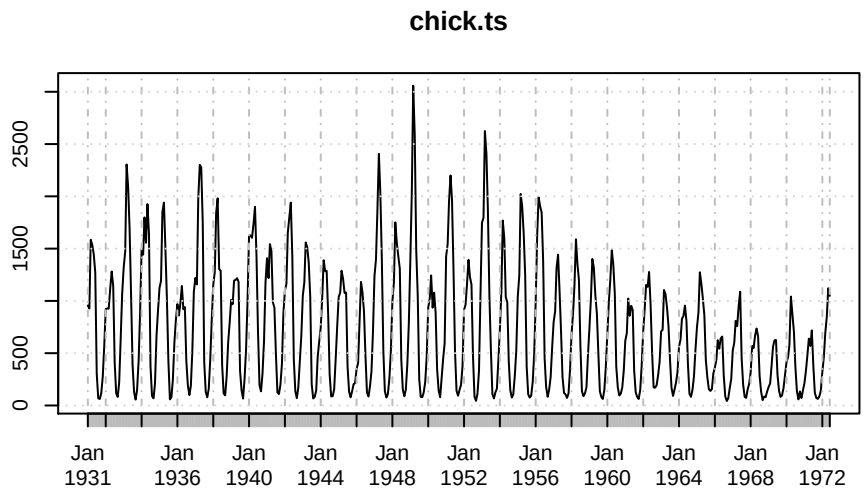


(c) Este conjunto de dados representa uma série temporal. Use a função `xts` da package `xts` para criar uma série temporal com valores indicados pela segunda coluna e datas indicadas pela primeira coluna.

```
library(xts)
chick.ts <- xts(chick$Nr_chickenpox,chick$Month)
```

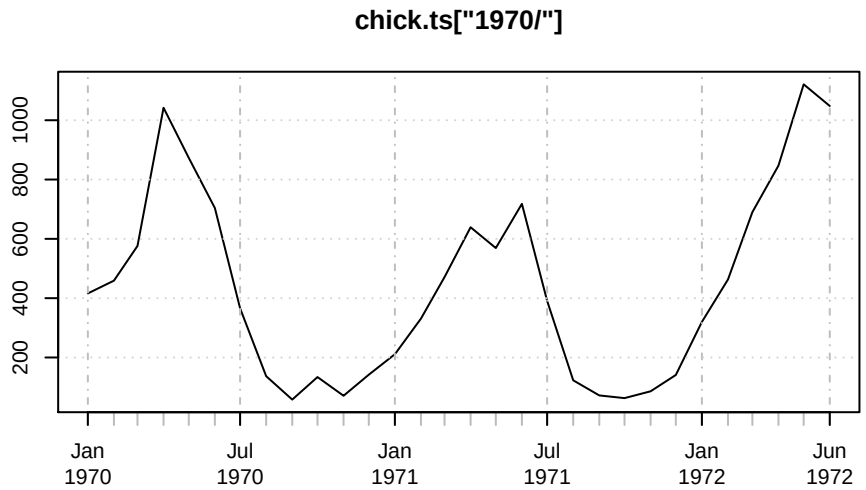
(d) Chame, novamente, a função `plot` mas agora sobre o objeto criado.

```
plot(chick.ts)
```



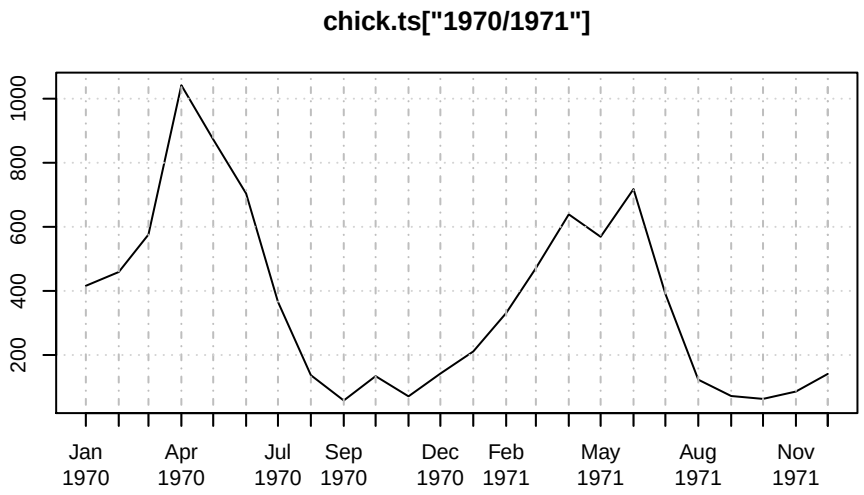
(e) Produza o gráfico correspondente ao período a partir de 1970.

```
plot(chick.ts["1970/"])
```



(f) Produza o gráfico correspondente ao período entre 1970 e 1971.

```r
plot(chick.ts["1970/1971"])
```



Manipulação de Dados com dplyr

3. Crie um *data frame table* a partir do conjunto de dados `Aids2` da package `MASS`. Sobre este conjunto de dados, execute as operações necessárias de forma a responder às seguintes questões.

```r
library(dplyr)
data(Aids2,package="MASS")
d <- tbl_df(Aids2)
```

(a) Inspecione as primeiras e as últimas linhas do dataset.

```r
head(d); tail(d)
```

```
## Source: local data frame [6 x 7]
##
##   state sex  diag death status T.categ age
## 1   NSW   M 10905 11081      D      hs  35
## 2   NSW   M 11029 11096      D      hs  53
## 3   NSW   M  9551  9983      D      hs  42
## 4   NSW   M  9577  9654      D    haem  44
## 5   NSW   M 10015 10290      D      hs  39
## 6   NSW   M  9971 10344      D      hs  36
## Source: local data frame [6 x 7]
##
##   state sex  diag death status T.categ age
## 1 Other   M 11359 11504      A      hs  27
## 2 Other   M 11475 11504      A     het  46
## 3 Other   F 11420 11504      A     het  34
## 4 Other   M 11496 11504      A    haem  49
## 5 Other   M 11460 11504      A      hs  55
## 6 Other   M 11448 11504      A      hs  37
```

(b) Mostre apenas as colunas cujo nome começa por `s`.

```
select(d,starts_with("s"))

## Source: local data frame [2,843 x 3]
##
##    state sex status
## 1    NSW   M      D
## 2    NSW   M      D
## 3    NSW   M      D
## 4    NSW   M      D
## 5    NSW   M      D
## 6    NSW   M      D
## 7    NSW   M      D
## 8    NSW   M      D
## 9    NSW   M      D
## 10   NSW   M      D
## ..   ... ...    ...
```

(c) Mostre apenas as linhas referentes a pacientes do sexo feminino que estejam vivos.

```
filter(d, sex=="M", status == "A")

## Source: local data frame [1,046 x 7]
##
##    state sex  diag death status T.categ age
## 1    NSW   M 10452 11504      A      hs  30
## 2    NSW   M 10923 11504      A    haem  21
## 3    NSW   M 10993 11504      A      hs  56
## 4    NSW   M 10996 11504      A      hs  38
## 5    NSW   M 10738 11504      A     het  26
## 6    NSW   M 11063 11504      A      id  39
## 7    NSW   M 11056 11504      A    haem  13
## 8    NSW   M 11283 11504      A      hs  34
## 9    NSW   M 11195 11504      A     het  39
## 10   NSW   M 10848 11504      A      hs  31
## ..   ... ...   ...   ...    ...     ... ...
```

(d) Ordene as linhas pela idade e pela categoria de transmissão do vírus.

```
arrange(d,age,T.categ)

## Source: local data frame [2,843 x 7]
##
##    state sex  diag death status T.categ age
## 1    QLD   M  9023  9039      D   blood   0
## 2    QLD   M  8963  8979      D   blood   0
## 3    QLD   M  8815  8815      D   blood   0
## 4  Other   F  9447  9447      D  mother   0
## 5    QLD   M  9199  9215      D   blood   1
## 6    NSW   M 11289 11504      A  mother   1
## 7    VIC   F 11312 11327      D  mother   1
## 8    NSW   M  9436 10938      D   blood   3
## 9    QLD   M  9868  9881      D   blood   3
## 10   NSW   M 10767 11504      A  mother   3
## ..   ... ...   ...   ...    ...     ... ...
```

(e) Crie, para os pacientes que morreram, uma nova coluna com a diferença em dias entre o momento do diagnóstico e o momento da morte.

```
filter(d,status=="D") %>% mutate(days=death-diag)

## Source: local data frame [1,761 x 8]
```

```
##
##    state sex  diag death status T.categ age days
## 1    NSW   M 10905 11081      D      hs  35  176
## 2    NSW   M 11029 11096      D      hs  53   67
## 3    NSW   M  9551  9983      D      hs  42  432
## 4    NSW   M  9577  9654      D    haem  44   77
## 5    NSW   M 10015 10290      D      hs  39  275
## 6    NSW   M  9971 10344      D      hs  36  373
## 7    NSW   M 10746 11135      D   other  36  389
## 8    NSW   M 10042 11069      D      hs  31 1027
## 9    NSW   M 10464 10956      D      hs  26  492
## 10   NSW   M 10439 10873      D    hsid  27  434
## ..   ... ...   ...   ...    ...     ... ...  ...
```

(f) Obtenha, para os pacientes que morreram, o número médio de dias de vida para cada categoria de transmissão.

```
filter(d,status=="D") %>% mutate(days=death-diag) %>%
  group_by(T.categ) %>% summarize(avg.days=mean(days))
```

```
## Source: local data frame [8 x 2]
##
##   T.categ avg.days
## 1      hs 367.1188
## 2    hsid 399.5556
## 3      id 394.4737
## 4     het 559.3529
## 5    haem 309.8276
## 6   blood 225.4211
## 7  mother 223.0000
## 8   other 218.2750
```

(g) Sobre o data set anterior obtenha, também para cada categoria, o número de pacientes total.

```
filter(d,status=="D") %>% mutate(days=death-diag) %>%
  group_by(T.categ) %>% summarize(avg.days=mean(days),total=n())
```

```
## Source: local data frame [8 x 3]
##
##   T.categ avg.days total
## 1      hs 367.1188  1532
## 2    hsid 399.5556    45
## 3      id 394.4737    19
## 4     het 559.3529    17
## 5    haem 309.8276    29
## 6   blood 225.4211    76
## 7  mother 223.0000     3
## 8   other 218.2750    40
```

(h) Adicione o sexo como outro nível de agrupamento e volte a executar a operação anterior.

```
filter(d,status=="D") %>% mutate(days=death-diag) %>%
  group_by(T.categ,sex) %>% summarize(avg.days=mean(days),total=n())
```

```
## Source: local data frame [12 x 4]
## Groups: T.categ
##
##    T.categ sex avg.days total
## 1       hs   M 367.1188  1532
## 2     hsid   M 399.5556    45
## 3       id   F 467.1667     6
## 4       id   M 360.9231    13
```

```
## 5      het   F 658.0000    10
## 6      het   M 418.4286     7
## 7     haem   M 309.8276    29
## 8    blood   F 200.4242    33
## 9    blood   M 244.6047    43
## 10  mother   F 223.0000     3
## 11   other   F  83.0000     1
## 12   other   M 221.7436    39
```