Validação de dados de Citrus em R

Elias T. Krainski & Paulo J. Ribeiro Jr.

Última Atualização: 2 de agosto de 2006

Erros de digitação ocorrem com bastante freqüência em grandes conjuntos de dados. Nos dados de doenças de plantas, também pode haver posições no talhão em que a planta já não existe mais. Neste caso, a respectiva posição da *lattice* que representa o talhão deve indicar esse fato e desconsideradas na análise. No caso específico de MSC, há também a validação temporal.

Para exemplo, será utilizado um conjunto de dados originais de MSC, estudado com detalhes no manual "Trabalhando com dados de Citrus em R". Carregando o conjunto de dados original:

```
> data(o303.geo)
```

1 Troca de códigos

Pode haver códigos nos dados que precisam ser recodificados. No caso dos dados `o303.geo`, pode-se fazer uma tabela com os códigos existentes nos dados.

```
> table.citrus(o303.geo)

object
    0     1     2     3     F     G     O     R
11090  6897  1363  4292    94     3     8   253
```

Neste caso, os códigos "G" e "O" representam plantas com "gomose" e "sadias", este último caso sendo um erro de digitação: devia ser 0 (zero) em lugar de *O* (letra ó). Ambos os casos deve ser 0 (zero).

A função `change.code()` pode ser utilizada para trocar um ou mais códigos por um ou mais outros códigos. Basta informar os códigos originais no argumento `ori.code` e os novos códigos no argumento `mod.code`.

Trocando códigos:

```
> cg303 <- change.code(data = o303.geo, ori.code = c("G", "O"),
+     mod.code = c(0, 0))
```

Inspencionando o objeto:

```
> cg303

Disease plant data in 25 evaluations of
20 rows of plants and 48 plants in each row.

> class(cg303)
```

```
[1] "citrus"  "geodata"
```

```
> table.citrus(cg303)
```

```
object
    0     1     2     3     F     R
11101  6897  1363  4292    94   253
```

```
> names(cg303)
```

```
[1] "coords"       "data"         "dates"        "n.subst.code"
```

```
> cg303$n.sub
```

```
$Gpor0
[1] 3

$Opor0
[1] 8
```

Observa-se que agora não há os códigos “G” e “O”

2 Seleção de dados

Em dados de citrus, freqüentemente ocorrem falhas (a planta não existe efetivamente na posição) ou replante (plantas plantadas posteriormente na posição e tem idade diferente das demais). Em algumas análises, essas plantas (posições) não são consideradas na análise, ou seja, considerar como inesistentes. Isso poderia ser feito trocando os códigos identificadores de plantas nessas condições, por `N`. Porém, é importante preservar a informação que se tem. Para isso, foi implementada a função `select.code()`. Esta função pode ser usada para realizar a separação desses dados, identificados por um ou mais códigos. Esse(es) códigos podem ser informados no argumento `unselect.cods`.

Fazendo a seleção:

```
> sg303 <- select.code(cg303, unselect.cods = c("F", "R"))
```

Inspecionando o objeto:

```
> sg303
```

```
Disease plant data in 25 evaluations of
20 rows of plants and 48 plants in each row.
```

```
> class(sg303)
```

```
[1] "citrus"  "geodata"
```

```
> table.citrus(sg303)
```

```
object
    0     1     2     3
11079  6897  1363  4286
```

```
> names(sg303)

[1] "coords"       "data"         "dates"        "n.subst.code"
[5] "unselect"
```

Os dados desconsiderados são colocados no elemento `unselect`, agrupados por cada código:

```
> names(sg303$unsel)

[1] "F" "R"
```

Cada elemento do elemento `unselect` contém as coordenadas e os dados das avaliações:

```
> names(sg303$unsel$F)

[1] "coords" "data"

> sg303$u$F

$coords
     row col
[1,] 150  12
[2,]  90  16
[3,]  90  20
[4,]  15  72

$data
    Av1 Av2 Av3 Av4 Av5 Av6 Av7 Av8 Av9 Av10 Av11 Av12 Av13 Av14 Av15 Av16
60  "F" "F" "F" "F" "F" "F" "F" "F" "F" "F"  "F"  "F"  "F"  "F"  "F"  "F"
72  "F" "F" "F" "F" "F" "F" "F" "F" "F" "F"  "F"  "F"  "F"  "F"  "F"  "F"
92  "F" "F" "F" "F" "F" "F" "F" "F" "F" "F"  "F"  "F"  "F"  "F"  "F"  "F"
342 "F" "F" "F" "F" "F" "F" "F" "F" "F" "F"  "F"  "F"  "F"  "F"  "F"  "F"
    Av17 Av18 Av19 Av20 Av21 Av22 Av23 Av24 Av25
60  "F"  "F"  "F"  "F"  "F"  "F"  "F"  "F"  "F"
72  "F"  "F"  "F"  "F"  "F"  "F"  "F"  "F"  "F"
92  "F"  "F"  "F"  "F"  "F"  "F"  "F"  "F"  "F"
342 "F"  "F"  "F"  "3"  "3"  "3"  "3"  "3"  "3"
```

3 Validação temporal - caso MSC

Os dados da MSC são codificados atribuido-se o código 1 à planta que inicia a doença, código 2 à planta com estado avançado da doença e 3 à planta morta. Até o momento é assumido que a planta não regride no estado da doença, então pode-se verificar a ocorrência de erros na seqüência temporal dos dados. Dessa forma, não deve ocorrer seqüências do tipo 00001000111122223, 000111211122223 ou 000011001112223.

No primeiro caso, é mais provável que na quinta avaliação a planta ainda estivesse sadia, pois nas três avaliações seguintes a planta estava saudável. No segundo caso, é mais provável que na sétima avaliação a planta estivesse ainda no estado inicial da doença. Assim, a metodologia adotada de correção de inconsistências nas seqüências temporais de códigos, compara o número de avaliações em que houve inconsistência, com o número de avaliações seguintes

à essas avaliações. No caso de empate, terceiro caso, optou-se por considerar que a o agravamento do estado da planta relmente ocorreu nas avaliações cinco e seis e a inconsistência está na sétima e oitava avaliações. Para corrigir essa inconsistência, pode-se utilizar a função `validStatusMSC.citrus()`:

A função `validStatusMSC()`, detecta as plantas com erros na sequência temporal do progresso da doença, podendo ser corrigidos automaticamente (`cor=TRUE`) ou não (`cor=FALSE`), neste caso sendo apenas identificados e separado dos demais dados. Definindo o argumento `cor=TRUE`, os dados invalidos são mantidos separados no elemento `invalids` e as linhas de dados onde havia inconsistência são colocados os dados validados.

Efetuando a validação temporal:

```
> g303 <- validStatusMSC.citrus(sg303, cor = TRUE)

9  inconsistences in  25 evaluations of 945 plants.
```

Inspecionando o objeto:

```
> g303

Disease plant data in 25 evaluations of
20 rows of plants and 48 plants in each row.

> class(g303)

[1] "citrus"  "geodata"

> names(g303)

[1] "coords"       "data"         "dates"        "n.subst.code"
[5] "unselect"     "invalids"
```

Os dados, coordenadas e atributos, com seqüência temporal inválida, são mantidos no elemento `invalids` do objeto:

```
> names(g303$inval)

[1] "coords" "data"

> g303$inv$dat

      Av1 Av2 Av3 Av4 Av5 Av6 Av7 Av8 Av9 Av10 Av11 Av12 Av13 Av14 Av15
 [1,]   0   0   0   0   0   0   1   1   1    1    1    1    1    3    1
 [2,]   0   0   0   0   0   1   0   0   1    1    1    1    1    1    1
 [3,]   0   0   0   0   0   1   0   0   0    0    1    1    1    1    1
 [4,]   0   0   0   0   0   0   0   0   0    1    0    0    1    1    1
 [5,]   0   0   0   0   0   0   0   0   0    1    1    0    0    0    1
 [6,]   0   0   0   0   0   1   1   2   1    1    1    1    1    1    1
 [7,]   1   0   3   3   3   3   3   3   3    3    3    3    3    3    3
 [8,]   0   1   0   2   2   2   2   2   2    2    2    2    2    2    2
 [9,]   0   0   2   2   2   2   0   0   0    0    0    0    0    0    0
      Av16 Av17 Av18 Av19 Av20 Av21 Av22 Av23 Av24 Av25
 [1,]    1    1    1    1    2    2    2    2    3    3
```

```
[2,]    2    2    2    2    3    3    3    3    3    3
[3,]    1    1    1    1    1    1    1    1    3    3
[4,]    1    1    1    1    1    1    1    1    3    3
[5,]    1    1    1    1    2    3    3    3    3    3
[6,]    1    2    2    2    3    3    3    3    3    3
[7,]    3    3    3    3    3    3    3    3    3    3
[8,]    2    3    3    3    3    3    3    3    3    3
[9,]    0    0    0    0    1    3    3    3    3    3
```

Agradecimentos

Este trabalho foi desenvolvido como parte das atividades do convênio firmado entre o Fundo de Defesa da Citricultura (FUNDECITRUS) e o Departamento de Estatística da Universidade Federal do Paraná e financiado pelo FUNDECITRUS.