

1

Criação de banco de dados

Criar um banco de dados com covariáveis mudando no tempo não é uma tarefa muito fácil. Neste apêndice daremos um exemplo da adaptação no próprio R, em um banco de dados com apenas 10 pacientes, de forma a acompanhar tudo que está sendo feito. As variáveis estão em dois arquivos: `aids.dat`, com as covariáveis que não mudam no tempo, e `aidscd4.dat`, com os registros das várias contagem de CD4 de cada paciente. Observe que ambos os arquivos possuem a variável `id` que é o número de identificação do paciente.

As variáveis do banco `aids.dat` são:

- `id`: número do registro dos pacientes;
- `ob`: óbito – 1 = sim, 0 = não;
- `dini`, `mini`, `aini`: dia, mês e ano do diagnóstico da Aids;
- `dfim`, `mfim`, `afim`: dia, mês e ano do óbito ou da censura;
- `dnas`, `mnas`, `anas`: dia, mês e ano de nascimento;
- `sexo`: M = masculino, F = feminino;
- `escol`: F = ensino fundamental, M = ensino médio, S = ensino superior.

```
> aids <- read.table("aids.dat",header=T)
> aids
```

id	ob	dini	mini	aini	dfim	mfim	afim	dnas	mnas	anas	sexo	esc
1	1	15	6	1994	14	12	1999	7	10	1937	M	F
2	1	20	3	1995	3	12	1997	21	NA	1945	M	F
3	0	1	6	1994	26	9	1997	30	8	1965	M	<NA>
4	1	15	8	1997	11	9	2001	6	4	1948	M	M
5	0	8	2	1997	30	10	2003	4	12	1970	M	M
6	0	20	1	1998	24	11	2000	25	8	1968	F	F
7	1	8	4	1999	31	7	2000	1	1	1964	F	M
8	0	30	6	1998	22	10	2002	19	12	1970	F	F
9	0	7	5	2001	5	11	2003	21	10	1961	M	S
10	0	24	7	2001	18	12	2003	1	1	1958	M	F

No segundo banco de dados temos as informações dos exames de CD4, com as seguintes variáveis:

- *id*: número do registro dos pacientes;
- *cd4*: contagem de células cd4;
- *dcd4*, *mcd4*, *acd4*: dia, mês e ano do exame.

```
> cd4 <- read.table("aidsd4.dat",header=T)
> cd4
```

id	cd4	dcd4	mcd4	acd4
1	1200	15	6	1994
1	600	31	10	1996
1	300	14	9	1999
2	430	20	3	1995
2	150	3	12	1996
3	750	1	6	1994
4	650	15	8	1997
4	500	28	8	1998
4	190	11	9	2000
5	1300	8	2	1997
6	900	20	1	1998
6	1200	24	8	2000
7	500	8	4	1999
8	1190	30	6	1998
8	1200	21	9	2000
8	1110	22	10	2001
9	1500	7	5	2001
9	680	5	11	2002
10	2500	24	7	2001

No R, para converter a data calendário em número de dias a partir de uma data de origem usamos a função `julian()` da biblioteca `chron`. Os argumentos dessa função são os vetores dos meses, dias e anos respectivamente. A origem padrão é a data de 1º de janeiro de 1970. Caso se deseje mudar a origem para 10 de abril de 1980, por exemplo, basta acrescentar o argumento `origin=c(4,10,1980)`. Vejamos, então, como podemos transformar esses dois arquivos em um único banco de dados com as variáveis `ini` (início), `fim` e `status`, necessárias para o modelo de Cox estendido.

```
> require(chron)
# Gerando a data do diagnóstico de Aids
> aids$dtini <- julian(aids$mini,aids$dini,aids$aini)
# Gerando a data do óbito ou último atendimento
> aids$dtfim <- julian(aids$mfim,aids$dfim,aids$afim)
# Gerando a data do exame de cd4
> cd4$dtcd4 <- julian(cd4$mcd4, cd4$dcd4, cd4$acd4)
> aids$dtini
```

```
[1] 8931 9209 8917 10088 9900 10246 10689 10407
[9] 11449 11527
```

Para criar a variável idade na data do diagnóstico de Aids, é só subtrair desta data a data de nascimento e dividir por 365,25, que é o número médio de dias do ano. Pode-se arredondar a idade ou usar o número de anos completos, utilizando-se as funções `round` e `floor`:

```
> aids$ida <- (aids$dtini - julian(aids$mnas, aids$dnas,
  aids$anas))/365.25
> aids$ida <- round(aids$ida,digits=2)
> aids$ida <- floor(aids$ida)
```

Para reunir todas as informações do paciente, com uma linha para cada medida de CD4, e incluindo as informações que não mudam ao longo do tempo, utiliza-se o comando `merge()`. Podem-se selecionar as variáveis que comporão o novo banco, mas sempre a variável que identifica o paciente deve ser mantida.

```
> dados <- merge(aids[,c(1,2,12:16)],cd4[,c(1,2,6)],by="id")
> dim(dados)
```

```
[1] 19 9
```

```
> dados
```

id	ob	sexo	esc	dtini	dtfim	ida	cd4	dtcd4
1	1	M	F	8931	10939	56	1200	8931
1	1	M	F	8931	10939	56	600	9800
1	1	M	F	8931	10939	56	300	10848
2	1	M	F	9209	10198	NA	430	9209
2	1	M	F	9209	10198	NA	150	9833
3	0	M	NA	8917	10130	28	750	8917
4	1	M	M	10088	11576	49	650	10088
4	1	M	M	10088	11576	49	500	10466
4	1	M	M	10088	11576	49	190	11211
5	0	M	M	9900	12355	26	1300	9900
6	0	F	F	10246	11285	29	900	10246
6	0	F	F	10246	11285	29	1200	11193
7	1	F	M	10689	11169	35	500	10689
8	0	F	F	10407	11982	27	1190	10407
8	0	F	F	10407	11982	27	1200	11221
8	0	F	F	10407	11982	27	1110	11617
9	0	M	S	11449	12361	39	1500	11449
9	0	M	S	11449	12361	39	680	11996
10	0	M	F	11527	12404	43	2500	11527

Observe as dimensões do banco: mesmo número de linhas que o banco *cd4*. Finalmente, para criar as variáveis *ini*, *fim* e *status*, sugerimos que se utilize uma função como a que é apresentada a seguir:

```
covdep<-function(obj){
  ind<-unique(obj$id)
  dadocomp<-NULL
  for (i in 1:length(ind)){
    aux<-obj[which(obj$id==ind[i]),]
    if (length(aux)!=1){
      ini<-aux$dtcd4
      fim<-c(aux$dtcd4[-1],unique(aux$dtfim))
      status<-rep(0,nrow(aux))
    }
  }
}
```

```

    status[length(status)]<-unique(aux$ob)
  }
  else{
    ini<-aux$dtcd4
    fim<-aux$dtfim
    status<-aux$ob
  }
  dadocomp<-rbind(dadocomp,data.frame(aux,ini,fim,status))
}
dadocomp
}

```

Para utilizar esta função, adapte-a ao seu banco de dados, trocando o nome das variáveis de identificação, do óbito e das datas para os nomes de seu banco de dados. A função *covdep* recebe como argumento apenas o nome do banco de dados.

```
> bancomudatempo <- covdep(dados)
> bancomudatempo
```

id	ob	sex	esc	dtini	dtfim	ida	cd4	dtcd4	ini	fim	status
1	1	M	F	8931	10939	56	1200	8931	8931	9800	0
1	1	M	F	8931	10939	56	600	9800	9800	10848	0
1	1	M	F	8931	10939	56	300	10848	10848	10939	1
2	1	M	F	9209	10198	NA	430	9209	9209	9833	0
2	1	M	F	9209	10198	NA	150	9833	9833	10198	1
3	0	M	<NA>	8917	10130	28	750	8917	8917	10130	0
4	1	M	M	10088	11576	49	650	10088	10088	10466	0
4	1	M	M	10088	11576	49	500	10466	10466	11211	0
4	1	M	M	10088	11576	49	190	11211	11211	11576	1
5	0	M	M	9900	12355	26	1300	9900	9900	12355	0
6	0	F	F	10246	11285	29	900	10246	10246	11193	0
6	0	F	F	10246	11285	29	1200	11193	11193	11285	0
7	1	F	M	10689	11169	35	500	10689	10689	11169	1
8	0	F	F	10407	11982	27	1190	10407	10407	11221	0
8	0	F	F	10407	11982	27	1200	11221	11221	11617	0
8	0	F	F	10407	11982	27	1110	11617	11617	11982	0
9	0	M	S	11449	12361	39	1500	11449	11449	11996	0
9	0	M	S	11449	12361	39	680	11996	11996	12361	0
10	0	M	F	11527	12404	43	2500	11527	11527	12404	0

Quaisquer que sejam a forma e o programa utilizados para criar o banco de dados, sempre selecione algumas linhas para conferir os registros gerados.