PONTIFÍCIA UNIVERSIDADE CATÓLICA DO RIO DE JANEIRO
Departamento de Engenharia Industrial
Rua Marquês de São Vicente, 225
22453-900 – Rio de Janeiro
Brasil

ENG1536 – Inferência Estatística

Laboratório – Guia de Estudo 7

Estudaremos agora o uso de **fatores** em R. É um estudo preparatório para você aprender como se fazem análises de variância em R (ANOVA), que será o objeto do próximo estudo. Uma das principais aplicações da inferência estatística é investigar a diferença que possa existir entre grupos diversos no valor esperado de alguma variável aleatória. Queremos saber se um grupo de pacientes que recebeu um novo medicamento tem uma sobrevida média maior do que um grupo que recebeu um placebo. Queremos saber se um grupo de alunos que estudou por um novo método didático tem um desempenho médio nas provas melhor do que um grupo que estudou pelo método antigo. Queremos saber se existe diferença entre os rendimentos médios de três campos de cultivo de soja que foram submetidos a três tipos diferentes de fertilizantes. Os exemplos são infinitos. Chamamos de “fator” (em inglês, *factor*), em Estatística, aquilo que distingue grupos nos dados. Chamamos de “nível” (*level*) cada um dos diferentes grupos que existem num fator. No primeiro exemplo acima, o fator é o tratamento e ele tem dois níveis: “medicação” e “placebo”. No segundo exemplo, o fator é o método didático, também com dois níveis: “novo” e “antigo”. No terceiro exemplo, o fator é o fertilizante, e existem três níveis: “tipo 1”, “tipo 2” e “tipo 3”.

No contexto do R, um fator é um vetor que usamos para especificar a que grupo (nível) pertence os elementos de outro vetor. No exemplo do medicamento, poderíamos ter os seguintes vetores:

```
sobrevida <- c(12.3,10.6,8.8,11.5,13.7,6.2,6.5,7.4)
tratamento <- c("M","M","P","M","M","P","P","P")
```

O primeiro vetor contém os tempos de sobrevida, em meses, de oito pacientes. O segundo vetor contém o grupo (medicamento ou placebo) a que cada paciente respectivamente pertence. (Observe que a ordem dos elementos do fator é crucial: se o paciente que sobreviveu 12,3 meses recebeu a medicação e esse dado é o primeiro elemento do vetor `sobrevida`, então o primeiro elemento do vetor `tratamento` tem de conter `"M"`. Por isso, essas informações normalmente são organizadas numa moldura de dados, não em vetores isolados como aqui.) O vetor `tratamento`, na verdade, não está pronto ainda para ser usado como um fator. Para que possamos aproveitar todas as facilidades que o R oferece, precisamos converter o vetor de texto num fator, através da função `factor`. Digite: `tratamentof <- factor(tratamento)`. Se você digitar agora `tratamentof` no console, deveria receber esta resposta:

```
[1] M M P M M P P P
Levels: M P
```

Quando você digita apenas o nome de um objeto na linha de comando para examinar o seu conteúdo, você na verdade está usando implicitamente a função `print`, que descobrimos num estudo anterior. Essa função trata objetos do tipo fator de uma maneira bem específica, como é visto acima. Os elementos que originalmente eram textos perdem as aspas, porque agora eles representam níveis. Além disso, a função `print` também faz automaticamente um levantamento de todos os níveis diferentes que existem dentro do fator, e apresenta uma lista deles ao usuário, logo abaixo do conteúdo do fator (depois da palavra “Levels”). Isso pode parecer redundante no exemplo, pois o fator é muito curto e só contém dois tipos diferentes de níveis; mas em bases de dados grandes essa pequena cortesia será útil. Para obter apenas essa lista dos valores diferentes de níveis dentro de um fator, você pode usar a função `levels`. Experimente digitar `levels(tratamentof)` no console.

O poder dos fatores em análise de dados começa a se revelar quando usamos a função `tapply`, que aprendemos a seguir. Ela pertence a uma família de funções em R que serve para realizar (ou “aplicar”, *to apply*) alguma computação sobre subconjuntos de uma base de dados. A função mais simples dessa família é `apply`. Serve para calcular o valor de uma função sobre os elementos das linhas, ou das colunas, de uma matriz. Exemplo:

```
m <- matrix(c(1,2,3,4),c(2,2)); apply(m,1,mean); apply(m,2,mean)
```

Na linha acima, a função `matrix`, que já encontramos brevemente, organiza os elementos do seu primeiro argumento (o vetor com os números 1, 2, 3 e 4) numa matriz `m` cujas dimensões são determinadas pelos dois elementos do segundo argumento (uma matriz 2x2, portanto). A matriz é preenchida coluna a coluna, da esquerda para a direita. Assim, a primeira linha da matriz contém os números 1 e 3; a segunda linha, os números 2 e 4. O segundo comando calcula a média de cada linha na matriz; o terceiro, a média de cada coluna da matriz. Estude a documentação da função `apply` se quiser aprender mais sobre a sua sintaxe.

A função `tapply` (*table apply*) é similar, mas agora a computação desejada é realizada com os valores de cada nível dos dados. Voltando ao exemplo da nova medicação, suponha que o pesquisador desejasse calcular a sobrevida média dos pacientes que receberam o medicamento e também a sobrevida média daqueles que receberam placebo. Eis o comando:

```
tapply(sobrevida,tratamentof,mean)
```

O primeiro argumento da função `tapply` especifica o vetor de dados numéricos; o segundo especifica o fator que divide os dados em diferentes níveis; por último, como na função `apply`, especificamos a função cujo valor desejamos calcular em cada nível (neste caso, a média). Neste exemplo, a sobrevida média dos pacientes que receberam o medicamento (12,025 meses) parece ter sido bem maior do que a dos que receberam placebo (7,225 meses). Mas será a diferença entre as médias estatisticamente significativa? Para começar a responder a isso, poderíamos obter o desvio padrão amostral da sobrevida em cada um dos dois grupos de pacientes (lembre-se de que a variância da média das sobrevidas é igual à variância de uma sobrevida dividida por *n*). Veja como usar uma função criada pelo usuário dentro de `tapply`:

```
sdavg <- function(amostra) sd(amostra)/sqrt(length(amostra))
30*tapply(sobrevida,tratamentof,sdavg)   #calculando em número de dias
```

Constatamos que o desvio padrão é pouco menor do que 20 dias nos dois grupos. Logo, muito menor do que a diferença de quase cinco meses entre as sobrevidas médias dos dois grupos. Isso é uma indicação rudimentar de que a medicação pode ser efetiva. Num estudo comercial sério, o número de pacientes nos dois grupos seria muito maior, e vários outros fatores seriam incluídos na pesquisa, tais como idade, sexo, comorbidades, etc. No nosso próximo estudo, aprenderemos mais sobre isso no contexto da análise de variância.

Lista de exercícios 7

(1) Crie um vetor de 50 elementos chamado `altura`, no qual os primeiros 25 elementos sejam números aleatórios uniformemente distribuídos entre 1.5 e 1.8, e os próximos 25 elementos sejam números aleatórios uniformemente distribuídos entre 1.6 e 2.0. Depois, crie um fator chamado `sexof` com níveis `M` e `H` que contenha a informação de que as 25 primeiras alturas no outro vetor são de mulheres e as demais alturas são de homens. Qual é a diferença entre o valor esperado (não a média amostral) da altura dos homens e da altura das mulheres?

(2) Calcule com um só comando, sem usar laços de repetição, a diferença entre a altura média dos homens e aquela das mulheres na amostra de alturas que você produziu no exercício anterior. Essa diferença é razoável?

(3) Calcule agora os desvios padrões amostrais das alturas médias dos homens e das mulheres. (Use o exemplo final do guia de estudo.) Compare com os desvios calculados com a diferença calculada no exercício anterior e comente.

(4) Repita os exercícios (1) a (3), mudando apenas o tamanho das amostras de alturas, de 25 alturas em cada sexo para somente duas alturas em cada. A diferença entre as médias amostrais parece significativa?

* * *