Estatística Computacional

Patrícia de Siqueira Ramos

PPGEAB
UNIFAL-MG

20 de Junho de 2018

Introdução

- O `R` utiliza algoritmos eficientes para calcular médias, variâncias e covariâncias
- Porém, se algoritmos ineficientes forem utilizados, os resultados podem ser incorretos ou imprecisos
- Algoritmos eficientes são mais úteis quando os dados têm grande magnitude ou apresentam valores muito próximos de 0

Algoritmos univariados

- Ideia básica: utilizar fórmulas recursivas
- Se queremos a média de n observações obtemos a média da primeira observação, depois acrescentamos a média da segunda e assim por diante (o mesmo procedimento pode ser aplicado para variâncias e covariâncias, no caso de mais de uma variável)
- Para uma amostra de tamanho n $X_1, X_2, \ldots X_n$, a média e a variância convencionais são obtidas por

$$\bar{X} = \frac{\sum_{i=1}^{n} X_i}{n} \qquad S^2 = \frac{1}{n-1}\left[\sum_{i=1}^{n} X_i^2 - \frac{\left(\sum_{i=1}^{n} X_i\right)^2}{n}\right]$$

Algoritmos univariados

- Alguns algoritmos tentaram adaptar essas fórmulas repassando as amostras duas vezes (eficientes, mas lentos)
- West (1979) apresentou um algoritmo que passa pela amostra uma única vez, atualizando a média e a variância para cada observação

Algoritmos univariados

- Para uma amostra de tamanho n,

$$\bar{X}_1 = X_1, \text{ se } n = 1$$

$$\bar{X}_2 = (X_1 + X_2)/2, \text{ se } n = 2$$

$$\vdots$$

$$\bar{X}_{k-1} = \frac{\sum_{i=1}^{k-1} X_i}{k-1}, \text{ no passo } (k-1)$$

$$\bar{X}_k = \frac{\sum_{i=1}^{k} X_i}{k}, \text{ no passo } k$$

Algoritmos univariados

- Podemos expressar a média do passo k em função da média do passo $k-1$

$$\begin{aligned}\bar{X}_k =& \frac{\sum_{i=1}^{k-1} X_i + X_k}{k} \\ =& \bar{X}_{k-1} - \frac{\bar{X}_{k-1}}{k} + \frac{X_k}{k}\end{aligned}$$

Algoritmos univariados

Equação recursiva:

$$\bar{X}_k = \bar{X}_{k-1} + \frac{X_k - \bar{X}_{k-1}}{k}, \tag{1}$$

para $2 \leq k \leq n$, *sendo* $\bar{X}_1 = X_1$.

Algoritmos univariados

- De forma similar, se definirmos as somas de quadrados corrigidas das k primeiras observações amostrais $1 < k \leq n$ por

$$W2_k = \sum_{i=1}^{k} X_i^2 - \frac{\left(\sum_{i=1}^{k} X_i\right)^2}{k},$$

- e a variância será dada por

$$S_k^2 = \frac{W2_k}{(k-1)},$$

Algoritmos univariados

- Expandindo e isolando o k-ésimo termo:

$$W2_k = \sum_{i=1}^{k-1} X_i^2 + X_k^2 - \frac{\left(\sum_{i=1}^{k-1} X_i + X_k\right)^2}{k}$$

$$= W2_{k-1} + \frac{k(X_k - \bar{X}_k)^2}{(k-1)}. \qquad (2)$$

Algoritmos univariados

- Se substituirmos $\bar{X}_k$ da equação (1) na equação (2), obtemos

$$W2_k = W2_{k-1} + \frac{(k-1)(X_k - \bar{X}_{k-1})^2}{k}, \tag{3}$$

para $2 \leq k \leq n$, sendo $W2_1 = 0$.

- A variância é obtida por

$$S^2 = \frac{W2_n}{(n-1)}. \tag{4}$$

Covariância

- Generalizar a expressão $W2_k$ para calcular a covariância S_{xy} entre as variáveis X e Y
- A expressão para a soma de produtos é

$$W2_{k,xy} = W2_{k-1,xy} + \frac{(k-1)(X_k - \bar{X}_{k-1})(Y_k - \bar{Y}_{k-1})}{k}, \qquad (5)$$

para $2 \leq k \leq n$, sendo $W2_{1,xy} = 0$.

- A covariância é obtida por

$$S_{xy} = \frac{W2_{n,xy}}{(n-1)}.$$

Algoritmos univariados

- Podemos estender os resultados vistos para obter somas das terceira e quarta potência dos desvios em relação à média
- As expressões obtidas são:

$$W3_k = W3_{k-1} + \frac{(k^2 - 3k + 2)(X_k - \bar{X}_{k-1})^3}{k^2} - \frac{3(X_k - \bar{X}_{k-1})W2_{k-1}}{k}$$

$$\begin{aligned} W4_k = & W4_{k-1} + \frac{(k^3 - 4k^2 + 6k - 3)(X_k - \bar{X}_{k-1})^4}{k^3} + \\ & + \frac{6(X_k - \bar{X}_{k-1})^2 W2_{k-1}}{k^2} - \frac{4(X_k - \bar{X}_{k-1})W3_{k-1}}{k} \end{aligned}$$

para $2 \leq k \leq n$, sendo $W3_1 = 0$ e $W4_1 = 0$.

Função medsqk

```
# função para retornar a média, somas de desvios em relação à média
# ao quadrado, ao cubo e quarta potência e variância
medsqk = function(x){
  n = length(x)
  if (n <= 1) stop('Dimensão do vetor deve ser maior que 1!')
  xb = x[1]
  W2 = 0
  W3 = 0
  W4 = 0
  for (ii in 2:n){
    aux = x[ii] - xb
    W4 = W4 + (ii**3 - 4*ii**2 + 6*ii - 3)*aux**4/ii**3 +
       6*W2*aux**2/ii**2 - 4*W3*aux/ii
    W3 = W3 + (ii**2 - 3*ii + 2)*aux**3/ii**2 - 3*W2*aux/ii
    W2 = W2 + (ii - 1)*aux**2/ii
    xb = xb + aux/ii
  }
  S2 = W2/(n - 1)
  return(list(media = xb, variancia = S2, SQ2 = W2, W3 = W3, W4 = W4))
}
x = c(1, 2, 3, 4, 5, 7, 8)
medsqk(x)
```

Algoritmos multivariados

- Extensão multivariada para obter o vetor de médias (e não só um valor $\bar{X}$) e as matrizes de somas de quadrados e produtos e de covariâncias
- Seja uma amostra de tamanho n em $\mathbb{R}^p$ dada por $\mathbf{X}_1, \mathbf{X}_2, \ldots \mathbf{X}_n$, sendo esses vetores dispostos em uma matriz $\mathbf{X}$ de dimensão $n \times p$

Algoritmos multivariados

- Para a média, no lugar de usarmos

$$\bar{\mathbf{X}} = \frac{\sum_{i=1}^{n} \mathbf{X}_i}{n}$$

utilizaremos

$$\bar{\mathbf{X}}_k = \bar{\mathbf{X}}_{k-1} + \frac{\mathbf{X}_k - \bar{\mathbf{X}}_{k-1}}{k}, \tag{6}$$

para $2 \leq k \leq n$, sendo $\bar{\mathbf{X}}_1 = \mathbf{X}_1$.

Algoritmos multivariados

- Da mesma forma, podemos adaptar a equação (3) para obtermos

$$\mathbf{W}_k = \mathbf{W}_{k-1} + \frac{(k-1)(\mathbf{X}_k - \bar{\mathbf{X}}_{k-1})(\mathbf{X}_k - \bar{\mathbf{X}}_{k-1})^T}{k}, \qquad (7)$$

  para $2 \leq k \leq n$, sendo $\mathbf{W}_1 = \mathbf{0}$, uma matriz de zeros com dimensão $p \times p$

- A matriz de covariâncias é obtida por

$$\mathbf{S} = \frac{\mathbf{W}_n}{(n-1)}. \qquad (8)$$

Algoritmos multivariados

- Da mesma forma, podemos adaptar a equação (3) para obtermos

$$\mathbf{W}_k = \mathbf{W}_{k-1} + \frac{(k-1)(\mathbf{X}_k - \bar{\mathbf{X}}_{k-1})(\mathbf{X}_k - \bar{\mathbf{X}}_{k-1})^T}{k}, \qquad (7)$$

  para $2 \leq k \leq n$, sendo $\mathbf{W}_1 = \mathbf{0}$, uma matriz de zeros com dimensão $p \times p$
- A matriz de covariâncias é obtida por

$$\mathbf{S} = \frac{\mathbf{W}_n}{(n-1)}. \qquad (8)$$

- A função `medcov` recebe uma matriz de dados com n linhas (observações) e p colunas (variáveis) e retorna o vetor de médias, a matriz de somas de quadrados e produtos e a matriz de covariâncias

Função medcov

```
# função para retornar o vetor de médias, a matriz de somas de
# quadrados e produtos e a matriz de covariâncias
medcov = function(x){
  n = nrow(x)
  p = ncol(x)
  if (n <= 1) stop('Dimensão linha da matriz deve ser maior que 1!')
  xb = x[1,]
  W = matrix(0, p, p)
  for (ii in 2:n){
    aux = x[ii,] - xb
    W = W + (ii - 1) * aux %*% t(aux) / ii
    xb = xb + aux/ii
  }
  S = W/(n - 1)
  return(list(vetmedia = xb, covariancia = S, SQP = W))
}
# uso
n = 1000
p = 5
library(mvtnorm) # Para gerarmos dados da normal multivariada
x = rmvnorm(n, matrix(0, p, 1), diag(p)) # simular da normal pentavariada
medcov(x)
# comparar os resultados
medcov(x)$vetmedia
apply(x, 2, mean)  # função pronta do R
medcov(x)$covariancia
var(x)  # função pronta do R
medcov(x)$SQP
(n-1) * var(x)  # função pronta do R
```

Algoritmos multivariados

- Como o R usa algoritmos de ótima qualidade para obter esses valores, não precisamos nos preocupar com a implementação de funções da forma que vimos nesta aula
- Porém, se formos utilizar um compilador da linguagem Pascal, Fortran, C++ etc., deveremos usar esses algoritmos para obter precisão elevada, especialmente se estivermos lidando com dados de grande magnitude ou muito próximos de zero

Exercícios

1. Implemente uma função chamada `m_var` que retorne a média para uma amostra de $n = 1.000$ números aleatórios. Use alguma estrutura de repetição para aplicar a fórmula:

$$\bar{X} = \frac{\sum_{i=1}^{n} X_i}{n}$$

2. Incremente a função `m_var` para que ela retorne também a variância. Use a fórmula:

$$S^2 = \frac{1}{n-1}\left[\sum_{i=1}^{n} X_i^2 - \frac{\left(\sum_{i=1}^{n} X_i\right)^2}{n}\right]$$

Confira se os resultados das duas funções foram os mesmos.