Analise Multivariada II

Luciane Alcoforado

agosto de 2018

Universidade Federal Fluminense
Instituto de Matemática e Estatística
Departamento de Estatística

1 Disciplina de Análise Multivariada II

Aulas: 3a. e 5a. de 11 às 13h

Recursos: Será necessário uso de computador/notebook com R, Rstudio e diversos pacotes instalados.

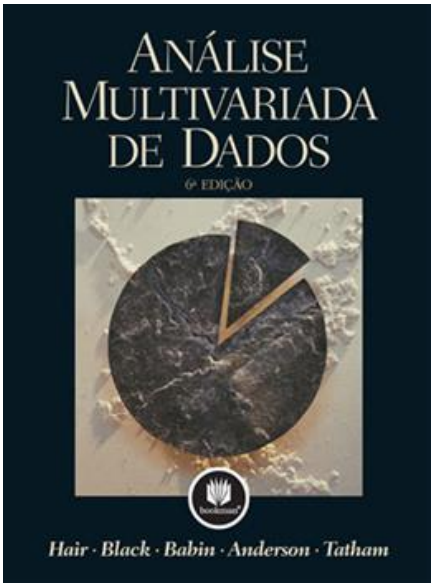
Avaliação: Avaliação escrita com conceitos básicos para análise multivariada (Cap 2 - Hair) + Trabalhos Práticos com entrega de relatório e apresentação.

Data da prova: 30/8 (sujeito a alteração)

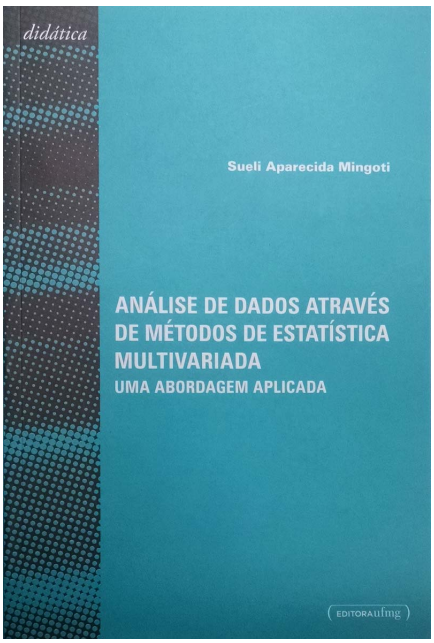
LatinR de 3 a 7/9 (professora irá neste evento apresentar trabalho)

Semana Acadêmica: 16 a 19/10 (os alunos devem participar da agenda de eventos)

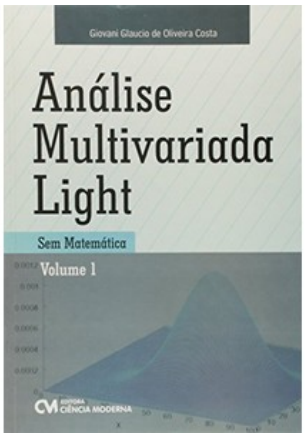
1.1 Bibliografia Básica



Análise Multivariada de Dados de autoria de Hair Jr, J.F. et al., 5a.edição, Porto Alegre: Bookman, 2005.



Análise de Dados através de Métodos de Estatística Multivariada, Sueli Aparecida Mingoti, Belo Horizonte: Editora UFMG, 2005.



Análise Multivariada Ligth, Giovani Glaucio de Oliveira Costa, Rio de Janeiro: Editora Ciência Moderna, 2017.

1.2 Aula 1: Conceitos Iniciais

Análise Multivariada refere-se a qualquer análise simultânea de mais de duas variáveis. (vide Hair pág 26). Seu propósito é medir, explicar e prever o grau de relacionamento entre variáveis estatísticas.

Variável Estatística é uma combinação linear de variáveis com pesos empiricamente determinados. As variáveis são determinadas pelo pesquisador e os pesos pela técnica multivariada para atingir um objetivo específico. (vide Hair pág 27)

Escala de Medida: dados métricos (quantitativos) e dados não-métricos (qualitativos)

Erro de medida é o grau em que os valores observados não são representativos dos valores "verdadeiros". Ex: Falta de habilidade do respondente em fornecer informação precisa como a renda familiar.

Erro tipo I: probabilidade de rejeitar a hipótese nula quando a mesma é verdadeira; é o falso positivo.

Poder do teste: probabilidade de rejeitar corretamente a hipótese nula quando esta deve ser rejeitada.

Dados Censurados observações incompletas de um indivíduo ou caso

Dados Perdidos informação não disponível de um indivíduo ou caso.

Método de atribuição: processo de estimação dos *dados perdidos* de um observação com base em valores válidos de outras variáveis.

Observação atípica: observação substancialmente diferente das outras, um valor extremo.

Homocedasticidade e Heterocedasticidade:quando a variância dos erros é constante ao longo do domínio de variáveis preditoras, diz-se que os dados são homocedásticos; quando a variância dos erros é crescente ou flutuante, diz-se que os dados são heterocedásticos.

Resíduo é a parte de uma variável dependente não explicada por uma técnica multivariada.

Variável dicotômica: variável com duas respostas possíveis: sim ou não, 0 ou 1, ausente ou presente, etc

2 Aula 2

2.1 Dados Perdidos - o que fazer?

Investigar os dados perdidos, perguntas a serem feitas:

- Os dados perdidos estão distribuídos ao acaso pelas observações ou são padrões distintos identificáveis?
- Qual é a frequência dos dados perdidos?

Para nos auxiliar na análise de padrão de dados perdidos usaremos a função *TestMCARNormality* do pacote **MissMech**.

Na prática, muitas vezes nos deparamos com conjuntos de dados perdidos. Excluir casos perdidos pode levar a inferência tendenciosa. Por outro lado, deve-se evitar adotar metodologias de atribuição de valores perdidos sem antes realizar a análise de padrão dos dados perdidos.

Desse modo, antes da adoção de métodos de atribuição, devemos realizar a análise de padrão dos dados perdidos.

O pacote **MissMech** (Jamshidian, Jalal e Jansen 2014) implementa testes MCAR (missing completely at random) de ponta desenvolvidos por Jamshidian e Jalal (2010). Como um subproduto da rotina principal, este pacote é capaz de testar a normalidade multivariada em alguns casos, e realizar uma série de outros testes. Para estudos aprofundados consultar [https://www.jstatsoft.org/article/view/v056i06]

Considere que n é o número de casos; p é o número de variáveis. O número de casos completos deve ser maior ou igual a 2*p, além disso, só utilizaremos este pacote no caso de haver valores perdidos e a função só se aplica se o número mínimo de casos para um grupo de casos perdidos for maior do que 1 e corresponde ao argumento *del.lesscases* = 1 (o default é 6); se o conjunto de dados não tiver mais do que 1 dado perdido para cada grupo de casos perdidos, a função retornará erro.

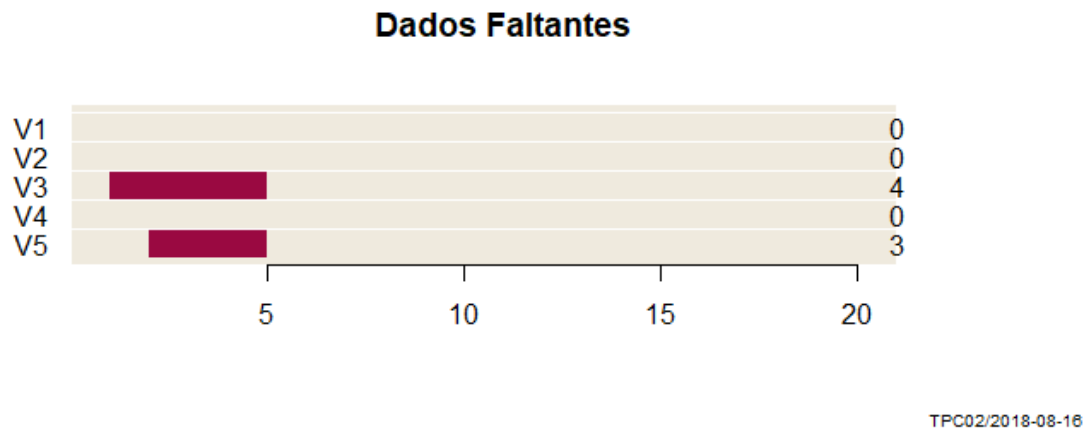
O teste de hipótese a ser realizado pela função *TestMCARNormality* do pacote **MissMech** é H_0 : As variância dos grupos são iguais (homocedasticidade)

```
#Exemplo: vamos criar um conjunto de dados com 20 casos e 5 variáveis.


#MAR - dados perdidos ao acaso (Missin at Random)

#MCAR - dados perdidos completamente ao acaso (Missing Completaly at Random)

require(tidyverse)

n <- 20
p <- 5
set.seed(1010)
y <- matrix(rnorm(n * p),nrow = n)

#Vamos definir alguns casos perdidos
y[1:4,3] <- NA
y[2:4,5] <- NA

#Visualizando os dados
y=as.tibble(y)
y
```

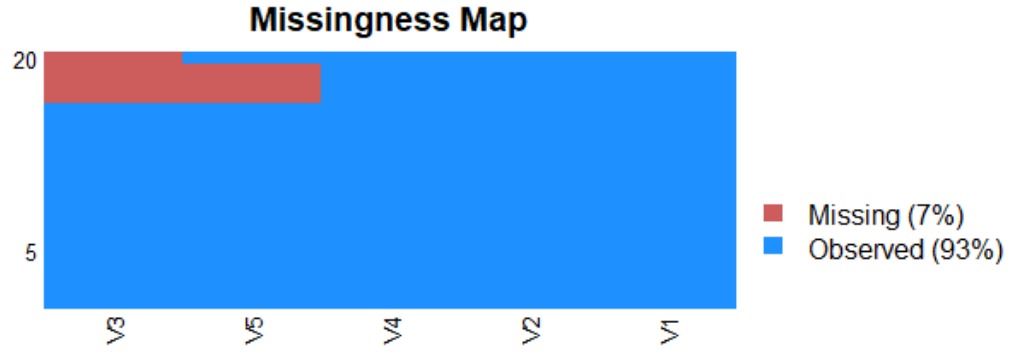
```
## # A tibble: 20 x 5
##         V1      V2       V3       V4       V5
##      <dbl>   <dbl>    <dbl>    <dbl>    <dbl>
##  1  0.132   0.0849  NA       1.57      1.23
##  2 -0.898   0.715   NA       0.645    NA
##  3  1.35   -0.966   NA       0.853    NA
##  4  0.420   0.462   NA      -0.360    NA
##  5 -0.288  -0.686  - 0.694   0.517     0.509
##  6  1.36    0.739    0.157  -1.98    - 1.36
##  7  1.94   -0.319    0.326   0.563   - 0.668
##  8 -2.36   -1.05   - 0.585  -0.437   - 1.55
##  9 -0.594  -0.286    0.337  -0.765     0.900
## 10 -0.545  -1.84     1.32   -0.00508 - 2.63
## 11 -0.196  -1.04     1.83    0.546   - 0.592
## 12 -0.830   1.92   - 0.773  -0.314     0.673
## 13 -1.44   -0.147  - 1.14   -0.752   - 1.31
## 14 -0.0761  0.330  - 0.372   1.05    - 0.0477
```

```
## 15  0.726  -1.04     0.0239 -0.409   - 1.14
## 16 -0.788   0.352  - 0.179  -2.35    - 0.528
## 17 -0.819  -0.275    0.930  -0.223   - 1.08
## 18  0.205  -0.387    1.59   -0.145   - 0.738
## 19 -0.952  -0.571  - 0.119  -0.00796   1.17
## 20 -0.966  -0.991    1.62    2.12    - 1.88
```

```
#Visualizando dados faltantes em gráfico
require(DescTools)
PlotMiss(y, main="Dados Faltantes")
```




```
require(Amelia)
missmap(y)
```




```
require(MissMech)
out <- TestMCARNormality(data=y, del.lesscases = 1)
```

```
summary(out)
```

```
##
## Number of imputation:  1
##
## Number of Patterns:  2
##
## Total number of cases used in the analysis:  19
##
##  Pattern(s) used:
##           V1   V2   V3   V4   V5   Number of cases
## group.1    1    1   NA    1   NA                 3
## group.2    1    1    1    1    1                16
##
##
##     Test of normality and Homoscedasticity:
##   ----------------------------------------
##
## Hawkins Test:
##
##     P-value for the Hawkins test of normality and homoscedasticity:  0.4359473
##
## Non-Parametric Test:
##
##     P-value for the non-parametric test of homoscedasticity:  0.584033
```

Observe que:

Há um valor perdido no caso 1 (variável 3);

Há 2 valores perdidos nos casos de 2 a 4 (variável 3 e variável 5) o que resultou em 2 grupos, grupo 1 com 3 casos e grupo 2 (caso completo) com 16 casos, totalizando 19 casos na análise;

O caso 1 não é considerado pois há apenas um grupo com este padrão;

O teste retorna p-valor 0.44 indicando a aceitação da normalidade multivariada e a aleatoriedade dos dados perdidos. Desse modo podemos realizar o processo de atribuição de valores aos casos perdidos.

Exercício:

1. Realize a análise para os dados da tabela 2.1 Ref. (Hair 6a.ed. pág 59)

TABELA 2-1 Exemplo hipotético de dados perdidos

						Dados perdidos por caso	
Identificação do caso	V_1	V_2	V_3	V_4	V_5	**Número**	**Percentual**
1	1,3	9,9	6,7	3,0	2,6	0	0
2	4,1	5,7			2,9	2	40
3		9,9		3,0		3	60
4	0,9	8,6		2,1	1,8	1	20
5	0,4	8,3		1,2	1,7	1	20
6	1,5	6,7	4,8		2,5	1	20
7	0,2	8,8	4,5	3,0	2,4	0	0
8	2,1	8,0	3,0	3,8	1,4	0	0
9	1,8	7,6		3,2	2,5	1	20
10	4,5	8,0		3,3	2,2	1	20
11	2,5	9,2		3,3	3,9	1	20
12	4,5	6,4	5,3	3,0	2,5	0	9
13					2,7	4	80
14	2,8	6,1	6,4		3,8	1	20
15	3,7			3,0		3	60
16	1,6	6,4	5,0		2,1	1	20
17	0,5	9,2		3,3	2,8	1	20
18	2,8	5,2	5,0		2,7	1	20
19	2,2	6,7		2,6	2,9	1	20
20	1,8	9,0	5,0	2,2	3,0	0	0
Dados perdidos por variável						Valores perdidos totais	
Número	2	2	11	6	2	Número: 23	
Percentual	10	10	55	30	10	Percentual: 23	

2. Simule um conjunto de dados com 300 casos e 8 variáveis e alguns dados perdidos distribuídos ao acaso. Realize a análise MCAR. OBS: quando há muitos dados pode-se omitir o argumento *del.lesscases*, neste caso o padrão da função *TestMCARNormality* do pacote **MissMech** é *del.lesscases*=6, o que significa que todos os grupo com número de casos menor do que 6 não são considerados na análise.

3. Realize o teste no conjunto de dados iris, imputando alguns valores perdidos, procurando estabelecer um padrão não aleatório.

```
irisna=tibble::as.tibble(iris)
irisna[1:10,1:2] <-NA
irisna[5:15,3]<-NA
irisna[1:60,1]<-NA
irisna[1:100,4]<-NA
irisna
out <- TestMCARNormality(data=irisna[,-5], del.lesscases = 1)

summary(out)
summary(irisna)
```

4. Apresente visualização de dados perdidos para os casos anteriores

Para aprofundar seus estudos em visualização de dados perdidos consulte [https://www.jstatsoft.org/index.php/jss/article/view/v068i06/v68i06.pdf]

2.1.1 Tratamentos para lidar com dados perdidos

- Incluir somente as observações com dados completos

Veja que no exemplo da tab 2-1 do Hair, 23% de dados perdidos levariam a excluir 15 casos dos 20, o que representa um conjunto de dados completos de apenas 5 casos!

- Eliminar casos ou variáveis problemáticas

Seguindo essa diretriz pensaríamos em eliminar o caso 13 e/ou a variável V3 que apresentam os maiores percentuais de dados perdidos. Não há uma orientação segura, desse modo o analista deverá considerar as perdas e ganhos no processo de elminação.

- Utilizar um método de atribuição (Hair pág 61 a 64)

O que são os métodos de atribuição? Qual a vantagem de se utilizar? Que pacotes do R posso utilizar para me auxiliar nesta tarefa?

Veja pacote **mice** função *mice* e pacote **VIM** função *kNN*.

```
y
```

```
## # A tibble: 20 x 5
##         V1      V2       V3       V4       V5
##      <dbl>   <dbl>    <dbl>    <dbl>    <dbl>
##  1  0.132   0.0849  NA       1.57      1.23
##  2 -0.898   0.715   NA       0.645    NA
##  3  1.35   -0.966   NA       0.853    NA
##  4  0.420   0.462   NA      -0.360    NA
##  5 -0.288  -0.686  - 0.694   0.517     0.509
##  6  1.36    0.739    0.157  -1.98    - 1.36
##  7  1.94   -0.319    0.326   0.563   - 0.668
##  8 -2.36   -1.05   - 0.585  -0.437   - 1.55
##  9 -0.594  -0.286    0.337  -0.765     0.900
## 10 -0.545  -1.84     1.32   -0.00508 - 2.63
## 11 -0.196  -1.04     1.83    0.546   - 0.592
## 12 -0.830   1.92   - 0.773  -0.314     0.673
## 13 -1.44   -0.147  - 1.14   -0.752   - 1.31
## 14 -0.0761  0.330  - 0.372   1.05    - 0.0477
## 15  0.726  -1.04     0.0239 -0.409   - 1.14
## 16 -0.788   0.352  - 0.179  -2.35    - 0.528
## 17 -0.819  -0.275    0.930  -0.223   - 1.08
## 18  0.205  -0.387    1.59   -0.145   - 0.738
```

```
## 19 -0.952  -0.571  - 0.119  -0.00796   1.17
## 20 -0.966  -0.991    1.62    2.12    - 1.88
```

```
require(mice)
complete(mice(y, print=FALSE))
```

```
##             V1          V2          V3          V4          V5
## 1   0.13154085  0.08487109 -0.17912952  1.567786166  1.22928982
## 2  -0.89790987  0.71486499 -0.11875332  0.644977071 -0.04770297
## 3   1.35194461 -0.96624080 -0.69431272  0.853215243 -1.14389243
## 4   0.42007469  0.46230137  0.15716619 -0.360481960  0.50897487
## 5  -0.28845968 -0.68605062 -0.69431272  0.517182812  0.50897487
## 6   1.36456136  0.73876017  0.15716619 -1.978282908 -1.36346885
## 7   1.93872531 -0.31882166  0.32590725  0.562623033 -0.66770311
## 8  -2.36098302 -1.05196788 -0.58511242 -0.437089345 -1.55340177
## 9  -0.59411413 -0.28592085  0.33682643 -0.765279280  0.89950783
## 10 -0.54455270 -1.83786541  1.31967884 -0.005079319 -2.62924785
## 11 -0.19590829 -1.03990318  1.83448990  0.546463286 -0.59212631
## 12 -0.83007345  1.92416803 -0.77276358 -0.313698940  0.67264880
## 13 -1.43538053 -0.14718335 -1.14073839 -0.751924345 -1.30631003
## 14 -0.07607103  0.32966774 -0.37153266  1.046418792 -0.04770297
## 15  0.72586799 -1.03558049  0.02385841 -0.408624843 -1.14389243
## 16 -0.78773341  0.35150814 -0.17912952 -2.349238899 -0.52763474
## 17 -0.81867894 -0.27533224  0.93012404 -0.223386083 -1.07938467
## 18  0.20464153 -0.38736785  1.58910048 -0.144527545 -0.73836031
## 19 -0.95241303 -0.57081846 -0.11875332 -0.007959542  1.16706441
## 20 -0.96580033 -0.99120874  1.61812729  2.124718712 -1.87932189
```

```
require(VIM)
y_knn=kNN(y)
y_knn
```

```
##             V1          V2          V3          V4          V5 V1_imp
## 1   0.13154085  0.08487109 -0.11875332  1.567786166  1.22928982  FALSE
## 2  -0.89790987  0.71486499 -0.37153266  0.644977071  0.67264880  FALSE
## 3   1.35194461 -0.96624080 -0.11875332  0.853215243 -0.04770297  FALSE
## 4   0.42007469  0.46230137 -0.11875332 -0.360481960  0.89950783  FALSE
## 5  -0.28845968 -0.68605062 -0.69431272  0.517182812  0.50897487  FALSE
## 6   1.36456136  0.73876017  0.15716619 -1.978282908 -1.36346885  FALSE
## 7   1.93872531 -0.31882166  0.32590725  0.562623033 -0.66770311  FALSE
## 8  -2.36098302 -1.05196788 -0.58511242 -0.437089345 -1.55340177  FALSE
```

```
## 9  -0.59411413 -0.28592085  0.33682643 -0.765279280  0.89950783  FALSE
## 10 -0.54455270 -1.83786541  1.31967884 -0.005079319 -2.62924785  FALSE
## 11 -0.19590829 -1.03990318  1.83448990  0.546463286 -0.59212631  FALSE
## 12 -0.83007345  1.92416803 -0.77276358 -0.313698940  0.67264880  FALSE
## 13 -1.43538053 -0.14718335 -1.14073839 -0.751924345 -1.30631003  FALSE
## 14 -0.07607103  0.32966774 -0.37153266  1.046418792 -0.04770297  FALSE
## 15  0.72586799 -1.03558049  0.02385841 -0.408624843 -1.14389243  FALSE
## 16 -0.78773341  0.35150814 -0.17912952 -2.349238899 -0.52763474  FALSE
## 17 -0.81867894 -0.27533224  0.93012404 -0.223386083 -1.07938467  FALSE
## 18  0.20464153 -0.38736785  1.58910048 -0.144527545 -0.73836031  FALSE
## 19 -0.95241303 -0.57081846 -0.11875332 -0.007959542  1.16706441  FALSE
## 20 -0.96580033 -0.99120874  1.61812729  2.124718712 -1.87932189  FALSE
##    V2_imp V3_imp V4_imp V5_imp
## 1   FALSE   TRUE  FALSE  FALSE
## 2   FALSE   TRUE  FALSE   TRUE
## 3   FALSE   TRUE  FALSE   TRUE
## 4   FALSE   TRUE  FALSE   TRUE
## 5   FALSE  FALSE  FALSE  FALSE
## 6   FALSE  FALSE  FALSE  FALSE
## 7   FALSE  FALSE  FALSE  FALSE
## 8   FALSE  FALSE  FALSE  FALSE
## 9   FALSE  FALSE  FALSE  FALSE
## 10  FALSE  FALSE  FALSE  FALSE
## 11  FALSE  FALSE  FALSE  FALSE
## 12  FALSE  FALSE  FALSE  FALSE
## 13  FALSE  FALSE  FALSE  FALSE
## 14  FALSE  FALSE  FALSE  FALSE
## 15  FALSE  FALSE  FALSE  FALSE
## 16  FALSE  FALSE  FALSE  FALSE
## 17  FALSE  FALSE  FALSE  FALSE
## 18  FALSE  FALSE  FALSE  FALSE
## 19  FALSE  FALSE  FALSE  FALSE
## 20  FALSE  FALSE  FALSE  FALSE
```

+Exercício: Com base nos dados da tabela 2.1 realize a avaliação de dados perdidos, realize o procedimento de substituição dos valores perdidos e faça uma comparação entre a média das variáveis antes e após a substituição dos valores perdidos.

3 Aula 3

3.1 Observações atípicas - o que fazer?

- Identificá-las

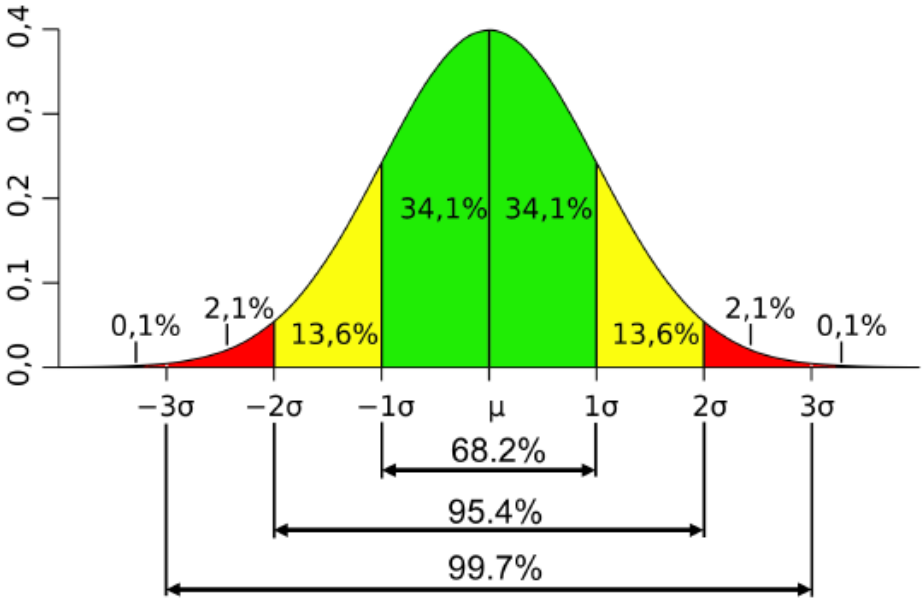
Motivos de ocorrência: erro de entrada de dados; resultado de um evento extraordinário para o qual haja uma explicação; resultado de um evento extraordinário para o qual não haja uma explicação; observações que estão no intervalo usual de valores para cada variável mas cuja combinação produz resultados fora do comum, por exemplo é possível observar pessoas com 1,50 a 1,90 m e com peso de 40 a 120 kg o que não é comum é uma pessoa de 1,90 m pesar 40 kg.

- Detecção Univariada

Identificar observações atípicas a partir do exame da distribuição de observações. Por exemplo na análise exploratória podemos utilizamos o boxplot ou se a variável possui distribuição normal, padronizar os valores observados que deverão estar entre -3 e 3 com 99.7% de probabilidade, fora deste intervalo é considerado atípico. Para amostras pequenas (80 ou menos), as diretrizes em Hair sugerem escores padronizados entre -2.5 e 2.5; caso contrário é considerado atípico.

Detection of Univariate Outliers: Location & Scale-Based Intervals in R

Gaussian distribution: $\mu \pm k \cdot \sigma, \quad k = 1,2,3$

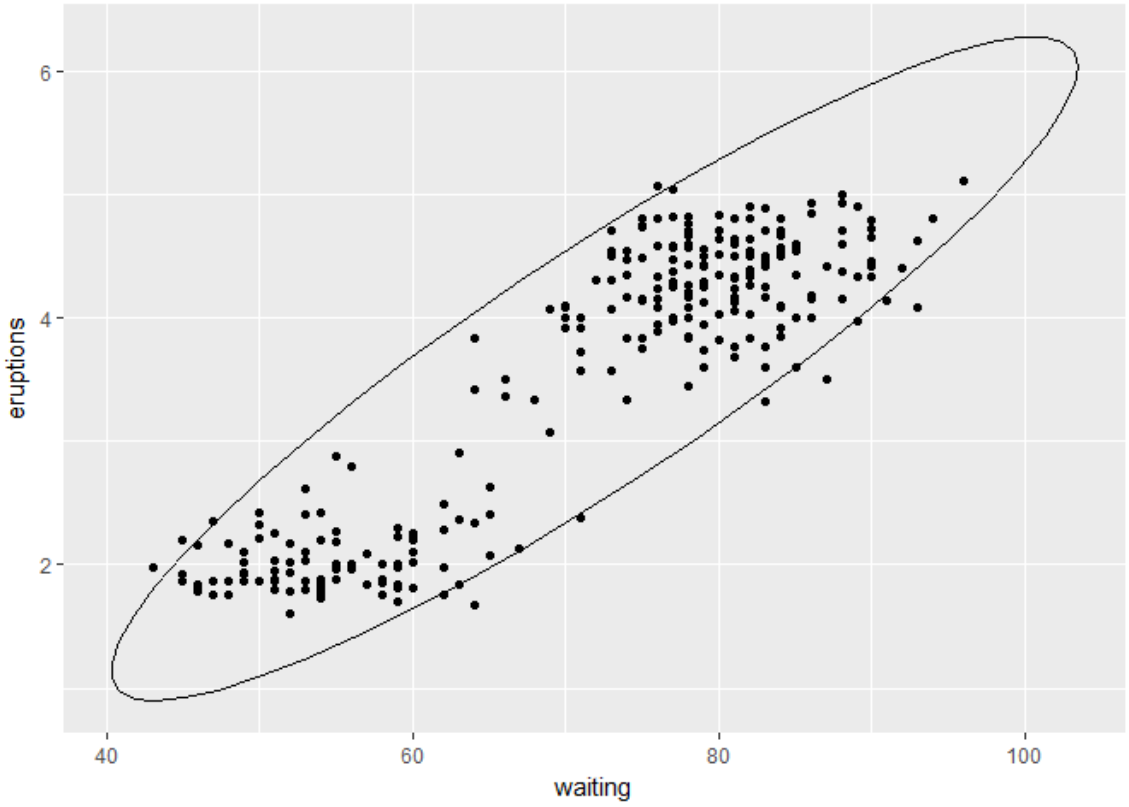


Source: http://www.muelaner.com/wp-content/uploads/2013/07/Standard_deviation_diagram.png

- Detecção bivariada

Utiliza-se o diagrama de dispersão. Os valores atípicos são aqueles que caem fora da elipse que representa o intervalo de confiança da distribuição normal bivariada.

```
library(ggplot2)
ggplot(faithful, aes(waiting, eruptions)) +
  geom_point() +
  stat_ellipse()
```



- Detecção Multivariada

Utiliza-se a Medida de Mahalanobis, é uma medida de distância no espaço multidimensional de cada observação em relação ao centro médio das observações. É possível realizar testes de significância para a medida de Mahalanobis.

```
df=data.frame(
  x1=c(1,2,3,5,3,1,8,4,5,4),      x2=c(2,1,3,5,3,3,8,4,5,5),
  x3=c(1,1,2,3,3,3,8,5,5,5), x4=c(3,1,2,5,2,1,7,4,4,4))
df #dados
```

```
##    x1 x2 x3 x4
## 1   1  2  1  3
## 2   2  1  1  1
## 3   3  3  2  2
## 4   5  5  3  5
## 5   3  3  3  2
## 6   1  3  3  1
## 7   8  8  8  7
## 8   4  4  5  4
## 9   5  5  5  4
## 10  4  5  5  4
```

```
cor(df) #matriz de correlação
```

```
##           x1        x2        x3        x4
## x1 1.0000000 0.9214520 0.8553220 0.8941547
## x2 0.9214520 1.0000000 0.9254530 0.9052685
## x3 0.8553220 0.9254530 1.0000000 0.7914936
## x4 0.8941547 0.9052685 0.7914936 1.0000000
```

```
d2=mahalanobis(df, center=colMeans(df),cov=cov(df))

#Se a amostra tiver distribuição aproximadamente normal
# a distância de mahalanobis terá distribuição
#qui-quadrada com g = n.variaveis da amostra graus de liberdade.

d2
```

```
##  [1] 6.3040841 4.7731721 2.3674093 4.9064417 1.0433109 6.0088335 4.5807909
##  [8] 4.1387594 0.7264117 1.1507864
```

```
qchisq(.975, ncol(df)) #distância acima do percentil 95 indica outlier.
```

```
## [1] 11.14329
```

```
#Neste exemplo não houve detecção de outlier multivariado usando a distância de Mahalanobis
```

4 Aula 4:

4.1 Normalidade Multivariada

- Teste univariado da normalidade

- Teste multivariado da normalidade: se uma variável é normal multivariada também é normal univariada. A recíproca nem sempre é verdadeira.
- Inicie testando a normalidade univariada: teste de shapiro ou teste de Kolmogorov-Smirnov

```
apply(df,2,shapiro.test)
```

```
## $x1
##
##  Shapiro-Wilk normality test
##
## data:  newX[, i]
## W = 0.93249, p-value = 0.4729
##
##
## $x2
##
##  Shapiro-Wilk normality test
##
## data:  newX[, i]
## W = 0.9366, p-value = 0.5158
##
##
## $x3
##
##  Shapiro-Wilk normality test
##
## data:  newX[, i]
## W = 0.91261, p-value = 0.2994
##
##
## $x4
##
##  Shapiro-Wilk normality test
##
## data:  newX[, i]
## W = 0.93298, p-value = 0.4778
```

- Pacote para testar normalidade multivariada:

```
mvnormtest::mshapiro.test(t(df))
```

```
##
##  Shapiro-Wilk normality test
##
## data:  Z
## W = 0.79992, p-value = 0.01446
```

Vamos criar um conjunto de dados com distribuição normal multivariada

```
#Teste muito sensível, tendência a rejeitar a normalidade
n <- 300
p <- 4
set.seed(1010)
y <- matrix(rnorm(n * p),nrow = n)

mvnormtest::mshapiro.test(t(y))
```

```
##
##  Shapiro-Wilk normality test
##
## data:  Z
## W = 0.99435, p-value = 0.3314
```

Exercício

1. Teste a normalidade multivariada nas variáveis numéricas do conjunto de dados iris
2. Crie um conjunto de dados multivariado com n = 300 e p = 5 e aplique o teste de shapiro multivariado ao mesmo.

```
#1
irism=as.matrix(iris[,-5])

mvnormtest::mshapiro.test(t(irism))

#2. Semelhante ao exemplo
```

3. Pesquise sobre o pacote MVN. Para que serve este pacote?

- Transformações para atingir a normalidade (Hair pág 81)

Que transformações são possíveis para atingir a normalidade dos dados? Realize uma pesquisa sobre este assunto, apresente um exemplo completo.

5 Aula 5

5.1 Dados não métricos com variáveis dicotômicas

Variável Sexo, possui 2 categorias feminino ou masculino. Definimos uma variável dicotômica X1 = 1 se feminino ocorre e 0 em caso contrário. Ou ainda X1 = 1 se masculino ocorre e 0 em caso contrário. A categoria omitida refere-se ao grupo de comparação. Assim na modelagem se X1 = 1 for para o caso feminino, estaremos comparando o resultado feminino em relação a categoria omitida que é o grupo masculino. (Hair pg 86 e 87 tabela 2.13 e 2.14)

5.2 Multicolinearidade

Ocorre quando qualquer variável independente é altamente correlacionada com o conjunto de outra variáveis independentes.

O ideal é ter diversas variáveis independentes altamente correlacionadas com a variável dependente, mas com pouca correlação entre elas próprias

5.3 Os efeitos da multicolinearidade

O caso extremo de colinearidade ou multicolinearidade no qual uma variável independente é perfeitamente prevista (uma correlação de ± 1,0) por uma ou mais variáveis independentes.Modelos de regressão não podem ser estimados quando existe uma singularidade. O pesquisador deve omitir uma ou mais das variáveis independentes envolvidas para remover a singularidade.

```
M=matrix(c(1,2,4,2,4,8,3,6,12,5,10,1),ncol=3,nrow=4, byrow = T)
M
```

```
##      [,1] [,2] [,3]
## [1,]    1    2    4
## [2,]    2    4    8
## [3,]    3    6   12
## [4,]    5   10    1
```

```
cor(M)
```

```
##            [,1]       [,2]       [,3]
## [1,]  1.0000000  1.0000000 -0.3159813
## [2,]  1.0000000  1.0000000 -0.3159813
## [3,] -0.3159813 -0.3159813  1.0000000
```

```
lm(M[,3]~M[,1]+M[,2]) #singularidade não permite estimar o parâmetro
```

```
##
## Call:
## lm(formula = M[, 3] ~ M[, 1] + M[, 2])
##
## Coefficients:
## (Intercept)       M[, 1]       M[, 2]
##      8.6857      -0.8857           NA
```

```
lm(M[,1]~M[,2]+M[,3])
```

```
##
## Call:
## lm(formula = M[, 1] ~ M[, 2] + M[, 3])
##
## Coefficients:
## (Intercept)       M[, 2]       M[, 3]
##   4.441e-16    5.000e-01    1.411e-17
```

5.4 Avaliação da multicolinearidade

Hair (pag 190 6a. edicao) Uma questão-chave na interpretação da variável estatística de regressão é a correlação entre as variáveis independentes. Esse é um problema de dados, e não de especificação de modelo. A situação ideal para um pesquisador seria ter diversas variáveis independentes altamente correlacionadas com a variável dependente, mas com pouca correlação entre elas próprias.

A tarefa do pesquisador inclui o seguinte:

- Avaliar o grau de multicolinearidade.
- Determinar seu impacto sobre os resultados.
- Aplicar as necessárias ações corretivas, se for o caso.

A maneira mais simples e óbvia de identificar colinearidade é um exame da matriz de correlação para as variáveis independentes. A presença de elevadas correlações (geralmente 0,90 ou maiores) é a primeira indicação de colinearidade substancial. No entanto, a falta de valores elevados de correlação não garante ausência de colinearidade.

Colinearidade pode ser proveniente do efeito combinado de duas ou mais variáveis independentes (o que se chama de multicolinearidade). Para avaliar multicolinearidade precisamos de uma medida que expresse o grau em que cada variável independente é explicada pelo conjunto de outras variáveis independentes. Em termos simples, cada variável independente se torna uma variável dependente e é regredida relativamente às demais variáveis independentes. As duas medidas mais comuns para se avaliar colinearidade aos pares ou múltipla são a tolerância e sua inversa, o fator de inflação de variância.

5.5 Tolerância.

Uma medida direta de multicolinearidade é tolerância, a qual é defi nida como a quantia de variabilidade da variável independente selecionada não explicada pelas outras variáveis independentes. Assim, para qualquer modelo de regressão com duas ou mais variáveis independentes, a tolerância pode ser simplesmente defi nida em dois passos:

1. Considere cada variável independente, uma por vez, e calcule R2* – a quantia da variável em questão que é explicada por todas as demais variáveis independentes no modelo de regressão. Neste processo, a variável independente escolhida é transformada em uma dependente prevista pelas demais.

2. Tolerância é então calculada como 1 – R2*. Por exemplo, se as outras variáveis independentes explicam 25% da variável independente X1 (R2* = 0,25), então o valor de tolerância de X1 é 0,75 (1,0 – 0,25 = 0,75).

O valor de tolerância deve ser alto, o que significa um pequeno grau de multicolinearidade (i.e., as outras variáveis independentes coletivamente não têm qualquer quantia considerável de variância compartilhada). A determinação de níveis apropriados de tolerância será abordada em uma seção adiante.

5.6 Exercício:

Pegue uma base de dados (Orange) multivariada e realize a análise de dados discrepantes, multicolinearidade e teste de normalidade.

6 Avaliação dos conceitos iniciais

Prova escrita.

6.1 Referências

Cheng, X., Cook, D., & Hofmann, H. (2015). Visually Exploring Missing Values in Multivariable Data Using a Graphical User Interface. Journal of Statistical Software, 68(6), 1 - 23. doi:http://dx.doi.org/10.18637/jss.v068.i06

Jamshidian, M., Jalal, S., & Jansen, C. (2014). MissMech: An R Package for Testing Homoscedasticity, Multivariate Normality, and Missing Completely at Random (MCAR). Journal of Statistical Software, 56(6), 1 - 31. doi:http://dx.doi.org/10.18637/jss.v056.i06

```
print(sessionInfo(), locale = FALSE)
```

```
## R version 3.4.3 (2017-11-30)
## Platform: x86_64-w64-mingw32/x64 (64-bit)
## Running under: Windows 10 x64 (build 17134)
##
## Matrix products: default
##
## attached base packages:
## [1] grid      stats     graphics  grDevices utils     datasets  methods
## [8] base
##
## other attached packages:
##  [1] VIM_4.7.0         data.table_1.11.4 colorspace_1.3-2
##  [4] mice_3.1.0        lattice_0.20-35   MissMech_1.0.2
##  [7] Amelia_1.7.5      Rcpp_0.12.17      DescTools_0.99.23
## [10] forcats_0.2.0     stringr_1.3.1     dplyr_0.7.5
## [13] purrr_0.2.4       readr_1.1.1       tidyr_0.8.1
## [16] tibble_1.4.2      ggplot2_2.2.1     tidyverse_1.2.1
##
## loaded via a namespace (and not attached):
##  [1] nlme_3.1-131      lubridate_1.7.4   httr_1.3.1
##  [4] rprojroot_1.3-2   tools_3.4.3       backports_1.1.2
##  [7] utf8_1.1.3        R6_2.2.2          rpart_4.1-11
## [10] lazyeval_0.2.1    jomo_2.6-2        nnet_7.3-12
## [13] sp_1.3-1          tidyselect_0.2.3  mnormt_1.5-5
## [16] curl_3.1          compiler_3.4.3    cli_1.0.0
## [19] rvest_0.3.2       expm_0.999-2      xml2_1.2.0
## [22] labeling_0.3      scales_0.5.0      lmtest_0.9-36
## [25] DEoptimR_1.0-8    mvtnorm_1.0-7     psych_1.8.4
## [28] robustbase_0.93-0 digest_0.6.15     foreign_0.8-69
## [31] minqa_1.2.4       rmarkdown_1.10    rio_0.5.9
## [34] pkgconfig_2.0.1   htmltools_0.3.6   lme4_1.1-15
## [37] manipulate_1.0.1  rlang_0.2.1       readxl_1.0.0
## [40] rstudioapi_0.7    prettydoc_0.2.1   bindr_0.1.1
## [43] zoo_1.8-1         jsonlite_1.5      car_3.0-0
## [46] magrittr_1.5      Matrix_1.2-14     munsell_0.4.3
## [49] abind_1.4-5       stringi_1.1.7     yaml_2.1.18
## [52] carData_3.0-0     MASS_7.3-50       plyr_1.8.4
## [55] parallel_3.4.3    mitml_0.3-5       crayon_1.3.4
## [58] haven_1.1.1       splines_3.4.3     hms_0.4.1
## [61] knitr_1.20        pillar_1.1.0      boot_1.3-20
## [64] mvnormtest_0.1-9  reshape2_1.4.3    pan_1.6
## [67] glue_1.2.0        evaluate_0.10.1   laeken_0.4.6
```

```
## [70] modelr_0.1.1      vcd_1.4-4          nloptr_1.0.4
## [73] cellranger_1.1.0  gtable_0.2.0       assertthat_0.2.0
## [76] openxlsx_4.0.17   broom_0.4.3        e1071_1.6-8
## [79] class_7.3-14      survival_2.41-3    bindrcpp_0.2.2
```