Pesquisa Quantitativa

Lupércio França Bessegato
Mestrado em Administração/UFJF

Análise Exploratória de Dados

Roteiro Geral

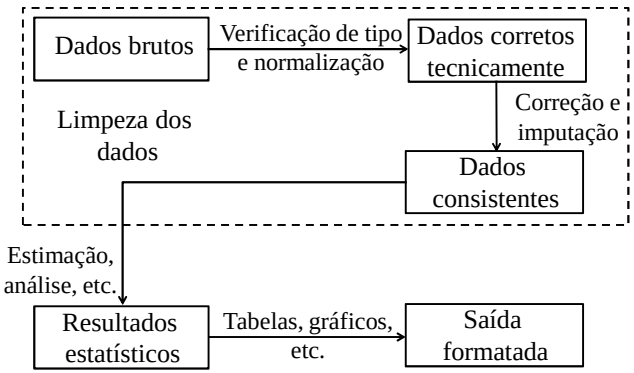
1. Introdução
2. Amostragem
3. Modelos probabilísticos
4. **Análise exploratória de dados**
5. Distribuições amostrais e estimação
6. Testes de significância
7. Comparações de médias
8. Tabelas de contagem
9. Regressão e correlação
10. Referências

Roteiro do Módulo

3. Análise exploratória de dados:
   a) Preparação e limpeza dos dados
   b) Imputação de dados
   c) Análise exploratória de dados
   d) Aplicação
   e) Referências

Preparação e Limpeza dos Dados

Passos da Análise Estatística



√ Dados brutos = dados não tratados

Pesquisa Quantitativa - 2017 7

Dados Brutos

- Pode haver:
  - √ Dados sem cabeçalho
  - √ Tipos de dados errados
    - Ex.: números armazenados como strings
  - √ Categorias com níveis incorretos
  - √ Codificação desconhecida ou inesperada
- Em geral, é necessário algum tipo de pré-processamento antes da importação dos dados

Pesquisa Quantitativa - 2017 8

Dados Tecnicamente Corretos

- Podem ser lidos pelo pacote estatístico, com nomes, tipos e níveis corretos
- Mesmo nessa situação pode haver erros nos dados:
  - √ Variável idade com valores negativos
  - √ Dados faltantes (*missing data*)
  - √ Etc.

Pesquisa Quantitativa - 2017 9

- Essas inconsistências nos dados dependem do assunto em análise
  - √ Inconsistências devem ser eliminadas antes da condução de inferência estatística

Dados Consistentes

- Dados prontos para inferência estatística
  - √ Dados utilizados como ponto de partida para a maioria dos procedimentos estatísticos
  - √ Métodos de limpeza dos dados influenciarão resultados estatísticos
    - Ex.: imputação
  - √ Eles devem ser explicados nas etapas seguintes de análise ou interpretação

- Recomendação:
  - √ Armazenar separadamente os dados de entrada em cada etapa para reutilização
    - (dados brutos, dados tecnicamente corretos, dados consistentes)
  - √ Manter script de execução de cada passo entre os estágios, para reprodutibilidade

Tipos de Variáveis e Técnicas de Indexação

- Numérico: dados quantitativos contínuos
- Inteiro: dados quantitativos discretos
- Fator: dados categóricos com classificações simples
- Ordenado: dados categóricos com classificações ordenadas
- Caractere: *strings*

Uso dos Dados

- Operações aritméticas
- Operadores de comparação
  - √ >, <, =
- Operadores lógicos
  - √ ∩, U, c
- Funções matemáticas básicas
  - √ log, exp, sqrt, etc.

Valores Especiais

- `NA`: dado não disponível (*missing data*)
  - √ Todas as operações básicas devem trabalhar com `NA` sem falhar
  - √ (pode haver resposta NA, se a entrada for NA)
- `NULL`: pode ser visto como conjunto vazio
  - √ Vetor de comprimento zero

- `Inf`: aplicado apenas a vetores de classe numérica
  - √ Tecnicamente `Inf` é um valor válido
  - √ Operações com números são bem definidas e os operadores de comparação funcionam
- `NaN`: not a number
  - √ Em geral, resultado de cálculo com resultado desconhecido
  - √ Cálculos envolvendo números e `NaN` resultam sempre em `NaN`

- Sugestão:
  - √ Verificar em cada vetor a ocorrência de valores especiais ou não numéricos

De Dados Brutos para Dados Tecnicamente Corretos

Conjunto de Dados

- Coleção de dados que descrevem valores de atributos (variáveis) de uma quantidade de objetos (sujeitos, unidades, itens)

Conjunto de Dados Tecnicamente Corretos

- Cada valor do conjunto de dados:
  - √ Pode ser reconhecido diretamente como pertencente a um certa variável
  - √ Está armazenado em um tipo de dados que representa o domínio do valor da variável no mundo real

- Ou seja, em cada item:
  - √ Variável de texto está armazenada como texto
  - √ Variável numérica, como número
  - √ Etc.
- Tudo está em um formato que está consistente ao longo do conjunto de dados

Exemplo

- No R:
  - √ Conjunto de dados está armazenado em `data.frame`, com nomes adequados de colunas
  - √ Cada coluna do `data.frame` é de um tipo que representa adequadamente o domínio da variável da coluna
    - Dados quantitativos armazenados como `numeric` ou `integer`
    - Strings, como `character`
    - Dados categóricos armazenados como `factor` ou `ordered`, com os níveis apropriados

- Boa prática:
  - √ Sempre que você precisar ler dados de um formato específico de arquivo (planilha, arquivo de pacotes estatísticos), faça o software exportar os dados para um formato aberto

Leitura de Dados

- Dados retangulares
  - √ Mesma quantidade de linhas em todas as colunas
    - Últimas células da coluna vazias

Importação de Arquivos de Dados

- Importação de arquivos de formato aberto, no R:
  - √ `read.csv`: valores separados por vírgulas e ponto como separador decimal
  - √ `read.csv2`: valores separados por ponto-e-vírgula e vírgula como separador decimal
  - √ `read.delim`: valores separados por tab e ponto como separador decimal
  - √ `read.delim2`: valores separados por tab e vírgula como separador decimal

√ `read.fwf`: dados com número predeterminado de bytes por coluna

Importante

- Atentar para:
  - √ Se primeira linha do conjuntos de dados é cabeçalho ou não
  - √ Linhas do arquivos de dados usadas para documentar o conjunto de dados
  - √ Variável numérica malformada
    - – Pode levar pacote a interpretar toda a coluna como uma variável de texto

Conversão de Tipo de Variável

- Verifique se a classe de cada cluna do conjunto de dados está de acordo com o tipo da variável
- Fatores:
  - √ Recodificação de fatores
  - √ Ordenação
  - √ Determinação de nível de referência
    - – Necessário para a aplicação de certas técnicas

- Datas:
  - √ Converter datas para o formato adequado de seu pacote e das ferramentas que serão utilizadas

Manipulação de Caracteres

- Compreende:
  - √ Remoção de espaços em branco
  - √ Modificação da largura das *strings*
  - √ Conversão para maiúsculas/minúsculas
  - √ Busca por *substrings* (padrões mais simples)
  - √ Procedimentos de correspondência baseados em intervalos da *string*

```
##   gender
## 1      M
## 2   male
## 3 Female
## 4   fem.
```

30

Normalização da *String*

- Transformação de uma variedade de *strings* para um conjunto menor de valores de *strings* que são processadas mais facilmente

31

- Funcionalidades:
  - √ Encontrar um padrão em uma *string*
  - √ Substituir um padrão por outro

```
library(stringr)
str_trim("  hello world ")
## [1] "hello world"
str_trim("  hello world ", side = "left")
## [1] "hello world "
str_trim("  hello world ", side = "right")
## [1] "  hello world"
```

```
str_pad(112, width = 6, side = "left", pad = 0)
## [1] "000112"
```

```
toupper("Hello world")
## [1] "HELLO WORLD"
tolower("Hello World")
## [1] "hello world"
```

32

Correspondência Aproximada de *Strings*

- Determinar se *substring* ocorre dentro de outra *string*
- Definir uma métrica de distância entre *strings* para medir quão ‘diferentes’ são duas *strings*

```
i <- apply(D, 1, which.min)
data.frame(rawtext = gender, coded = codes[i])
##   rawtext  coded
## 1       M   male
## 2    male   male
## 3  Female female
## 4    fem. female
```

33

Problemas de Codificação de Caracteres

- Considerar que o arquivo de texto está no mesmo esquema definido pela formatação local do sistema operacional

De Dados Tecnicamente Corretos para Dados Consistentes

Dados Consistentes

- Dados tecnicamente corretos que são adequados para análise estatística
  - √ Compreende, remoção, correção ou imputação de dados
    - – *missing values*, valores especiais, erros (óbvios) e outliers

Tipos de Consistência

- Consistência da variável (*in-record consistency*)
  - √ Nenhuma informação contraditória está armazenada em um registro único
- Consistência cruzada:
  - √ Resumos estatísticos de diferentes variáveis não estão em conflito entre si

- Consistência entre conjuntos de dados:
  - √ Conjunto de dados em análise é consistente com outros conjuntos de dados pertinentes com o mesmo assunto

Obtenção da Consistência dos Dados

1. Detecção de inconsistência:
   - √ Estabelecimento de quais restrições foram violadas
     - Ex.: Idade está restrita a valores não negativos
2. Seleção do campo ou campos que causam inconsistência
   - Espera-se a manutenção das relações das variáveis
   - Ex.: uma criança deve ser solteira
   - Se ocorrer violação não está claro qual dos dados está incorreto (ou ambos)

3. Correção dos campos que foram considerados errôneos
   - √ Correção pode ser feita por métodos determinísticos (baseados em modelos) ou estocásticos
- Essas etapas não são necessariamente separadas para muitos métodos de correção
- É útil reconhecer essas etapas para deixar claro quais os pressupostos assumidos durante o processo de limpeza dos dados.

Detecção e Localização de Erros

- Técnicas para detectar violações de restrições univariadas e multivariadas
  - √ Valores faltantes
  - √ Valores especiais
  - √ Outliers
  - √ Inconsistências óbvias
  - √ Localização dos erros

Valores Faltantes

- Espaço reservado para um dado cujo tipo é conhecido, mas não seu valor
  - √ Não é possível realizar análises estatísticas nos dados em que faltam um ou mais valores
  - √ Esses elementos podem ser omitidos do conjunto de dados ou imputado um valor
  - √ A ausência do valor é tratada antes de qualquer análise

- √ Analista deve decidir como tratar os valores vazios
  - Imputação padronizada pode levar a resultados inesperados ou errados e serem difíceis de rastrear
  - É comum preferir omitir os registros com dados faltantes
- √ Se 'desconhecido' for uma categoria, ele deve ser considerado um nível de fator, para uma análise adequada
  - Ex.: Local de nascimento
  - 'desconhecido': não temos conhecimento de onde a pessoa nasceu
  - NA: não há informação para determinar se o local de nascimento é conhecido ou não

- Pode acontecer que um valor faltante significa 0 ou não aplicável
  - √ Aquele valor deveria ser imputado explicitamente
  - √ Valor não é desconhecido, mas foi codificado como vazio

Valores Especiais

- ±Inf, NA, NaN
  - √ Em geral, cálculos envolvendo valores especiais têm como resultado valores especiais
  - √ É desejável manipular valores especiais antes da análise
  - √ Afirmações estatísticas sobre fenômenos do mundo real não devem incluir valores especiais
  - √ Valores especiais em variáveis quantitativas:
    - Não são elementos do conjunto dos números reais

- No R:
  - √ `is.finite`: determina quais valores são ‘regulares’
    - Resposta `FALSE` para `Inf`, `NaN`, `NA`

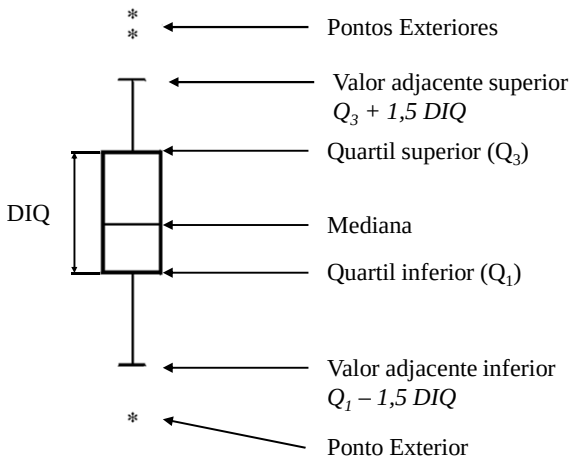
Outliers

- Há várias definições para *outlier*
- Definição geral
  - √ “*Outlier* em um conjunto de dados é uma observação (ou conjunto de observações) que aparenta estar inconsistente”

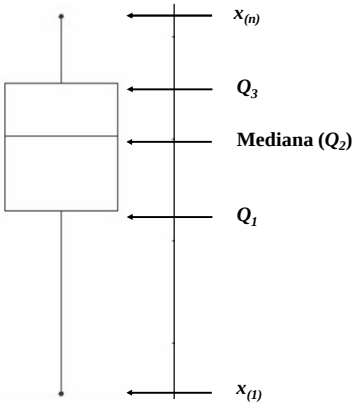
    BARNETT, V.; LEWIS, T. *Outliers in statistical data*. Wiley, 3rd. edition, New York, 1994.

- *Outlier* não é o mesmo que erro:
  - √ Devem ser detectados, mas não necessariamente removidos
  - √ Sua inclusão na análise é um decisão estatística

- Método do Box-plot:
  - √ Frequentemente adequado para dados com distribuição aproximadamente simétrica e unimodal
  - √ Considerado outlier se estiver na região atípica do gráfico



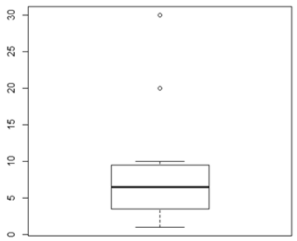
- Se não houver pontos exteriores:



Exemplo

- Box-plot:

```
> x <- c(1:10, 20, 30)
> boxplot(x)
```



- Identificação dos valores *outliers*

```
> # statistics necessary for producing box plots (invoked separately).
> boxplot.stats(x)$out
[1] 20 30
> # factor 2.0 used to compute the whisker
> boxplot.stats(x, coef = 2)$out
[1] 30
```


- O método do box-plot falha quando os dados são distribuídos assimetricamente
  - √ Solução possível:
    - Transformação log ou raiz quadrada dos dados

- Outros métodos para detecção de *outliers*:
  - √ Método de Hidiroglov e Berthelot
    - – Para uso em dados periódicos
  - √ Identificação de outliers com limites baseados em estimativas robustas de locação e escala
  - √ Identificação de outliers com razões centradas com relação à mediana

Inconsistências Óbvias

- Registros que contêm um valor ou combinação de valores que não correspondem à situações reais (regras ou restrições)
  - √ Ex.:
    - – Idade negaiva
    - – Homem grávido

- Regras multivariadas podem crescer rapidamente e pode ser vantajoso gerenciá-las separadamente
  - √ Ex.:

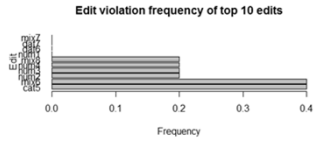
```
people <- read.csv("people.txt")
head(people)
  age agegroup height  status yearsmarried
1  21    adult    6.0  single           -1
2   2    child    3.0 married            0
3  18    adult    5.7 married           20
4 221  elderly    5.0 widowed            2
5  34    child   -7.0 married            3
```


- √ Restrições:

```
> library(editrules)# It allows to define rules on categorical, numerical
> E <- editfile("edits.txt")
> E
Data model:
dat6 : agegroup %in% c('adult', 'child', 'elderly')
dat7 : status %in% c('married', 'single', 'widowed')

Edit set:
num1 : 0 <= age
num2 : 0 < height
num3 : age <= 150
num4 : yearsmarried < age
cat5 : if( agegroup == 'child' ) status != 'married'
mix6 : if( age < yearsmarried + 17 ) !( status %in% c('married', 'widowed') )
mix7 : if( age < 18 ) !( agegroup %in% c('adult', 'elderly') )
mix8 : if( 18 <= age & age < 65 ) !( agegroup %in% c('child', 'elderly') )
mix9 : if( 65 <= age ) !( agegroup %in% c('adult', 'child') )
```

√ Verificação dos dados

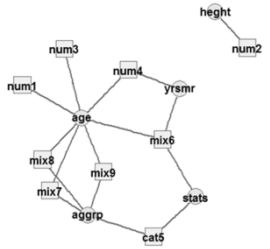
```
> ve <- violatedEdits(E, people)
> #summarizing the result
> summary(ve)
Edit violations, 5 observations, 0 completely missing (0%):

 editname freq rel
     cat5    2 40%
     mix6    2 40%
     num2    1 20%
     num3    1 20%
     num4    1 20%
     mix8    1 20%

Edit violations per record:

 errors freq rel
      0    1 20%
      1    1 20%
      2    2 40%
      3    1 20%
```



59

- Visualização das regras e de seus relacionamentos com as variáveis

```
plot(E)
```



60

Localização dos Erros

- Minimizar a quantidade de campos sendo alterados
  - √ (Princípio de Fellegi e Holt)
  - √ Ideia:
    - Erros ocorrem, relativamente poucas vezes e, quando acontecem, eles ocorrem aleatoriamente entre as variáveis
  - √ Pode ser usado em procedimento de correção e imputação

61

- No exemplo:

```
> id <- c(2, 5)
> people[id, ]
  age agegroup height  status yearsmarried
2   2    child      3 married            0
5  34    child     -7 married            3
> # finding the minimal set of variables to adapt.
> le <- localizeErrors(E, people[id, ], method = "mip")
> le$adapt
    age agegroup height status yearsmarried
2 FALSE    FALSE  FALSE   TRUE        FALSE
5 FALSE     TRUE   TRUE  FALSE        FALSE
> # repairing records to full compliance with all edit rules.
> people[2, "status"] <- "single"
> people[5, "height"] <- 7
> people[5, "agegroup"] <- "adult"
```

62

Métodos de Correção

- Visam corrigir observações inconsistentes, alterando valores inválidos em um registro, com base em informações de valores válidos

Regras de Transformação Simples

- Em geral, os procedimentos de limpeza envolvem muitas transformações *ad-hoc*.

Exemplo

- Conjunto de dados:

```
> (marx <- read.csv("marx.csv", stringsAsFactors = FALSE))
   name height unit
1 Groucho 170.00   cm
2  Zeppo   1.74    m
3  Chico  70.00 inch
4  Gummo 168.00   cm
5  Harpo   5.91   ft
```

- Regras de correção:

```
> R <- correctionRules("conversions.txt")
> R
Object of class 'correctionRules'
##  1-------
  if (unit == "cm") {height <- height/100}
##  2-------
  if (unit == "inch") {height <- height/39.37}
##  3-------
  if (unit == "ft") {height <- height/3.28}
##  4-------
  unit <- "m"
```


- Correções efetuadas

```
> cor <- correctWithRules(R, marx)
> # list containing the corrected data
> cor$corrected
     name   height unit
1 Groucho 1.700000    m
2  Zeppo  1.740000    m
3  Chico  1.778004    m
4  Gummo  1.680000    m
5  Harpo  1.801829    m
> cor$corrections[1:4]
  row variable  old      new
1   1   height  170      1.7
2   1     unit   cm        m
3   2     unit    m        m
4   3   height   70 1.778004
5   3     unit inch        m
6   4   height  168     1.68
7   4     unit   cm        m
8   5   height 5.91 1.801829
9   5     unit   ft        m
```

Correção Dedutiva

- Em dados produzidos por pessoas podem ocorrer certos erros típicos
  - √ Erros de digitação de números
  - √ Erros de arredondamento
  - √ Erros de sinal
  - √ Trocas de variáveis

Exemplo

- Erros de medidas em variáveis:

```
> e <- editmatrix("x + y == z")
> d <- data.frame(x = 100, y = 101, z = 200)
> # correcting deviations of the size of one or two measurement units.
> cor <- correctRounding(e, d)
> cor$corrected
   x   y   z
1 99 101 200
> cor$corrections
  row variable old new
1   1        x 100  99
```


- Erros de sinal:

```
> # detecting and repairing sign errors
> e <- editmatrix("x + y == z")
> d <- data.frame(x = 100, y = -100, z = 200)
> cor <- correctSigns(e, d)
> cor$corrected
    x   y   z
1 100 100 200
> cor$corrections
  row variable  old new
1   1        y -100 100
```

- Erros de digitação

```
> # detecting and correcting typographic errors in numbers.
> e <- editmatrix("x + y == z")
> d <- data.frame(x = 123, y = 132, z = 246)
> cor <- correctTypos(e, d)
> cor$corrected
    x   y   z
1 123 123 246
> cor$corrections
  row variable old new
1   1        y 132 123
```


Imputação Determinística

- Em alguns casos, o valor faltante pode ser determinado porque os valores observados combinados forçam uma solução única

Exemplo – Dados Quantitativos

- Pessoal + Limpeza + Habitação = Total

```
> E <- editmatrix(expression(staff + cleaning + housing == total, staff >= 0,
+ housing >= 0, cleaning >= 0))
> dat <- data.frame(staff = c(100,100,100), housing = c(NA,50,NA),
+ cleaning = c(NA,NA,NA), total = c(100,180,NA))
> dat
  staff housing cleaning total
1   100      NA       NA   100
2   100      50       NA   180
3   100      NA       NA    NA
> cor$corrections
  row variable old new
1   1        x 100  99
```


√ Imputa casos com solução única

```
> # recognize cases where unique solutions exist and
> # compute the unique imputations
> cor <- deduImpute(E,dat)
> cor$corrected
  staff housing cleaning total
1   100       0        0   100
2   100      50       30   180
3   100      NA       NA    NA
```


Exemplo – Dados Categóricos

- Pessoal + Limpeza + Habitação = Total

```
> E <- editarray(expression(age %in% c("adult","under-aged"),
+ driverslicense %in% c(TRUE, FALSE),
+ if ( age == "under-aged" ) !driverslicense))
> dat <- data.frame(age = NA, driverslicense = TRUE)
> dat
  age driverslicense
1  NA           TRUE
```

√ Imputação de dado com solução única

```
> # recognize cases with unique solutions and compute the unique imputations
> cor <- deduImpute(E,dat)
> cor$corrected
    age driverslicense
1 adult           TRUE
```


Imputação de Dados

Imputação de Dados

- Processo para estimar ou derivar valores de campos em que o dado é faltante
- Não há um método ótimo de imputação que funciona em todos os casos

- Funcionalidades de imputação no R:

Package	Numeric			Hot deck			kNN	Longitudinal		
	mean	ratio	reg.	rand	seq	pmm		int	lo/no	LS
Amelia[37]	·	·	·	·	·	·	·	·	·	·
BaBoon[20]	·	·	·	·	·	✓	·	·	·	·
cat[27]	·	·	·	✓	·	·	·	·	·	·
deducorrect[33]	·	·	·	·	·	·	·	·	·	·
e1071[21]	✓	·	·	·	·	·	·	·	·	·
ForImp	✓	·	·	·	·	·	✓‡	·	·	·
Hmisc[18]	✓	·	·	✓	·	✓	·	·	·	·
imputation[37]	✓	·	✓†	·	·	·	✓	·	·	·
impute[14]	·	·	·	·	·	·	✓	·	·	·
mi[23]	·	·	✓*	✓	·	✓	·	·	·	·
mice[30]	✓	·	✓*	✓	·	✓	·	·	·	·
mix[3]	·	·	·	·	·	·	·	·	·	·
norm[25]	·	·	·	·	·	·	·	·	·	·
robCompositions[25]	·	·	✓*	✓	·	·	✓	·	·	·
rrcovNA[28]	·	·	·	·	·	·	·	·	·	·
StatMatch[8]	·	·	·	✓	·	·	✓	·	·	·
VIM[24]	·	·	✓*	✓	·	·	✓	·	·	·
yaImpute[5]	·	·	·	·	·	·	✓	·	·	·
zoo[39]	✓	·	·	·	·	·	·	✓	✓	·

*Methods are ultimately based on some form of regression, but are more involved than simple linear regression.
†Uses a non-informative auxiliary variable (row number).
‡Uses nearest neighbor as part of a more involved imputation scheme.

Análise Exploratória de Dados

O que é Análise Exploratória de Dados?

- Filosofia/abordagem para análise de dados
- Emprega principalmente técnicas de visualização gráfica:
  - √ Diagramas de dispersão
  - √ Boxplot
  - √ Gráficos para identificação de outliers
  - √ Curvas de crescimento
  - √ Etc.

- As técnicas buscam:
  - √ maximizar o “insight” do conjunto de dados;
  - √ perceber a estrutura subjacente;
  - √ extrair variáveis importantes;
  - √ detectar valores atípicos (*outliers*), anomalias conglomerados
  - √ testar hipóteses fundamentais;
  - √ desenvolver modelos parcimoniosos
  - √ determinar conjunto ótimo de fatores

80

Idéia Básica

- Modelo = Suave + Irregular (tosco)
  - √ Frequentemente, as técnicas gráficas podem separar o “suave” do “irregular” (“ruído”)

81

Clássica vs Exploratória

- Seqüência Clássica:
  - √ Problema > Dados > Modelo> Análise > Conclusões
- Exploratória:
  - √ Problema > Dados > Análise > Modelo > Conclusões

82

Tratamento de Dados

- Clássica:
  - √ Média e desvio padrão = estimativas pontuais
  - √ Medida de variabilidade explicada – r de Pearson

83

- Exploratória
  - √ Resumo Numérico (5): Min, Q1, Median, Q3, Max
  - √ todos (maioria) dados=resumos visuais
  - √ Dispersão
  - √ Histograma
  - √ boxplot

Análise Descritiva

- Verificação dos tipos de variáveis disponíveis
  - √ Variáveis podem ser resumidas por:
    - Gráficos
    - Medidas
    - Tabelas

Classificação das Variáveis

- Qualitativas (ou Categóricas)
  - √ Nominais:
  - √ Ordinais
- Quantitativas:
  - √ Discretas
  - √ Contínuas

Objetivos

- Familiarização com os dados
- Detecção de estruturas interessantes
- Presença de valores atípicos (*outliers*)

Razões para Uso de AED

- √ Identificação de erros e inconsistências
- √ Verificação de pressupostos do modelo
- √ Seleção preliminar de modelos apropriados
- √ Determinação das relações entre as variáveis explicativas
- √ Avaliação da direção e da dimensão das relações entre as variáveis explicativas e as variáveis respostas.

Análise Multivariada

- Para um conjunto de variáveis correlacionadas:
  - √ Avaliar as relações entre as variáveis
  - √ Considerar os efeitos dos "tratamentos" sobre essas relações
  - √ Considerar como uma "resposta" depende dessas relações

- Métodos multivariados para redução de dados:
  - √ Resumir as correlações entre variáveis
  - √ Produzir um conjunto menor de variáveis (não correlacionadas) contendo as informações mais importantes
- Para um conjunto de objetos "relacionados"
  - √ Identificar grupos de objetos semelhantes
  - √ Identificar diferenças entre grupos de objetos semelhantes
    - (e o que faz com que os objetos sejam semelhantes)

Análise Exploratória de Dados Multivariados

- Sequência básica inicial:
  - √ Medidas-resumo e gráficos:
    - Variabilidade para cada variável
    - Forma da distribuição de cada variável
  - √ Grupos de observações:
    - Pré-determinados
    - (para encontrar diferenças potenciais)
  - √ Diagrama de dispersão/correlações
    - Associações entre pares de variáveis

- Recomenda-se executar análise exploratória de dados univariados em cada um dos componentes, antes de realizar a AED multivariada.

Importante

- A Análise Exploratória de Dados é um passo inicial crítico em qualquer análise de dados.

Aplicação

Exemplo

- Current Population Survey, Maio/85.
  - √ Amostra aleatória com 534 observações
  - √ 11 variáveis (quantitativas e categóricas)
  - √ Dados: *CPS1985*{*AER*} ou *CPS1985.csv*

√ Variáveis:

- wage: salário, em US$ por hora
- education: anos de escolaridade
- experience: anos de experiência profissional potencial (age – education – 6).
- age: idade, em anos
- ethnicity: etnia. (cauc, hispanic, other)
- region: mora no Sul? (south, other)
- gender: sexo. (male, female).
- occupation: ocupação. (worker, technical, services, office, sales, management.
- sector: setor de ocupação. (manufacturing, construction, other).
- union: trabalho sindicalizado? (no, yes).
- married: É casado? (no, yes).

- Importação dos dados – pacote AER:

```
> # carregamento direto do pacote
> data("CPS1985", package = "AER")
> cps <- CPS1985
> head(cps)
 wage education experience age ethnicity region gender occupation
1     5.10         8         21  35  hispanic  other female     worker
1100  4.95         9         42  57      cauc  other female     worker
2     6.67        12          1  19      cauc  other   male     worker
           sector union married
1    manufacturing    no     yes
1100 manufacturing    no     yes
2    manufacturing    no      no
> str(CPS1985)
'data.frame':   534 obs. of  11 variables:
 $ wage      : num  5.1 4.95 6.67 4 7.5 ...
 $ education : num  8 9 12 12 12 13 10 12 16 12 ...
 $ experience: num  21 42 1 4 17 9 27 9 11 9 ...
 $ age       : num  35 57 19 22 35 28 43 27 33 27 ...
 $ ethnicity : Factor w/ 3 levels "cauc","hispanic",..: 2 1 1 1 1 1 1 1 1 1 ...
 $ region    : Factor w/ 2 levels "south","other": 2 2 2 2 2 2 1 2 2 2 ...
 $ gender    : Factor w/ 2 levels "male","female": 2 2 1 1 1 1 1 1 1 1 ...
 $ occupation: Factor w/ 6 levels "worker","technical",..: 1 1 1 1 1 1 1 1 1 1 ...
 $ sector    : Factor w/ 3 levels "manufacturing",..: 1 1 1 3 3 3 3 3 1 3 ...
 $ union     : Factor w/ 2 levels "no","yes": 1 1 1 1 1 2 1 1 1 1 ...
 $ married   : Factor w/ 2 levels "no","yes": 2 2 1 1 2 1 1 1 2 1 ...
```


- Importação pelo arquivo CPS1985.csv:

```
> # carregamento do arquivo csv
cps <- read.csv("CPS1985.csv")
> head(cps)
 wage education experience age ethnicity region gender occupation
1     5.10         8         21  35  hispanic  other female     worker
1100  4.95         9         42  57      cauc  other female     worker
2     6.67        12          1  19      cauc  other   male     worker
           sector union married
1    manufacturing    no     yes
1100 manufacturing    no     yes
2    manufacturing    no      no
> str(CPS1985)
'data.frame':   534 obs. of  11 variables:
 $ wage      : num  5.1 4.95 6.67 4 7.5 ...
 $ education : num  8 9 12 12 12 13 10 12 16 12 ...
 $ experience: num  21 42 1 4 17 9 27 9 11 9 ...
 $ age       : num  35 57 19 22 35 28 43 27 33 27 ...
 $ ethnicity : Factor w/ 3 levels "cauc","hispanic",..: 2 1 1 1 1 1 1 1 1 1 ...
 $ region    : Factor w/ 2 levels "south","other": 2 2 2 2 2 2 1 2 2 2 ...
 $ gender    : Factor w/ 2 levels "male","female": 2 2 1 1 1 1 1 1 1 1 ...
 $ occupation: Factor w/ 6 levels "worker","technical",..: 1 1 1 1 1 1 1 1 1 1 ...
 $ sector    : Factor w/ 3 levels "manufacturing",..: 1 1 1 3 3 3 3 3 1 3 ...
 $ union     : Factor w/ 2 levels "no","yes": 1 1 1 1 1 2 1 1 1 1 ...
 $ married   : Factor w/ 2 levels "no","yes": 2 2 1 1 2 1 1 1 2 1 ...
```


- Abreviação níveis do fator occupation

√ Dados carregados do pacote

```
> levels(cps$occupation)
[1] "worker"     "technical"  "services"   "office"     "sales"
[6] "management"
> levels(cps$occupation)[c(2, 6)] <- c("mgmt", "techn")
> levels(cps$occupation)
[1] "worker"   "techn"    "services" "office"   "sales"    "mgmt“
```

√ Dados carregados do arquivo csv

```
> levels(cps$occupation)
[1] "management" "office"     "sales"      "services"   "technical"
[6] "worker"
> levels(cps$occupation)[c(1, 5)] <- c("mgmt", "techn")
> levels(cps$occupation)
[1] "mgmt"     "office"   "sales"    "services" "techn"    "worker“
```

√ Anexando conjunto de dados para trabalho:

```
> # anexando o arquivo
> attach(cps)
```

- Distribuição de `wage` na amostra:

√ Resumo dos 5 números e média

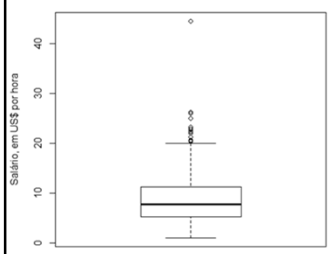
```
> summary(wage)
   Min. 1st Qu.  Median    Mean 3rd Qu.    Max.
  1.000   5.250   7.780   9.024  11.250  44.500
> mean(wage)
[1] 9.024064
> median(wage)
[1] 7.78
```

√ Outros comandos para quantis

```
> fivenum(wage)
[1]  1.00  5.25  7.78 11.25 44.50
> quantile(wage)
   0%   25%   50%   75%  100%
 1.00  5.25  7.78 11.25 44.50
> quantile(wage, probs = c(0.05, 0.95))
   5%   95%
 3.50 19.98
> max(wage); min(wage)
[1] 44.5
[1] 1
```


- Box-plot da variável `wage`:

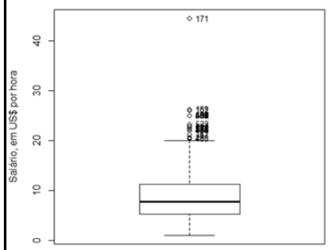
```
> boxplot(wage, ylab = "Salário, em US$ por hora")
```



√ Gráfico dos 5 números
√ Há *outliers*

- Identificação de *outliers* em `wage`:

```
> # identificação outliers
> wage.bxp <- boxplot(wage, ylab = "Salário, em US$ por hora", plot = F)
> outliers.row <- which(wage %in% wage.bxp$out)
> for(i in 1:length(wage.bxp$group)){
+     # adiciona texto no boxplot
+     text(wage.bxp$group[i], wage.bxp$out[i], which(wage==wage.bxp$out[i]),
+ pos = 4, cex = 0.8)
+ }
```



√ Há empates

- Detalhamento dos outliers

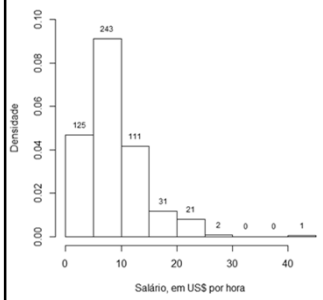
```
> outliers.row
 [1]  18  20 107 157 162 169 171 178 181 185 211 410 432 434 436 450 480 485 486
[20] 495 497 503 522 532
> length(outliers.row)
[1] 24
> cps[outliers.row,]
 wage education experience age ethnicity region gender occupation
17  22.20        12         26  44      cauc  other   male     worker
19  20.55        12         33  51      cauc  other   male     worker
106 26.00        14         21  41      cauc  other   male     worker
156 24.98        16         18  40      cauc  other   male       mgmt
161 21.25        13         32  51      cauc  other   male       mgmt
sector union married
17  manufacturing   yes     yes
19          other    no     yes
106         other   yes     yes
156         other    no     yes
161         other    no      no
```

- Medidas de dispersão de `wage`:
  - √ Desvio-padrão, variância e distância interquartílica

```
> sd(wage)
 [1] 5.139097
> var(wage)
[1] 26.41032
> IQR(wage)
[1] 6
```


- Histograma da variável `wage`:

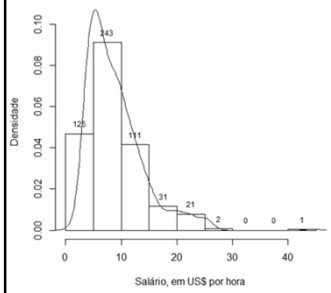
```
> wage.hst <- hist(wage, freq = FALSE, main = "", ylab = "Densidade",
+ xlab = "Salário, em US$ por hora", ylim = c(0, 0.105))
> text(wage.hst$mid, wage.hst$density + 0.005, wage.hst$counts, cex = 0.75)
```



- √ Fortemente assimétrica
- √ Apenas um *outlier*?

- Histograma com suavização:

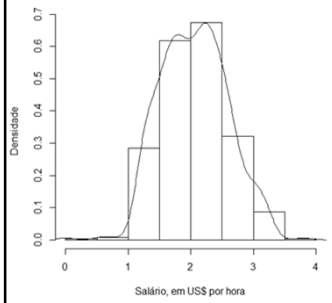
```
> # suavização por núcleo estimador
> lines(density(wage), col = 4)
```



- √ Indício de bimodalidade?
  - − poucos dados para afirmar

- Histograma de `log(wage)`:

```
> hist(log(wage), freq = FALSE, main = "", ylab = "Densidade",
+ xlab = "Salário, em US$ por hora")
> lines(density(log(wage)), col = 4)
```



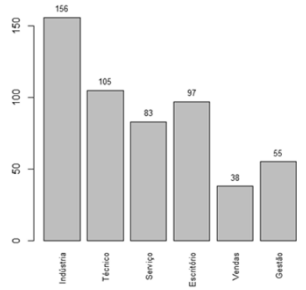
- √ Distribuição de `log(wage)` é menos assimétrica

- Resumo da variável `occupation`:
  - √ Tabelas de frequências absolutas e relativas

```
> summary(occupation)
  worker    techn services   office    sales     mgmt
     156      105       83       97       38       55
> tab <- table(occupation)
> prop.table(tab)
occupation
    worker      techn   services     office      sales       mgmt
0.29213483 0.19662921 0.15543071 0.18164794 0.07116105 0.10299625
```


- *Barplot* da variável `occupation`:

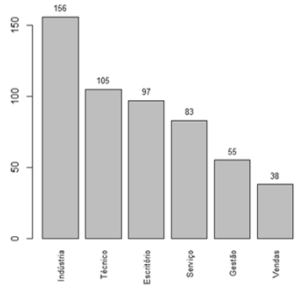
```
> nomes <- c("Indústria", "Técnico", "Serviço", "Escritório", "Vendas",
+ "Gestão")
> occup.bp <- barplot(tab, names.arg = nomes, cex.names = 0.85, las = 3,
+     ylim = c(0, 165))
> text(occup.bp, tab, labels = tab, cex = 0.85, pos = 3, offset = 0.5)
```



√ Ficaria melhor se as barras estivessem ordenadas por frequência?

- *Barplot* com as barras ordenadas:

```
> ordem <- order(tab, decreasing = T)
> occup.ord <- barplot(tab[ordem], names.arg = nomes[ordem], cex.names = 0.85,
+     las = 3, ylim = c(0, 165))
> text(occup.ord, tab[ordem], labels = tab[ordem], cex = 0.85, pos = 3,
+ offset = 0.5)
```



√ Pode ser conveniente quando houver muitos níveis

- *Pie chart* da variável `occupation`:

```
> pie(tab)
```

√ Eventualmente, podem ser úteis.

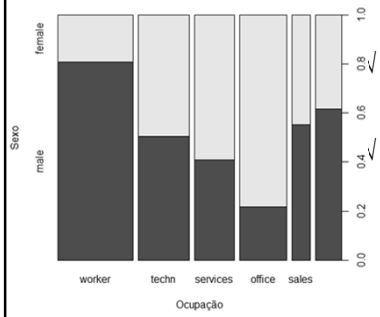
- Análise bivaraida – categóricas
  (Variáveis: `gender` e `occupation`)
  √ Tabela de dupla entrada

```
> xtabs(~ gender + occupation, data = cps))
        occupation
gender   worker techn services office sales mgmt
  male      126    53       34     21    21   34
  female     30    52       49     76    17   21
```

116

- *Barplot* com duas variáveis categóricas:

```
> plot(gender ~ occupation, xlab = "Ocupação", ylab = "Sexo")
```



√ Proporção de homens e mulheres variam consideravelmente com a ocupação
√ Há mais pessoas trabalhando em "workers" que em "sales"

117

- Análise bivariada – quantitativas
  (Variáveis: `log(wage)` e `education`)
  √ Correlação de Pearson

```
> cor(wage, education)
[1] 0.3819221
> cor(log(wage), education)
[1] 0.3803983
```

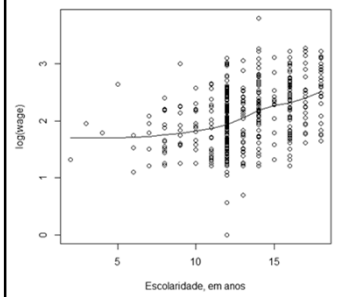
√ Correlação de Spearman – Não Paramétrico

```
> cor(wage, education, method = "spearman")
[1] 0.3813425
> cor(log(wage), education, method = "spearman")
[1] 0.3813425
```

118

- *Plot* para duas variáveis quantitativas:

```
> plot(log(wage) ~ education, xlab = "Escolaridade, em anos")
```

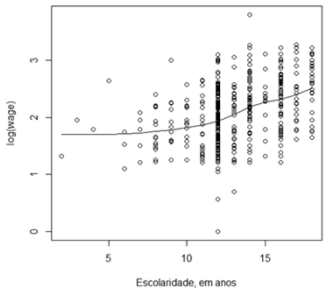


√ Difícil perceber tendência
  − Correlação linear baixa

119

- *Plot* com suavizador:

```
> plot(log(wage) ~ education, xlab = "Escolaridade, em anos")
> wage.lo <- loess(log(wage) ~ education)
> lines(sort(of.lo$x),sort(of.lo$fit))
```



√ Correlação baixa entre as variáveis

- Bivariada – quantitativa *vs*. categórica:
  (Variáveis: `log(wage)` e `gender`)
  √ Estatísticas descritivas por estrato

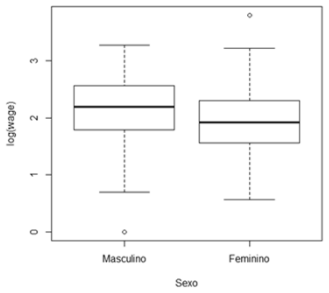
```
> # média por estrato
> tapply(log(wage), gender, mean)
    male   female
2.165286 1.934037
> # média e desvio padrão por estrato
> aggregate(log(wage) ~ gender, FUN = function(x) c(M=mean(x), SD=sd(x)))
gender log(wage).M log(wage).SD
1   male    2.165286     0.534453
2 female    1.934037     0.492118
```

√ Divisão da variável `wage` por `gender`:

```
> cor(wage, education, method = "spearman")
[1] 0.3813425
> cor(log(wage), education, method = "spearman")
[1] 0.3813425
```


- *Boxlot* `wage` por `gender`:

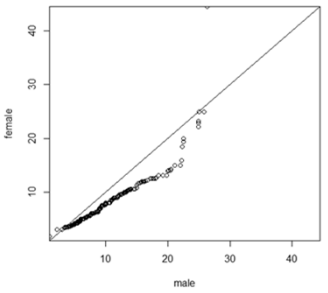
```
> plot(log(wage) ~ gender, xlab = "Sexo", xaxt = "n")
> axis(1, at = 1:2, labels = c("Masculino", "Feminino")
```



√ Similares as formas gerais de ambas as distribuições
√ Homens levam vantagem, principalmente na ‘faixa média’

- *qq-plot* de `wage` por `gender`:

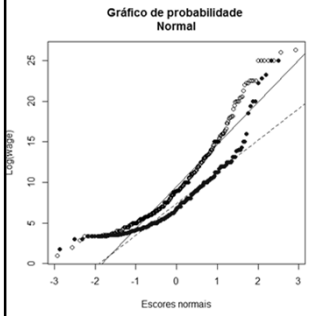
```
> plot(log(wage) ~ gender, xlab = "Sexo", xaxt = "n")
> axis(1, at = 1:2, labels = c("Masculino", "Feminino")
```



√ Na maioria dos quantis, dalário dos homens é tipicamente mais alto

- Gráfico de probabilidade normal:

```
> qqnorm(mwage, ylab = "Log(wage)", xlab = "Escores normais",
+  main="Gráfico de probabilidade \nNormal")
> qqline(mwage)
> norm.fem <- qqnorm(fwage, plot.it = F)
> points(norm.fem$x, norm.fem$y, pch = 21, col = "blue", bg = "blue")
> qqline(fwage, lty = 2, col = "blue")
```



√ Distribuições similares com valores menores de quantis para Mulheres

124

Referências

Bibliografia Recomendada

- AGRESTI, A.; FINLAY, B. *Métodos estatísticos para as ciências sociais.* Penso, 2012.
- JONGE, E.; DER LOO, M. *An introduction to data cleaning with R.* Statistics Netherlands, 2013
- MOORE, D. S.; MCCABE, G. P.; DUCKWORTH, W. M.; SLOVE, S. L. *A prática da estatística empresarial: como usar dados para tomar decisões*. LTC, 2006.

126