ME111 - Atividade 4
Profa. Tatiana Benaglia - 27/04/2020 - 1S2020
Introdução à Análise de Dados

Algumas pessoas definem a Estatística como a ciência que tem por objetivo transformar informação em conhecimento. O primeiro passo no processo é sumarizar e descrever a informação bruta - os dados. Neste laboratório, você obterá novos conhecimento sobre saúde pública gerando sumários gráficos e numéricos de um conjunto de dados coletados pelo Centro para o Controle e Prevenção de Doenças ("Centers for Disease Control and Prevention", CDC). Como esse conjunto de dados é grande, ao longo do caminho você também aprenderá as habilidades indispensáveis de processamento de dados e organização de subconjuntos.

Introdução

O Sistema de Monitoramento de Fatores de Risco Comportamental ("Behavioral Risk Factor Surveillance System", BRFSS) é um *survey* anual realizado por telefone com 350.000 pessoas nos Estados Unidos. Como seu nome implica, o BRFSS foi desenvolvido para identificar fatores de risco na população adulta e relatar tendências emergentes na saúde. Por exemplo, os respondentes são indagados sobre sua dieta e atividades físicas semanais, seu diagnóstico de HIV/AIDS, uso provável de tabaco, e mesmo seu nível de cobertura por planos de saúde. O *website* do BRFSS (*http://www.cdc.gov/brfss*) contém uma descrição completa desta pesquisa, incluindo as questões de pesquisa que motivaram o estudo e muitos resultados interessantes derivados dos dados.

Nós nos focaremos numa amostra aleatória de 20.000 pessoas do BRFSS conduzido no ano de 2000. Ainda que existam mais de 200 variáveis neste conjunto de dados, nós trabalharemos com um subconjunto menor.

Primeiramente, iremos importar os dados das 20.000 observações para o R. Depois de inicializar o RStudio, exectute o seguinte comando:

```
> source("http://www.openintro.org/stat/data/cdc.R")
```

O conjunto de dados `cdc` que aparece em seu espaço de trabalho é um *data frame*, com cada linha representando um *caso* e cada coluna representando uma *variável*.

Para visualizar o nome das variáveis, digite o comando:

```
> names(cdc)
```

Este comando retorna os nomes `genhlth`, `exerany`, `hlthplan`, `smoke100`, `height`, `weight`, `wtdesire`, `age`, e `gender`. Cada uma dessas variáveis corresponde a uma questão que foi feita na pesquisa. Por exemplo, para `genhlth`, os respondentes foram indagados sobre sua saúde geral, respondendo excelente, muito bom, bom, razoável ou ruim. A variável `exerany` indica se o respondente se exercitou no último mês (1) ou não (0). Da mesma forma, `hlthplan` indica se o respondente tem alguma forma de cobertura (1) ou não (0). A variável `smoke100` indica se o respondente fumou pelo menos 100 cigarros ao longo da vida. As outras variáveis registram a altura (`height`) em polegadas, o peso (`weight`) em libras, bem como o peso desejado (`wtdesire`), idade (`age`) em anos, e gênero (`gender`).

> **Exercício 1** Há quantos casos neste conjunto de dados? Quantas variáveis? Para cada variável, identifique seu tipo de dado (por exemplo: categórica, contínua).

Nós podemos dar uma olhada nas primeiras entradas (linhas) de nossos dados com o comando

```
> head(cdc)
```

e, similarmente, podemos verificar as últimas digitando

```
> tail(cdc)
```

Você também pode verificar *toda* a base de dados de uma vez só digitando seu nome no console, mas isso pode não ser muito sábio neste contexto. Sabemos que `cdc` tem 20.000 linhas, portanto verificar o conjunto de dados inteiro significa inundar sua tela. É melhor dar pequenas espiadas nos dados utilizando `head`, `tail`, ou as técnicas de construção de subconjunto que você aprenderá logo em seguida.

Sumários e Tabelas

O questionário do BRFSS é um tesouro enorme de informações. Um primeiro passo útil em qualquer análise é destilar toda essa informação em algumas estatísticas sumárias e gráficos. Como um exemplo simples, a função `summary` retorna um sumário numérico: mínimo, primeiro quartil (Q1), mediana, média, segundo quartil (Q3), e máximo. Para a variável `weight`, esse sumário é:

```
> summary(cdc$weight)
```

> **Exercício 2** O que acontece quando você utiliza a função `summary` no conjunto de dados todo, isto é, sem especificar alguma variável? Experimente.

O R também funciona como uma calculadora poderosa. Se vocês quisesse calcular o intervalo interquartil para o peso dos respondentes, você pode se basear na saída do comando acima e então digitar

```
> 190 - 140
```

O R também tem funções embutidas para calcular estatísticas descritivas uma por uma. Por exemplo, para calcular a média, mediana, e variância da variável `weight`, digite

```
> mean(cdc$weight)
> var(cdc$weight)
> median(cdc$weight)
```

Ainda que faça sentido descrever uma variável quantitativa como `weight` em termos destas estatísticas, o que fazer com dados categóricos? Nós podemos considerar a frequência da amostra ou a distribuição relativa de frequência. A função `table` faz isso por você contando o número de vezes que cada tipo de resposta é dada. Por exemplo, para ver o número de pessoas que fumaram 100 cigarros ao longo de sua vida, digite

```
> table(cdc$smoke100)
```

Ou então verifique a distribuição de frequência relativa digitando

```
> table(cdc$smoke100)/20000
```

Perceba como o R automaticamente divide todas as entradas na tabela por 20.000 no comando acima. Isso é similar a algo que observamos no último laboratório; quando multiplicamos ou dividimos um vetor por um número, o R aplica essa ação a todas as entradas dos vetores. Como vimos acima, isso também funciona para tabelas. Em seguida, criamos um gráfico de barras para as entradas na tabela inserindo a tabela dentro do comando para gráficos de barra.

```
> barplot(table(cdc$smoke100))
```

Preste atenção no que fizemos agora! Nós computamos a tabela da variável `cdc$smoke100` e então imediatamente aplicamos a função gráfica, `barplot`. Esta é uma ideia importante: os comandos do R podem ser aninhados. Você também pode dividir esse procedimento em dois passos digitando o seguinte:

```
> smoke <- table(cdc$smoke100)
> barplot(smoke)
```

Agora, criamos um novo objeto, uma tabela, denominada `smoke` (seu conteúdo pode ser verificado digitando `smoke` no console) e então a utilizamos como entrada para o comando `barplot`. O símbolo especial `<-` realiza uma *atribuição*, tomando a saída de uma linha de código e salvando-a em um objeto no seu espaço de trabalho. Esta é outra ideia importante para a qual retornaremos mais tarde.

Exercício 3 Crie um sumário numérico para `height` (altura) e `age` (idade), e calcule o intervalo interquartílico para cada um. Calcule a distribuição de frequência relativa para `gender` e `exerany`. Quantos homens compõem a amostra? Qual proporção da amostra diz estar com saúde excelente?

O comando `table` pode ser utilizado para tabular qualquer número de variáveis que você quiser. Por exemplo, para examinar quais participantes fumam, dividido por gênero, nós podemos utilizar o seguinte código.

```
> table(cdc$gender, cdc$smoke100)
```

Aqui, vemos colunas formadas por 0 e 1. Lembre-se que o 1 indica que o respondente fumou pelo menos 100 cigarros. As linhas se referem ao gênero. Para criar um gráfico de mosaico para essa tabela, entramos com o seguinte comando.

```
> mosaicplot(table(cdc$gender, cdc$smoke100))
```

Nós poderíamos ter conseguido esse resultado em duas etapas: salvando a tabela em uma linha e aplicando `mosaicplot` em seguida (veja o exemplo de tabela/gráfico de barras acima).

Exercício 4 O que o gráfico de mosaico revela sobre os hábitos de fumar e gênero?

Um outra opção é usar gráfico de barras. Primeiro, podemos olhar, só entre as mulheres, a proporção de fumantes e não fumantes. Depois, repetir o mesmo procedimento para os homens. O ideal é que se apresente homens e mulheres na mesma figura, para ficar mais fácil de comparar os padrões de fumantes e não fumantes em cada gênero.

Primeiro, vamos salvar a tabela de frequência na variável a seguir:

```
> tabela <- table(cdc$smoke100, cdc$gender, dnn=c("Smoke", "Gênero"))
```

Iremos usar a função `prop.table`. O primeiro argumento da função é a tabela de frequências, o segundo argumento indica qual marginal da tabela queremos usar para calcular a proporção. Por exemplo:

```
> prop.table(tabela, 1)
```

Usando o segundo argumento como 1, iremos calcular entre os não fumantes, as proporção de homens e mulheres (na primeira linha). E na segunda linha da tabela, temos as proporções de homens e mulheres entre os fumantes. Repare que as proporções de cada linha somam 1.

Podemos também fazer diferente, fixar só entre mulheres e observar a proporção de fumantes e não fumantes. Fazer o mesmo só entre os homens. Para isso, usamos a opção 2 no segundo argumento da função.

```
> prop.table(tabela, 2)
```

Repare que agora as proporções das colunas somam 1.

Portanto, o segundo argumento da função `prop.table` indica qual total marginal, se de linhas (opção 1) ou colunas (opção 2) devemos dividir as frequências para obter a proporção desejada.

Agora, vamos fazer um gráfico de barras da proporção de fumantes e não fumantes separada por gênero.

```
> barplot(prop.table(tabela, 2), xlab="Gênero", main=" ", beside=TRUE,
+         legend.text=TRUE, ylim=c(0, 0.8), col=c("lightpink", "lightgreen"),
+         cex.axis=1.5, cex.lab=1.5)
```

Trabalhando com Data Frames no R

Mencionamos que o R armazena os dados em bases de dados, que você pode pensar como um tipo de planilha. Cada linha é uma observação diferente (um respondente diferente) e cada coluna é uma variável diferente (a primeira é `genhlth`, a segunda é `exerany` e assim por diante). Nós podemos visualizar o tamanho da base de dados ao lado do nome do objeto na área de trabalho ou podemos digitar

```
> dim(cdc)
```

o que faz retornar o número de linhas e colunas. Agora, se quisermos acessar um subconjunto da base de dados completa, nós podemos utilizar a notação de linhas-e-colunas. Por exemplo, para visualizar a sexta variável do 567º respondente, utilize o comando

```
> cdc[567,6]
```

que significa que nós queremos o elemento de nosso conjunto de dados que está na 567ª linha (ou seja, a 567ª pessoa ou observação) e na 6ª coluna (nesse caso, o peso). Sabemos que `weight` (peso) é a 6ª variável porque ela é a 6ª entrada na lista de nomes de variáveis.

```
> names(cdc)
```

Para visualizar os pesos para os primeiros 10 respondentes, podemos digitar

```
> cdc[1:10,6]
```

Nesta expressão, nós pedimos somente pelas linhas no intervalo entre 1 e 10. O R usa o ":" para criar um intervalo de valores, de tal forma que 1:10 se expande para 1, 2, 3, 4, 5, 6, 7, 8, 9, 10. Você pode visualizar isso digitando

```
> 1:10
```

Finalmente, se nós queremos todos os dados dos 10 primeiros respondentes, digite

```
> cdc[1:10,]
```

Ao deixar de fora um índice ou intervalo (nós não digitamos nada entre a vírgula e o colchete), nós obtemos todas as colunas. Quando iniciamos o uso do R, isso parece um pouco contra-intuitivo. Como um regra geral, omitimos o número da coluna para ver todas as colunas numa base de dados. Da mesma forma, se deixamos de fora um índice ou intervalo para as linhas, nós acessaríamos todas as observações, não apenas a 567ª, ou as linhas 1 a 10. Experimente o código seguinte para ver o peso de todos os 20.000 respondentes passarem voando por sua tela

```
> cdc[,6]
```

Recorde que a coluna 6 representa o peso dos respondentes, e portanto o comando acima mostra todos os pesos no conjunto de dados. Um método alternativo para acessar os dados sobre peso é utilizar o seu nome. Anteriormente, digitamos `names(cdc)` para ver todas as variáveis contidas no conjunto de dados `cdc`. Nós podemos utilizar qualquer um dos nomes de variáveis para selecionar itens no seu conjunto de dados.

```
> cdc$weight
```

O cifrão informa ao R para recuperar na base de dados `cdc` a coluna denominada `weight`. Uma vez que se trata de um único vetor, podemos formar subconjuntos utilizando apenas um único índice dentro dos colchetes. Nós verificamos o peso para o 567º respondente digitando

```
> cdc$weight[567]
```

Da mesma forma, para apenas os 10 primeiros respondentes

```
> cdc$weight[1:10]
```

O comando acima retorna o mesmo resultado que o comando `cdc[1:10,6]`. Tanto a notação linha-e-coluna quanto a notação utilizando o cifrão são amplamente utilizadas. Qual você escolhe depende da sua preferência pessoal.

Um Pouco Mais Sobre Formação de Subconjuntos

É frequentemente útil extrair todos os sujeitos (casos) de um conjunto de dados que possuem características específicas. Nós conseguimos isso por meio de comando *condicionais*. Primeiramente, considere expressões como

```
> cdc$gender == "m"
```

ou

```
> cdc$age > 30
```

Esses comandos produzem uma série de valores TRUE (verdadeiro) e FALSE (falso). Há um valor para cada respondente, sendo que TRUE indica que a pessoa era do sexo masculino (pelo primeiro comando) ou mais velha que 30 anos (segundo comando).

Vamos supor que queiramos extrair apenas os dados para homens na amostra, ou apenas para aqueles acima de 30 anos. Nós podemos utilizar a função do R subset para fazer isso por nós. Por exemplo, o comando

```
> mdata <- subset(cdc, cdc$gender == "m")
```

criará um novo conjunto de dados denominado mdata que contém apenas os homens do conjunto de dados cdc. Além de poder encontrá-lo em seu espaço de trabalho junto com suas dimensões, você pode dar uma olhada nas primeiras linhas como já fizemos

```
> head(mdata)
```

Este novo conjunto de dados contém as mesmas variáveis mas cerca de metade das linhas. Também é possível pedir para o R manter apenas variáveis específicas, um tópico que abordaremos num laboratório no futuro. Por enquanto, o importante é que podemos desmembrar os dados com base nos valores de uma ou mais variáveis.

Você também pode utilizar vários condicionais em conjunto com `&` e `|`. O `&` é lido como "e" de tal forma que

```
> m_and_over30 <- subset(cdc, cdc$gender == "m" & cdc$age > 30)
```

resultará nos dados para homens acima de 30 anos de idade. O caractere `|` é interpretado como "ou" de tal forma que

```
> m_or_over30 <- subset(cdc, cdc$gender == "m" | cdc$age > 30)
```

selecionará pessoas que são homens ou então acima de 30 anos (por que esse grupo seria interessante é difícil dizer, mas por enquanto entender o comando é o mais importante). A princípio, você pode utilizar quantos "e" e "ou" você quiser quando formar um subconjunto.

> **Exercício 5** Crie um novo objeto denominado under23_and_smoke (ou, se preferir, abaixo23_e_fuma) que contém todas as observações dos respondentes com menos de 23 anos que fumaram pelo menos 100 cigarros ao longo de sua vida. Escreva o comando que você utilizou para criar o novo objeto como resposta para esse exercício.

Dados Quantitativos

Com nossas ferramentas para criar subconjuntos a postos, podemos retornar à tarefa de hoje: criar sumários básicos do questionário BRFSS. Nós já olhamos os dados categoriais como smoke (fumante) e gender (gênero). Agora vamos nos concentrar nos dados quantitativos. Duas formas comuns de visualizar dados quantitativos é por meio de gráfico de caixas e histogramas. Nós podemos construir um gráfico de caixas para uma única variável com o seguinte comando.

```
> boxplot(cdc$height)
```

Você pode comparar a localização dos componentes da caixa examinando as estatísticas sumárias.

```
> summary(cdc$height)
```

Confirme que a mediana e os quartis superior e inferior informados no sumário numérico batem com os apresentados no gráfico. O objetivo de um gráfico de caixa é prover um pequeno esboço de uma variável com o propósito de comparar entre várias categorias. Podemos, por exemplo, comparar as alturas de homens e mulheres com

```
> boxplot(cdc$height ~ cdc$gender)
```

A notação aqui é nova. O caractere ~ pode ser lido como "versus" ou "como uma função de". Estamos, portanto, pedindo ao R para nos dar um gráfico de caixas das alturas no qual os grupos são definidos pelo gênero.

Na sequência, consideremos uma nova variável que não aparece diretamente neste conjunto de dados: o Índice de Massa Corporal (IMC). IMC é uma razão entre peso e altura que pode ser calculado da seguinte maneira:

$$IMC = \frac{peso\ (lbs)}{altura\ (pols)^2} * 703^{\dagger}$$

As duas linhas seguintes criam um novo objeto chamado `bmi` (de *Body Mass Index*) e então criamos um gráfico de caixas para esses valores, definindo grupos pela variável `cdc$genhlth`

```
> bmi <- (cdc$weight / cdc$height^2) * 703
> boxplot(bmi ~ cdc$genhlth)
```

Perceba que a primeira linha acima é apenas aritmética, mas é aplicada para todos os 20.000 número do conjunto de dados `cdc`. Ou seja, para cada um dos 20.000 participantes, pegamos seu peso, dividimos pelo quadrado de sua altura e multiplicamos por 703. O resultado é 20.000 valores de IMC, um para cada respondente. Essa é uma das razões pela qual gostamos do R: ele nos permite realizar cálculos como esse utilizando expressões bem simples.

> **Exercício 6** O que este gráfico de caixas mostra? Escolha outra variável categorial do conjunto de dados e verifique como ela se relaciona ao IMC. Liste a variável que você escolheu, por que você pensou que ela poderia ter relação com o IMC e indique o que o gráfico parece sugerir.

Por fim, vamos fazer alguns histogramas. Nós podemos verificar o histograma da idade de nossos respondentes com o comando

```
> hist(cdc$age)
```

Histogramas são geralmente uma boa maneira de visualizar a forma de uma distribuição, mas essa forma pode mudar dependendo como os dados são divididos entre os diferentes segmentos. Você pode controlar o número de segmentos adicionando um argumento ao comando. Nas próximas duas linhas, primeiro fazemos um histograma padrão da variável `bmi` e depois um com 50 segmentos.

```
> hist(bmi)
> hist(bmi, breaks = 50)
```

Perceba que você pode alternar entre gráficos que você criou clicando nas flechas de avançar e retroceder na região inferior direita do RStudio, logo acima dos gráficos. Quais as diferenças entre esses histogramas?

A esta altura, fizemos uma boa primeira exposição sobre análise das informações no questionário BRFSS. Nós descobrimos uma associação interessante entre fumo e gênero, e nós podemos comentar algo a respeito da relação entre a avaliação de saúde em geral dada pelas pessoas e seu próprio IMC. Nós também nos apropriamos de ferramentas computacionais essenciais – estatísticas sumárias, subconjuntos, e gráficos – que nos servirão bem ao longo deste curso.

[†] 703 é um fator de conversão aproximado para mudar as unidades do sistema métrico (metro e kilograma) para o sistema imperial (polegadas e libras). Isso é necessário porque os dados disponíveis estão no sistema imperial. No sistema métrico basta dividir o peso em quilogramas pelo quadrado da altura em metros.