Aula prática de modelos lineares

Leo Bastos e Marcelo Gomes (PROCC)

9 e 16 de agosto de 2018

Pré requisitos

Antes começarmos vamos aos pré-requisitos. Vocês precisam ter o R e o RStudio (opcional, mas recomendado) instalados. Ambos podem ser obtidos em:

- R: www.r-project.org
- RStudio: www.rstudio.org

Vamos usar na maior parte do tempo a sintaxe usada nos pacotes do tidyverse, mas eventualmente o R-base também será utilizado. Sempre que algum pacote for necessário vamos dizer qual é, mas inicialmente recomendamos que já instale o tidyverse (www.tidyverse.org).

```
# Comando para checar se o tidyverse já está instalado, se não ele será instala lo)
# Precisa de internet!
library(tidyverse)

# Se o tidyverse não estiver instalado rode o comando abaixo
# install.packages("tidyverse")
```

Para facilitar, e não perdermos tempo na leitura dos dados, disponibilizei os bancos a serem utilizados no arquivo "pratica_lm.RData".

```
# Lendo os dados
load("pratica_lm.RData")
```

Desfecho contínuo e exposição categórica

Exemplo: Dieta versus colesterol

o data frame *dieta* contém a medida de colesterol (mmol/L) de 18 indivíduos, e seu respectivo tipo de dieta {onívoro, (lacto-)vegetariano, vegano}.

```
# Descritiva (média e variância) do colesterol por tipo de dieta
dieta %>% group_by(Dieta) %>%
  summarise( n = n(),
             Media = mean(Colesterol),
             Variancia = var(Colesterol)
  )

# Boxplot
ggplot(data = dieta,
       mapping = aes(x = Dieta,
                     y = Colesterol,
                     fill = Dieta)) +
  geom_boxplot(show.legend = F) +
  theme_bw(base_size = 18) +
  xlab("") + ylab("Colesterol (mmol/L)")
```

Notem que existem diferentes formas de tratar um banco de dados e gerar figuras no R. Por uma preferência pessoal estou usando os pacotes do *tidyverse*.

Análise de variância

Realizando a análise de variância:

```
anova.dieta <- aov(formula = Colesterol ~ Dieta, data = dieta)
summary(anova.dieta)
```

Interprete a saída!

Faça as comparações 2 a 2, perceba o que está sendo testado.

```
pairwise.t.test(x = dieta$Colesterol,
                g = dieta$Dieta )
```

Testes não paramétricos

Teste Kruskal-Wallis é o teste equivalente a ANOVA, assim como o Teste de Wilcoxon é o equivalente ao teste t. Ambos baseados em manipulações de estatísticas de ordem, por isso costuma se dizer que são testes para medianas, mas na verdade testam de forma não paramétrica a igualdade das distribuições de probabilidade dos grupos.

```
kruskal.test(Colesterol ~ Dieta, data = dieta)


pairwise.wilcox.test(x = dieta$Colesterol,
                     g = dieta$Dieta)
```

Exercício 1

O data frame *racao* contém o peso em gramas de 45 ratos com 20 dias de idade alimentados a partir do desmame por duas semanas com uma das três marcas diferentes de ração. O objetivo do estudo é avaliar se a ração controle, que já vem sendo adotada, deve ser trocada. Avaliem os dados e concluam com base nesses dados qual a melhor estratégia para compra de ração.

Notem que há uma coluna com a linhagem do rato, será que ela é importante? Muda algo na conclusão?

Desfecho contínuo e exposição contínua

Consumo de açucar e cáries (DMFT)

O data frame *DMFT* contém a informação de 90 países a respeito do consumo de açucar, medido em kg per capita / ano, e o número médio de dentes com cárie, perdidos ou preenchidos (obturação) em crianças de 12 anos.

O data frame tem 3 colunas:

- Pais que indica se o país é industrializado (1) ou em desenvolvimento (2).
- Consumo que indica o consumo de açucar.
- DMFT que indica o número médio de DFMT em crianças de 12 anos.

O scatter plot pode ser obtido com o seguinte comando:

```
p <- ggplot(dmft, aes(y = DMFT, x = Consumo))
p <- p + geom_point() +
  xlab("Consumo de açucar (kg per capita/ano)") +
  theme_bw(base_size = 18)
p
```

Notem que a Figura está "salva" no objeto p, esse objeto será usado mais adiante para incluirmos a reta ajustada e posteriormente o intervalo.

Vamos ajustar um modelo linear aos dados. A fórmula deve contem na esquerda o desfecho, e separados por ~ a(s) exposição(ções) do lado direito. Nesse exemplo, o desfecho é o indice DMFT, e a exposição o consumo de açucar. Os comandos para ajustar um modelo linear no R são:

```
output <- lm(DMFT ~ Consumo, data = dmft)
output
```

A saída completa é dada por

```
summary(output)

# Só os coeficientes
output$coefficients

# Ou
coef(output)
```

Adicionando a reta ao scatter plot já construído p:

```
p + geom_abline(slope = output$coefficients[2],
                intercept = output$coefficients[1])
```

Análise de resíduos

O modelo linear tem algumas suposições que precisam ser verificadas:

- Normalidade
- Independência
- Variância constante

Essas suposições podem ser verificadas de forma descritiva via análise dos resíduos do modelo. Quando se ajusta um modelo linear no R, os resíduos são guardados no objeto do modelo. No exemplo, os resíduos se encontram no objeto *output$residuals*. Para visualizar os resíduos use o comando

```
output$residuals
```

A suposição de normalidade é verificada de forma visual através de um histograma, onde procuramos por uma forma de "sino". Outra forma seria via qqplot (quantile-quantile plot) onde cruza-se os quantis amostrais contra os quantis teóricos se a distribuição fosse normal, nesse caso espera-se uma reta.

```
hist(output$residuals)

qqnorm(output$residuals)
```

Podemos testar estatísticamente se os resíduos são normalmente distribuídos, usaremos dois testes: (i) O teste Shapiro-Wilks que é específicos para normalidade; (ii) o teste de Kolmogorov-Smirnov que é um teste mais geral, onde o teste para normalidade é um caso particular. Em ambos os casos, a hipótese nula é

H_0: As observações de interesse seguem uma distribuição normal.

Os comandos para dois testes:

```
# Teste Shapiro-Wilks
shapiro.test(output$residuals)

# Teste Kolmogorov-Smirnov
ks.test(x = output$residuals, y = "pnorm")
```

Percebam que a suposição de normalidade não parece ser respeitada.

A suposição de variância constante e independência pode ser verificada fazendo um scatter plot do resíduo versus a ordem dos dados, ou resíduos versus alguma covariável usada no modelo e não usada também, com a finalidade de observar algum padrão. Esperamos observar uma nuvem de pontos sem nenhum padrão óbvio e com variabilidade constante.

```
# Residuos versus ordem dos dados
plot(output$residuals)

# Resíduos versos Consumo
plot(x = dmft$Consumo, y = output$residuals)
```

Note a presença de um cluster de menor variabilidade nos resíduos, sugerindo uma violação da suposição de variância constante. A presença dessa cluster também pode estar associada com uma violação a hipótese de independência.

Previsões

Assumindo que as suposições do modelo não foram violadas, temos indícios que foram as suposições foram violadas, vamos fazer previsões usando o modelo linear.

Para isso, usamos a função *predict*. Os parâmetros importantes são:

- *object* que deve entrar o objeto do modelo;
- *se.fit* se você quer (ou não) o desvio padrão da previsão (tecnicamente conhecido como erro padrão, *standard error*), o valor *default* é falso (F ou FALSE);
- *newdata* um data frame somente com os valores da(s) exposição(ções) que você gostaria de prever. o valor *default* é o seu próprio banco.

```
# Suponha que queremos saber qual o número médio de DMFT
# quando um país consome 30 Kg per capita / ano, e também
# quando outro país consome 45 Kg per capita / ano.

previsao0 <- predict(object = output,
                     se.fit = F,
                     newdata = data.frame(
                       Consumo = c(30,45)
                     )
)

# Varie o valor de se.fit de F para T e perceba a diferença
# no objeto resultante
previsao0

# Agora suponha que queremos estimar indíce DMFT para valores
# de consumo variando de 0 a 64 Kg per capita / ano.
previsao <- predict(object = output,
                    se.fit = T,
```

```
                    newdata = data.frame(
                      Consumo = 0:64
                    )
)
```

Agora vamos incluir as estimativas de previsão ao scatter plot. Lembrando que agora o intervalo de confiança para a previsão pode ser calculado simplesmente calculando $Prev \pm 1.96se$.

```
# Para incluir na figura, um novo data frame é criado onde temos
# uma coluna com as previsoes, outra com o desvio paradrao das
# previsões, e com a função mutate criou duas novas colunas com
# os limites inferior e superior do intervalo de confinça, LI,
# LS respectivamente.
previsao.df <- data.frame(Prev = previsao$fit,
                          sd = previsao$se.fit) %>%
  mutate(LI = Prev - 1.96 * sd,
         LS = Prev + 1.96 * sd,
         ID = 0:64)

# Para o plot podemos usar a funcao geom_ribbon para o
# intervalo sombreado.
p +  geom_ribbon(data = previsao.df, aes(x = ID, y = Prev,
                                         ymin = LI, ymax = LS),
                 alpha = 0.25 ) +
  geom_line(data = previsao.df, aes(x = ID, y = Prev))
```

Exercício 2

Faça suponha que um novo país tenha sido criado e ainda não teve seu índice DMFT avaliado pela OMS, mas alguns estudos apontam que o consumo de açucar desse novo país varie de 40 a 50 Kg per capita / ano. Assumindo que o modelo linear é o melhor modelo que se tem em mãos, o que podemos esperar da média de DMFT para crianças de 12 anos desse novo país?

Função não linear

Vamos agora ajustar uma função não linear, vimos que podemos ajustar um modelo exponencial, $Y = Ae^{BX}\varepsilon$, usando um modelo linear simples da forma $\log(Y) = \alpha + \beta X + \epsilon$, onde $\alpha = \log(A)$ e $\epsilon = \log(\varepsilon)$. No R, isso se resume a

```
output2 <- lm(log(DMFT) ~ Consumo, data = dmft)
output2
```

Adicionando a curva ao scatter plot já construído p:

```
# Criando um banco auxilar com uma coluuna com os valores
# ajustados. Perceba que o valor ajustado é para log(Y), e
# portanto devemos tomar a exponencial do valor ajustado para
# retornarmos a escala de Y.
aux <- dmft %>% bind_cols( Fit = exp(output2$fitted.values))

# A nova curva pode ser incluida usando a funcao geom_line e
# o "novo" banco de dados.
p + geom_line(data = aux, aes(x = Consumo, y = Fit))

# Removendo o banco auxiliar
rm(aux)
```

A transformação logaritimica é uma transformação usada para estabilizar variância, outra transformação utilizada com esse objetivo é a transformação Box-Cox (Veja *?MASS::boxcox*). O ponto negativo no uso de transformações é a dificuldade de interpretação de coeficientes. No caso, a transformação logaritimica no desfecho implica no ajuste do modelo exponencial.

Exercício 3

Construa a curva dos intervalos de confiança agora para esse modelo não linear. Analise os resíduos e repita o exercício 2.

Duas variáveis explicativas

Continuando o exemplo DMFT

Ainda no exemplo dos DMFT versus consumos de açucar, vamos incluir a análise a variável *Pais*, e avaliar sua influência em quatro modelos:

```
# Redefinindo a variável Pais
dmft <- dmft %>%
  mutate(
    Pais = factor(Pais, levels= 1:2,
                  labels = c("Industrializado",
                             "Em desenvolvimento")
    )
  )

# Scatter plot agora colorindo por tipo de pais
p <- ggplot(dmft, aes(y = DMFT, x = Consumo, colour = Pais))
p <- p + geom_point() + xlab("Consumo de açucar (kg per capita/ano)") +
  theme_bw(base_size = 18) + theme(legend.position = c(.25, 0.8)) +
  labs(colour = "País") +
  theme(
    legend.background = element_rect(
      linetype = 1, size = 0.25, colour = 1)
  )
p
```

Vamos agora ajustar quatro modelos distintos:

```
# Modelo ignorando o tipo de país
modelo1 <- lm(DMFT ~ Consumo, data = dmft)

# Modelo variando intercepto
modelo2 <- lm(DMFT ~ Consumo + Pais, data = dmft)

# Modelo variando slope
modelo3 <- lm(DMFT ~ Consumo + Consumo:Pais,
              data = dmft)

# Modelo variando intercepto e slope
modelo4 <- lm(DMFT ~ Consumo + Pais + Consumo:Pais,
              data = dmft)
```

Os comandos abaixo servem para replicar as diferentes curvas ajustadas por cada modelo:

```
aux <- coef(modelo1)
p + geom_abline(slope = aux[2], intercept = aux[1])

aux <- coef(modelo2)
p + geom_abline(slope = aux[2],
                intercept = aux[1],
                color = "#F8766D"
) +
  geom_abline(slope = aux[2],
              intercept = aux[1]+aux[3],
              color = "#00BFC4"
  )

aux <- coef(modelo3)
p + geom_abline(slope = aux[2],
                intercept = aux[1],
                color = "#F8766D") +
  geom_abline(slope = aux[2]+aux[3],
              intercept = aux[1],
              color = "#00BFC4")

aux <- coef(modelo4)
p + geom_abline(slope = aux[2],
                intercept = aux[1],
                color = "#F8766D") +
  geom_abline(slope = aux[2]+aux[4],
              intercept = aux[1]+aux[3],
              color = "#00BFC4")
```

Exercício 4

Qual dos quatro modelos parece ser o mais razoável? Verifique a significância dos coeficientes desse modelo. Compare também os modelos usando critérios de informação (AIC e BIC).

Exemplo do estudo de saúde do coração escocês (SHHS)

Esse estudo contém a informação de 150 indivíduos nas linhas, e nas colunas temos o índice de massa copórea (*BMI*), sexo (*Sex*) e histórico de tabagismo (*Smoking*).

Os comandos abaixo nos permite construir todas as combinações de modelos usando as variáveis sexo e histórico de tabagismo.

```
# modelo 1 (somente sexo)
modelo1.shhs <- lm(BMI ~ Sex, data = SHHS)

# modelo 2 somente historico de tabagismo
modelo2.shhs <- lm(BMI ~ Smoking, data = SHHS)

# modelo 3 sexo e historico de tabagismo
modelo3.shhs <- lm(BMI ~ Sex + Smoking, data = SHHS)
```

```r
# modelo 4 sexo, historico de tabagismo e sua interação
modelo4.shhs <- lm(BMI ~ Sex * Smoking, data = SHHS)
```

Exercício 5

Para cada modelo tente interpretar os coeficientes, avalie a significância estatística dos mesmos. Escolha um dos modelos (usando critérios de informação), avalie os resíduos do modelo escolhido, e, finalmente, estime o IMC esperado para uma mulher ex fumante, faça isso somando os coeficientes apropriados, e também usando a função *predict*.

Exercício 6

Considere agora o data frame *VigitelRJ2016*, contendo 1934 observações de pessoas entrevistadas por telefone na pesquisa Vigitel do ano de 2016 na cidade do Rio de Janeiro, com IMC (autoreferido), histórico de tabagismo, sexo e idade. Onde a idade é representada de duas formas, como idade em anos e faixa etária.

a) Avalie a diferença entre as variáveis Idade e Faixa Etária, i.e. encontre o melhor modelo usando a idade e o melhor modelo com faixa etária, compare de forma quantitativa via AIC ou BIC, quanto de forma "qualitativa" avaliando a interpretação dos dois modelos.

b) Com o modelo final escolhido, analise os resíduos, e interprete os coeficientes. Usando a função *predict* construa uma tabela com IMC esperado com intervalo de confiança para todas as combinações de sexo e histórico de tabagismo. (Para a função predict funcionar, a idade deve ser fixada em algum valor ou categoria qualquer.)