

Seleção em modelos de regressão

Elias T Krainski

Curso de Especialização em
Data Science & Big Data
Universidade Federal do Paraná

September 15, 2018



Introdução

Melhor conjunto de preditoras

Seleção passo-a-passo

Regularização



Introdução

Introdução



Introdução

- Em modelos de regressão temos a média de uma variável resposta a modelada em função de p variáveis preditoras

$$\begin{aligned} E(y) &= g^{-1}(\eta) \\ \eta &= f(x_1, x_2, \dots, x_p) \end{aligned} \tag{1}$$

Introdução

- ▶ Em modelos de regressão temos a média de uma variável resposta a modelada em função de p variáveis preditoras

$$\begin{aligned} E(y) &= g^{-1}(\eta) \\ \eta &= f(x_1, x_2, \dots, x_p) \end{aligned} \tag{1}$$

- ▶ Suponha ainda que temos $m \gg p$ variáveis preditoras candidatas

Introdução

- ▶ Em modelos de regressão temos a média de uma variável resposta a modelada em função de p variáveis preditoras

$$\begin{aligned} E(y) &= g^{-1}(\eta) \\ \eta &= f(x_1, x_2, \dots, x_p) \end{aligned} \tag{1}$$

- ▶ Suponha ainda que temos $m \gg p$ variáveis preditoras candidatas
- ▶ Quais p preditoras escolher entre as m (muitas) candidatas?

Estratégias comuns

- ▶ Melhor conjunto de preditoras
 - ▶ estima todos os possíveis modelos e selecionar o melhor

Estratégias comuns

- ▶ Melhor conjunto de preditoras
 - ▶ estima todos os possíveis modelos e selecionar o melhor
- ▶ Seleção passo a passo (*stepwise*)
 - ▶ *forward*: parte de um modelo sem preditora e adiciona a melhor candidata a cada passo
 - ▶ *backward*: parte do modelo com todas e retira a pior a cada passo
 - ▶ *both*: combina anteriores

Estratégias comuns

- ▶ Melhor conjunto de preditoras
 - ▶ estima todos os possíveis modelos e selecionar o melhor
- ▶ Seleção passo a passo (*stepwise*)
 - ▶ *forward*: parte de um modelo sem preditora e adiciona a melhor candidata a cada passo
 - ▶ *backward*: parte do modelo com todas e retira a pior a cada passo
 - ▶ *both*: combina anteriores
- ▶ Regularização
 - ▶ solução para outro problema que tem o efeito (colateral) de selecionar preditoras



Melhor conjunto de preditoras

Número de possíveis modelos

- ▶ o número de possíveis modelos com p variáveis dentre m candidatas é

$$\binom{m}{p} = \frac{m!}{p!(m-p)!}$$

Número de possíveis modelos

- ▶ o número de possíveis modelos com p variáveis dentre m candidatas é

$$\binom{m}{p} = \frac{m!}{p!(m-p)!}$$

- ▶ suponha que temos $m = 4$ preditoras candidatas
 - ▶ o número de possíveis modelos com $p = 0$ é 1
 - ▶ o número de possíveis modelos com $p = 1$ é $m = 4$
 - ▶ o número de possíveis modelos com $p = 2$ é 6
 - ▶ o número de possíveis modelos com $p = 3$ é 4
 - ▶ $f(x_2, x_3, x_4)$, $f(x_1, x_3, x_4)$, $f(x_1, x_2, x_4)$ e $f(x_1, x_2, x_3)$
 - ▶ o número de possíveis modelos com $p = 4$ é 1

Número de possíveis modelos

- ▶ o número de possíveis modelos com p variáveis dentre m candidatas é

$$\binom{m}{p} = \frac{m!}{p!(m-p)!}$$

- ▶ suponha que temos $m = 4$ preditoras candidatas
 - ▶ o número de possíveis modelos com $p = 0$ é 1
 - ▶ o número de possíveis modelos com $p = 1$ é $m = 4$
 - ▶ o número de possíveis modelos com $p = 2$ é 6
 - ▶ o número de possíveis modelos com $p = 3$ é 4
 - ▶ $f(x_2, x_3, x_4)$, $f(x_1, x_3, x_4)$, $f(x_1, x_2, x_4)$ e $f(x_1, x_2, x_3)$
 - ▶ o número de possíveis modelos com $p = 4$ é 1
- ▶ Número total de possíveis modelos é 2^m
 - ▶ $2^{10} = 1024$, $2^{20} = 1048576$, $2^{50} = 1.1259e+15, \dots$

Número de possíveis modelos

- ▶ o número de possíveis modelos com p variáveis dentre m candidatas é

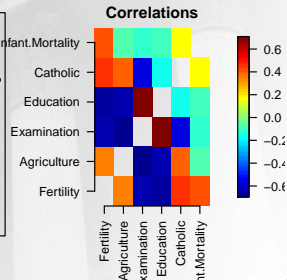
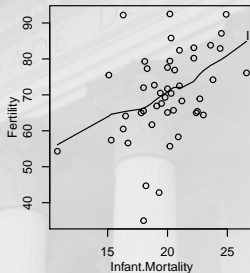
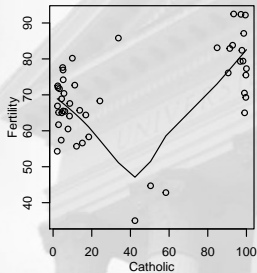
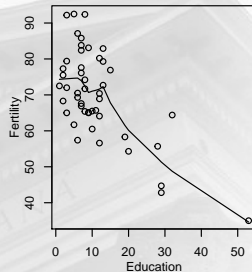
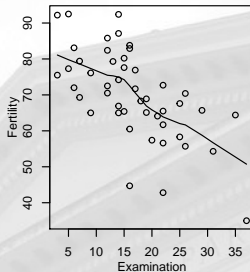
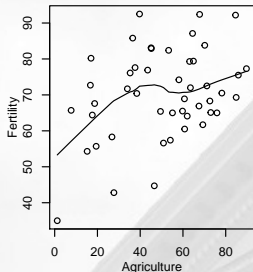
$$\binom{m}{p} = \frac{m!}{p!(m-p)!}$$

- ▶ suponha que temos $m = 4$ preditoras candidatas
 - ▶ o número de possíveis modelos com $p = 0$ é 1
 - ▶ o número de possíveis modelos com $p = 1$ é $m = 4$
 - ▶ o número de possíveis modelos com $p = 2$ é 6
 - ▶ o número de possíveis modelos com $p = 3$ é 4
 - ▶ $f(x_2, x_3, x_4)$, $f(x_1, x_3, x_4)$, $f(x_1, x_2, x_4)$ e $f(x_1, x_2, x_3)$
 - ▶ o número de possíveis modelos com $p = 4$ é 1
- ▶ Número total de possíveis modelos é 2^m
 - ▶ $2^{10} = 1024$, $2^{20} = 1048576$, $2^{50} = 1.1259e+15$, ...
 - ▶ A probabilidade de ganhar na megacena com um único bilhete de 6 números é $1/\binom{60}{6} = 1/50063860$

Estimando todos os possíveis modelos

- ▶ No modelo linear ($y \sim \text{Normal}$), usa-se elementos da decomposição QR (usada na estimação)
 - ▶ pacote leaps
- ▶ Para GLMs ver <https://github.com/hadley/meifly/>

Exemplo Swiss



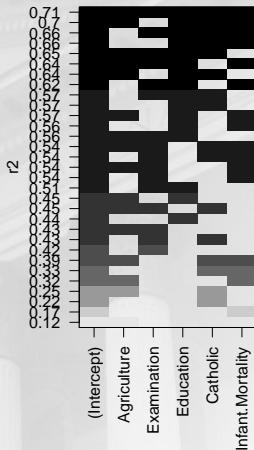
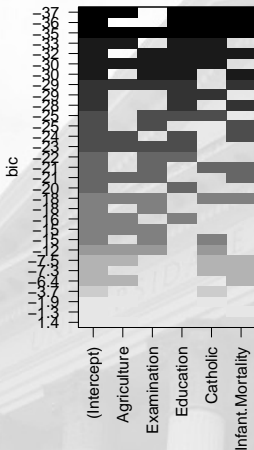
Swiss: regressão múltipla

```
summary(lm(Fertility ~., swiss))  
##  
## Call:  
## lm(formula = Fertility ~ ., data = swiss)  
##  
## Residuals:  
##      Min       1Q   Median       3Q      Max   
## -15.2743  -5.2617   0.5032   4.1198  15.3213   
##  
## Coefficients:  
##              Estimate Std. Error t value Pr(>|t|)      
## (Intercept)   66.91518    10.70604   6.250 1.91e-07 ***  
## Agriculture   -0.17211     0.07030  -2.448 0.01873 *    
## Examination   -0.25801     0.25388  -1.016 0.31546        
## Education      -0.87094     0.18303  -4.758 2.43e-05 ***  
## Catholic       0.10412     0.03526   2.953 0.00519 **     
## Infant.Mortality 1.07705     0.38172   2.822 0.00734 **     
## ---  
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1  
##  
## Residual standard error: 7.165 on 41 degrees of freedom  
## Multiple R-squared:  0.7067, Adjusted R-squared:  0.671   
## F-statistic: 19.76 on 5 and 41 DF,  p-value: 5.594e-10
```

Exemplo: todos os possíveis modelos

```
library(leaps)
```

```
a <- regsubsets(Fertility~., nbest=10, data=swiss)
```





Seleção passo-a-passo

Comparação de modelos

- ▶ Quando dois modelos diferentes são propostos, é necessário compará-los
- ▶ Há diversas medidas de comparação de modelos
- ▶ Essas medidas podem ser classificadas em dois grupos
 - ▶ Medidas de ajuste (do modelo aos dados)
 - ▶ Medidas de predição (fora da amostra)
- ▶ Modelos mais flexíveis tendem à se ajustar melhor aos dados mas predizer pior

Seleção via inclusão passo-a-passo (*forward*)

1. Inicia com modelo sem covariáveis
2. Para cada uma das covariáveis candidatas
 - ▶ Calcula o ganho pela sua inclusão no modelo atual
3. Ordena as variáveis pelo seu ganho
4. Se não houver ganho positivo, para
5. Se houver ganho positivo, adiciona a variável com maior ganho e volta ao passo 2

Seleção via retirada passo-a-passo (*backward*)

1. Ajusta o modelo com todas as covariáveis
2. Para cada covariável presente no modelo atual
 - ▶ Calcula o ganho pela sua retirada do modelo
3. Ordena as variáveis pelo seu ganho
4. Se não houver ganho positivo, para
5. Se houver ganho positivo, retira a variável com maior ganho e volta ao passo 2

Seleção passo-a-passo com ambas as direções (*both*)

1. Ajusta um modelo inicial
2. Para cada covariável presente no modelo atual
 - ▶ Calcula o ganho pela sua inclusão no modelo atual
3. Para cada covariável candidata não presente no modelo atual
 - ▶ Calcula o ganho pela sua retirada do modelo
4. Ordena o ganho devido a cada covariável
 - ▶ Se o maior ganho for negativo, para
5. Identifica a variável com maior ganho
 - ▶ Se a covariável com maior ganho está presente no modelo, retira-a e volta ao passo 2
 - ▶ Se a covariável com menor ganho está fora do modelo, adiciona-a e volta ao passo 2

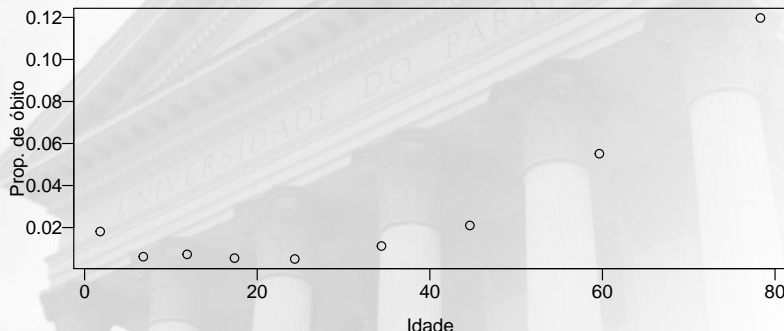
Exemplo RD de interações Paraná - Julho 2018

```
library(read.dbc)
dim(rd <- read.dbc("~/dados/datasus/rd/PR/RDPR1807.dbc"))
names(rd)
```

```
## [1] "UF_ZI" "ANO_CMPT" "MES_CMPT" "ESPEC" "CGC_HOSP" "N_AIH" "IDENT"
## [8] "CEP" "MUNIC_RES" "NASC" "SEXO" "UTI_MES_IN" "UTI_MES_AN" "UTI_MES_AL"
## [15] "UTI_MES_TO" "MARCA_UTI" "UTI_INT_IN" "UTI_INT_AN" "UTI_INT_AL" "UTI_INT_TO" "DIAR_ACOM"
## [22] "QT_DIARIAS" "PROC_SOLIC" "PROC_REA" "VAL_SH" "VAL_SP" "VAL_SADT" "VAL_RN"
## [29] "VAL_ACOMP" "VAL_ORTP" "VAL_SANGUE" "VAL_SADTSR" "VAL_TRANSP" "VAL_OBSANG" "VAL_PED1AC"
## [36] "VAL_TOT" "VAL_UTI" "US_TOT" "DT_INTER" "DT_SAIDA" "DIAG_PRINC" "DIAG_SECUN"
## [43] "COBRANCA" "NATUREZA" "NAT_JUR" "GESTAO" "RUBRICA" "IND_VDRL" "MUNIC_MOV"
## [50] "COD_IDADE" "IDADE" "DIAS_PERM" "MORTE" "NACIONAL" "NUM_PROC" "CAR_INT"
## [57] "TOT_PT_SP" "CPF_AUT" "HOMONIMO" "NUM_FILHOS" "INSTRU" "CID_NOTIF" "CONTRACEP1"
## [64] "CONTRACEP2" "GESTRISCO" "INSC_PN" "SEQ_AIH5" "CBOR" "CNAER" "VINCPREV"
## [71] "GESTOR_COD" "GESTOR_TP" "GESTOR_CPF" "GESTOR_DT" "CNES" "CNPJ_MANT" "INFEHOSP"
## [78] "CID_ASSO" "CID_MORTE" "COMPLEX" "FINANC" "FAEC_TP" "REGCT" "RACA_COR"
## [85] "ETNIA" "SEQUENCIA" "REMESSA" "AUD_JUST" "SIS_JUST" "VAL_SH_FED" "VAL_SP_FED"
## [92] "VAL_SH_GES" "VAL_SP_GES" "VAL_UCI" "MARCA_UCI" "DIAGSEC1" "DIAGSEC2" "DIAGSEC3"
## [99] "DIAGSEC4" "DIAGSEC5" "DIAGSEC6" "DIAGSEC7" "DIAGSEC8" "DIAGSEC9" "TPDISEC1"
## [106] "TPDISEC2" "TPDISEC3" "TPDISEC4" "TPDISEC5" "TPDISEC6" "TPDISEC7" "TPDISEC8"
## [113] "TPDISEC9"
```

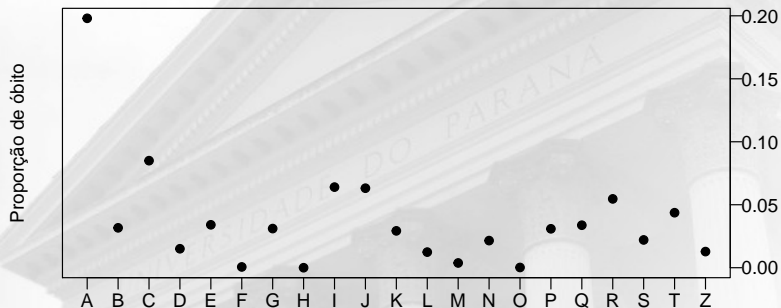
Eexmplo RD: Morte por idade

```
idade.b <- c(0,5,10,15,20,30,40,50,70, max(rd$IDADE)+1e-6)
idade.g <- cut(rd$IDADE, idade.b, right=FALSE)
idade.m <- tapply(rd$IDADE, idade.g, mean)
p.idade.g <- tapply(rd$MORTE, idade.g, mean)
plot(idade.m, p.idade.g, xlab='Idade', ylab='Prop. de óbito')
```



Exemplo: Morte por diagnóstico principal

```
cid1 <- substr(rd$DIAG_PRINC, 1, 1)
p.cid1 <- tapply(rd$MORTE, cid1, mean)
plot(p.cid1, axes=FALSE, ylab='Proporção de óbito', pch=19)
box(); axis(4); axis(1, 1:length(p.cid1), names(p.cid1))
```



```
round(100*sort(prop.table(table(cid1))))
```

```
## cid1
## B Q H L R E Z M G P D T A F N C S K I J O
## 1 1 1 2 2 2 2 2 2 2 3 3 4 6 7 8 10 13 14 14
```

Exemplo: RD regressão logística

```
dados <- as.data.frame(model.matrix(~cid1-1))
names(dados) <- gsub('cid1', '', names(dados))
dados <- dados[, c1s <- c('A', 'C', 'I', 'J', 'O', 'R')]
dados$idade <- cut(rd$IDADE, c(0,5,40,60,100), right=TRUE)
dados$y <- rd$MORTE; dados <- dados[complete.cases(dados),]
summary(m3 <- glm(y ~ ., family=binomial, data=dados))
```

```
##
## Call:
## glm(formula = y ~ ., family = binomial, data = dados)
##
## Deviance Residuals:
##      Min       1Q   Median       3Q      Max
## -0.9055  -0.3258  -0.2042  -0.1199   4.1455
##
## Coefficients:
##              Estimate Std. Error z value Pr(>|z|)
## (Intercept)  -5.34585    0.14573  -36.684 < 2e-16 ***
## A             2.22955    0.06594   33.812 < 2e-16 ***
## C             0.98895    0.06293   15.715 < 2e-16 ***
## I             0.57411    0.05801    9.896 < 2e-16 ***
## J             0.96368    0.05737   16.796 < 2e-16 ***
## O            -3.66015    0.70872   -5.164 2.41e-07 ***
## R             0.80684    0.12489    6.460 1.04e-10 ***
## idade(5,40]   0.41342    0.15722    2.630 0.00855 **
## idade(40,60]  1.48630    0.14776   10.059 < 2e-16 ***
## idade(60,100] 2.43648    0.14308   17.029 < 2e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## (Dispersion parameter for binomial family taken to be 1)
##
##    Null deviance: 24917  on 73048  degrees of freedom
## Residual deviance: 20892  on 73039  degrees of freedom
## AIC: 20912
##
## Number of Fisher Scoring iterations: 10
```

Teste de retirada de uma variável do modelo

```
drop1(m3, test='Ch')
## Single term deletions
##
## Model:
## y ~ A + C + I + J + O + R + idade
##      Df Deviance   AIC      LRT Pr(>Chi)
## <none>      20892 20912
## A      1      21845 21863   953.27 < 2.2e-16 ***
## C      1      21118 21136   226.50 < 2.2e-16 ***
## I      1      20987 21005    95.52 < 2.2e-16 ***
## J      1      21162 21180   270.01 < 2.2e-16 ***
## O      1      21006 21024   114.81 < 2.2e-16 ***
## R      1      20926 20944    34.68  3.88e-09 ***
## idade  3      22422 22436  1530.92 < 2.2e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
drop1(m3, test='Ch', k=log(nrow(dados)))
## Single term deletions
##
## Model:
## y ~ A + C + I + J + O + R + idade
##      Df Deviance   AIC      LRT Pr(>Chi)
## <none>      20892 21004
## A      1      21845 21946   953.27 < 2.2e-16 ***
## C      1      21118 21219   226.50 < 2.2e-16 ***
## I      1      20987 21088    95.52 < 2.2e-16 ***
## J      1      21162 21262   270.01 < 2.2e-16 ***
## O      1      21006 21107   114.81 < 2.2e-16 ***
## R      1      20926 21027    34.68  3.88e-09 ***
## idade  3      22422 22501  1530.92 < 2.2e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

Exemplo RD *backward*

```
selb <- step(m3, direction='backward', trace=0)
summary(selb)
```

```
selb <- step(m3, direction='backward', trace=0)
summary(selb)
##
## Call:
## glm(formula = y ~ A + C + I + J + O + R + idade, family = binomial,
##      data = dados)
##
## Deviance Residuals:
##      Min       1Q   Median       3Q      Max
## -0.9055  -0.3258  -0.2042  -0.1199   4.1455
##
## Coefficients:
##              Estimate Std. Error z value Pr(>|z|)
## (Intercept)  -5.34585    0.14573  -36.684 < 2e-16 ***
## A              2.22955    0.06594   33.812 < 2e-16 ***
## C              0.98895    0.06293   15.715 < 2e-16 ***
## I              0.57411    0.05801    9.896 < 2e-16 ***
## J              0.96368    0.05737   16.796 < 2e-16 ***
## O             -3.66015    0.70872   -5.164 2.41e-07 ***
## R              0.80684    0.12489    6.460 1.04e-10 ***
## idade(5,40]   0.41342    0.15722    2.630 0.00855 **
## idade(40,60]  1.48630    0.14776   10.059 < 2e-16 ***
## idade(60,100] 2.43648    0.14308   17.029 < 2e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## (Dispersion parameter for binomial family taken to be 1)
##
##      Null deviance: 24917  on 73048  degrees of freedom
## Residual deviance: 20892  on 73039  degrees of freedom
## AIC: 20912
```

Teste de adição de uma variável no modelo

```
m0 <- glm(y ~ 0, family=binomial, data=dados)
sc0 <- paste('~.+ ', paste(c(c1s, 'idade'), collapse='+'))
add1(m0, as.formula(sc0), test='Ch')
```

```
m0 <- glm(y ~ 0, family=binomial, data=dados)
sc0 <- paste('~.+ ', paste(c(c1s, 'idade'), collapse='+'))
add1(m0, as.formula(sc0), test='Ch')
## Single term additions
##
## Model:
## y ~ 0
##      Df Deviance    AIC   LRT Pr(>Chi)
## <none>      101267 101267
## A      1   100348 100350   920 < 2.2e-16 ***
## C      1    96941  96943   4327 < 2.2e-16 ***
## I      1    92766  92768   8501 < 2.2e-16 ***
## J      1    91983  91985   9285 < 2.2e-16 ***
## O      1    86991  86993 14277 < 2.2e-16 ***
## R      1    99915  99917  1352 < 2.2e-16 ***
## idade  4    22165  22173 79103 < 2.2e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

Exemplo RD forward

```
self <- step(m0, as.formula(sc0), test='Ch', trace=0)
summary(self)
```

```
self <- step(m0, as.formula(sc0), test='Ch', trace=0)
summary(self)
##
## Call:
## glm(formula = y ~ idade + A + O + J + C + I + R - 1, family = binomial,
##      data = dados)
##
## Deviance Residuals:
##      Min       1Q   Median       3Q      Max
## -0.9055  -0.3258  -0.2042  -0.1199   4.1455
##
## Coefficients:
##              Estimate Std. Error z value Pr(>|z|)
## idade(0,5]      -5.34585    0.14573 -36.684 < 2e-16 ***
## idade(5,40]     -4.93243    0.07357 -67.041 < 2e-16 ***
## idade(40,60]    -3.85954    0.05068 -76.154 < 2e-16 ***
## idade(60,100]  -2.90937    0.04247 -68.502 < 2e-16 ***
## A                2.22955    0.06594  33.812 < 2e-16 ***
## O               -3.66015    0.70872  -5.164 2.41e-07 ***
## J                0.96368    0.05737  16.796 < 2e-16 ***
## C                0.98895    0.06293  15.715 < 2e-16 ***
## I                0.57411    0.05801   9.896 < 2e-16 ***
## R                0.80684    0.12489   6.460 1.04e-10 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## (Dispersion parameter for binomial family taken to be 1)
##
##      Null deviance: 101267  on 73049  degrees of freedom
## Residual deviance: 20892  on 73039  degrees of freedom
## AIC: 20912
```




Regularização

Regularização

- ▶ Há casos em que $p > n$ (genética por exemplo)
- ▶ Nesses casos é impossível estimar todos os p coeficientes
- ▶ A regularização é uma forma de tornar isso possível

Regularização

- ▶ É uma penalização dos coeficientes
 - ▶ Ridge: $\sum_j \beta_j^2 < t$
 - ▶ LASSO: $\sum_j |\beta_j| < t$

Regularização: material

Trevor Hastie

[home](#) [biography](#) [research](#) [software & data](#) [lectures](#) [students](#) [publications](#) [links](#)

MY PUBLICATIONS

BOOKS ■■■

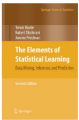
- [Books](#)
- [Papers - 2017](#)
- [Papers - 2016](#)
- [Papers - 2015](#)
- [Papers - 2014](#)
- [Papers - 2013](#)
- [Papers - 2012](#)
- [Papers - 2011](#)
- [Papers - 2010](#)
- [Papers - 2009](#)
- [Papers - 2008](#)
- [Papers - 2007](#)
- [Papers - 2006](#)
- [Papers - 2005](#)
- [Papers - 2004](#)
- [Papers - 2003](#)
- [Papers - 2002](#)
- [Papers - 2001](#)
- [Papers - 2000](#)
- [Papers - 1999](#)
- [Papers - 1998](#)
- [Papers - 1997](#)




[Computer Age Statistical Inference: Algorithms, Evidence and Data Science](#)
by Bradley Efron and Trevor Hastie (August 2016)
[Book Homepage pdf \(8.5 Mb, corrected online\)](#)




[An Introduction to Statistical Learning](#)
by Gareth James, Daniela Witten, Trevor Hastie and Robert Tibshirani (June 2013)
[Book Homepage pdf \(9.4Mb, 6th corrected printing\)](#)




[The Elements of Statistical Learning: Data Mining, Inference, and Prediction \(Second Edition\)](#)
by Trevor Hastie, Robert Tibshirani and Jerome Friedman (2009)
[Book Homepage pdf \(13Mb, correct, 12th print\)](#)



[Statistical Learning with Sparsity: the Lasso and Generalizations](#)
by Trevor Hastie, Robert Tibshirani and Martin Wainwright (May 2015)
[Book Homepage pdf \(10.5Mb, corrected online\)](#)



[Generalized Additive Models](#)



[Statistical Inference](#)



[Statistical Models in Epidemiology](#)

Figure 1: Página com lista de livros do prof. Trevor Hastie
web.stanford.edu/~hastie/pub.htm

DSBD

Exemplo de LASSO

Simula 10 covariáveis, das quais três são correlacionadas entre si

```
n <- 1000
```

```
p <- 10
```

```
set.seed(1)
```

```
x <- matrix(runif(n*p), n)
```

```
x[,2] <- x[,2]+x[,1]
```

```
x[,3] <- x[,3]-x[,4]+x[,5]
```

```
round(100*cor(x))
```

```
round(100*cor(x))
```

```
##      [,1] [,2] [,3] [,4] [,5] [,6] [,7] [,8] [,9] [,10]
## [1,]  100   69  -2    4    3    1    1  -1    0    0
## [2,]   69  100  -3    4    3    3    3  -3    1  -1
## [3,]  -2   -3  100  -60   59    0  -3  -3  -2    3
## [4,]    4    4  -60  100  -3    2    5    3    1  -1
## [5,]    3    3   59  -3   100    1    1  -4  -5    3
## [6,]    1    3    0    2    1   100    1    1    9  -4
## [7,]    1    3   -3    5    1    1   100   -3    9    1
## [8,]   -1   -3   -3    3   -4    1   -3   100   -2   -2
## [9,]    0    1   -2    1   -5    9    9   -2   100    2
## [10,]   0   -1    3   -1    3   -4    1   -2    2   100
```

Exemplo de LASSO (cont.)

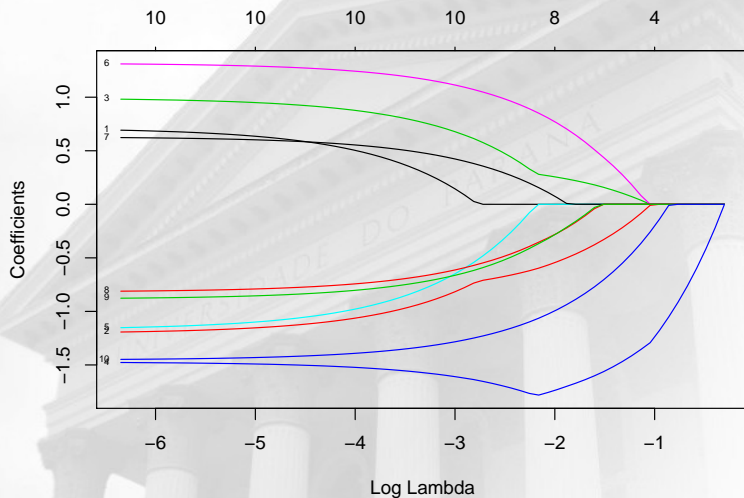
```
set.seed(2)
(b <- sample(0.5+(1:p)/p)*(rbinom(p, 1, 0.5)*2-1))
## [1] 0.7 -1.2 1.0 -1.5 -1.1 1.3 0.6 -0.8 -0.9 -1.4
s <- 0.5
set.seed(3)
y <- 10 + drop(x%*%b) + rnorm(n)*s

library(glmnet)

lasso.res <- glmnet(x, y)
```

Exemplo de LASSO (cont.)

```
plot(lasso.res, xvar='lambda', label=TRUE)
```



LASSO características

- ▶ A partir do momento que o coeficiente de x_1 zerou, a trajetória do coeficiente de x_2 altera
- ▶ A partir do momento que o coeficiente de x_5 zerou,
 - ▶ a trajetória dos coeficientes de x_3 e x_4 se alteram.
 - ▶ A partir do momento que o coeficiente de x_3 zera, a trajetória do coeficiente de x_4 altera
- ▶ A trajetória dos demais não são influenciadas pelo fato do coeficiente de qualquer das outras serem zerados