Aprendizado de Máquina

UFMG EST171 - 1ª Lista de exercícios

Eduardo E. R. Junior & Matheus H. Sales

12 de setembro de 2016

Exercício 1

> Baixe os dados worldDevelopmentIndicators.csv, que contém os dados do PIB per capita (X) e a expectativa de vida (Y) de diversos países. O objetivo é criar preditores de Y com base em X. Em aula vimos como isso pode ser feito através de polinômios. Aqui, faremos isso via expansões de Fourier.

Como são dados bidimensionais uma representação gráfica é realizada na Figure 1. Note que não há um padrão cíclico aparente que justifique a utilização de expansões via séries de Fourier, porém essas serão utilizadas para ilustração das técnicas apresentadas na disciplina.

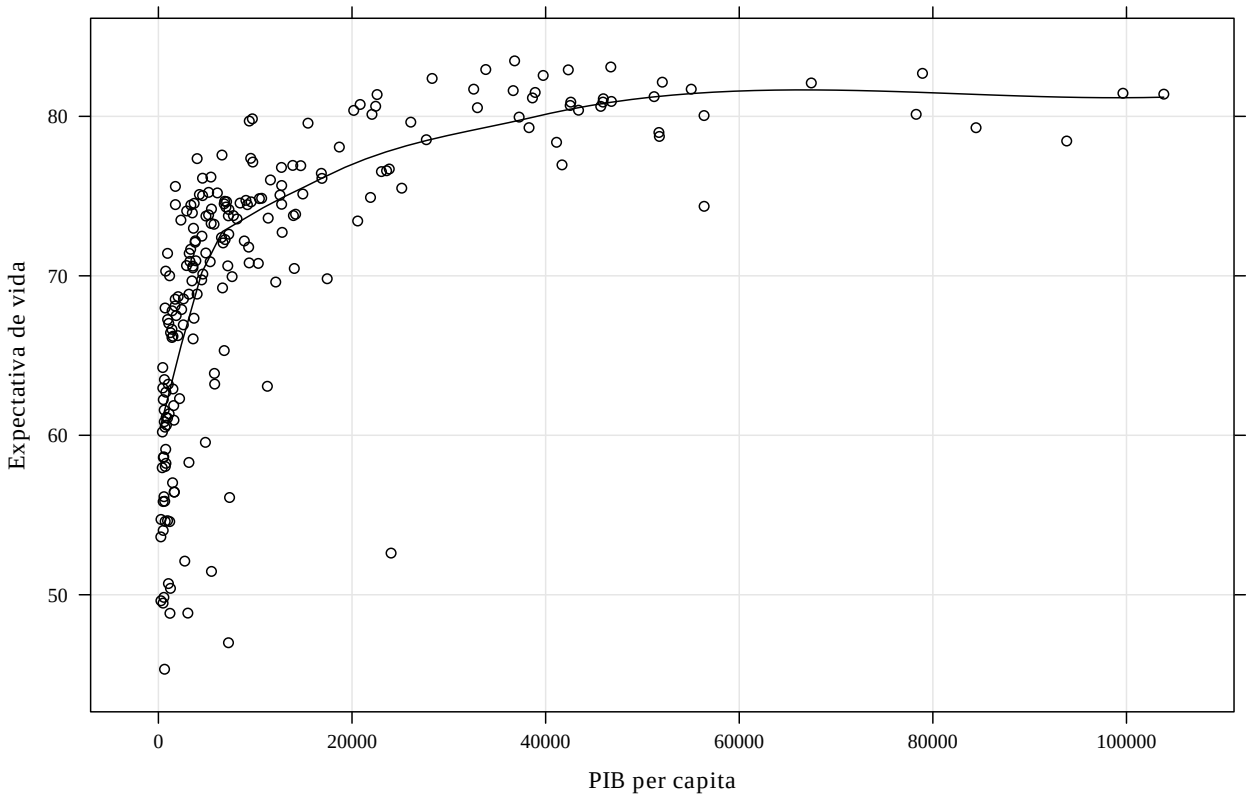

Figura 1: Dispersão do dados

---

a) *Normalize a covariável de modo que* $x \in (0, 1)$*. Para isso, faça* $x = \frac{x - xmin}{xmax - xmin}$*, onde* $xmin$ *e* $xmax$ *são os valores mínimo e máximo de x segundo a amostra usada.*

Após realizada a padronização da variável conforme indicação do exercício, apresenta-se seu comportamento via densidade estimada na Figure 2, note a forte assimetria da variável com quase 75% dos seus valores abaixo de 0.1.

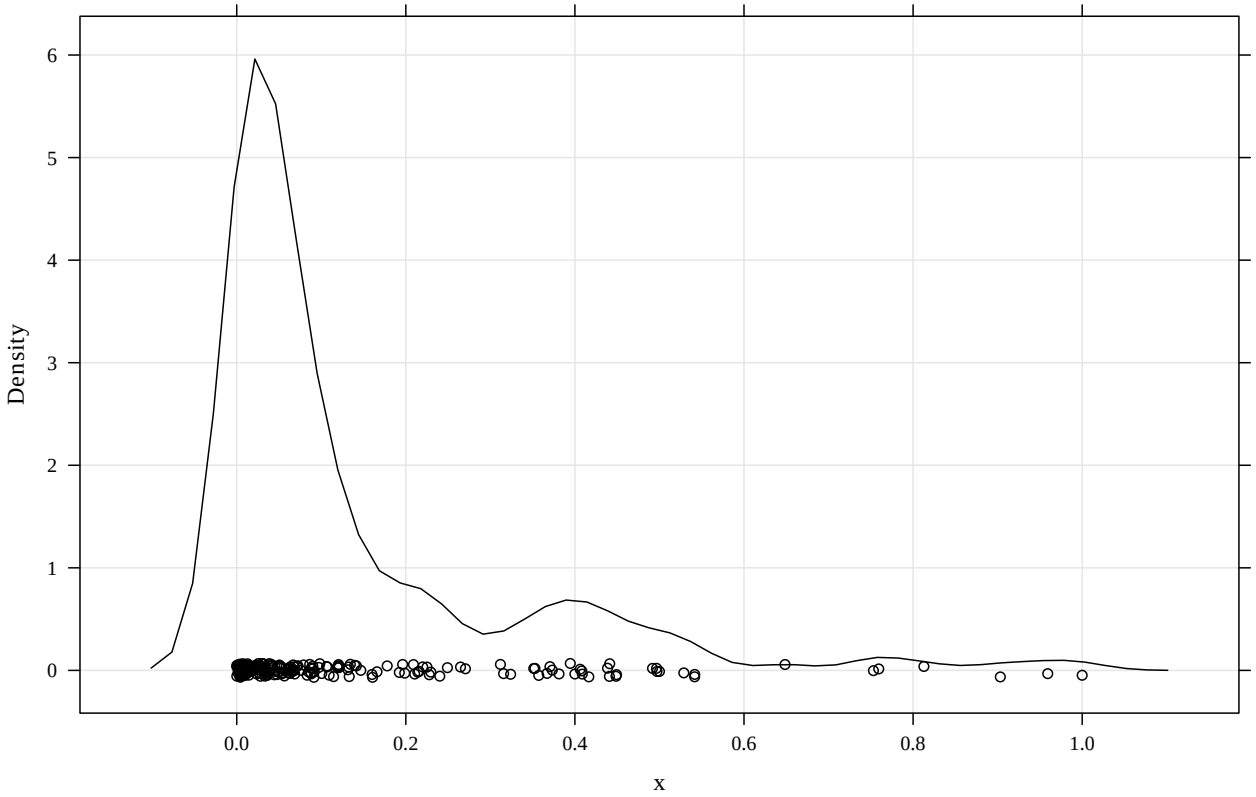

Figura 2: Densidade estimada do PIB per capita padronizado

---

b) *Usando o método dos mínimos quadrados e a validação cruzada do tipo leave-one-out, estime o erro quadrático médio das regressões*

$$g(x) = \sum_{i=1}^{p} \beta_{si} sin(2\pi i x) + \beta_{ci} cos(2\pi i x), \qquad para\ p = 1, 2, \ldots, 30$$

Para o ajuste e estimação do erro quadrático médio de tais regressões utilizou-se o software R com a rotina descrita abaixo:

1. Ajuste dos modelos, tendo os modelos armazenados outras medidas de qualidade podem ser extraídas.

```
X <- cbind()
models <- vector("list", 30)
v <- 2 * pi * dados$x

## Ajuste todos os modelos
for (p in 1:30) {
    m <- cbind(xs = sin(v * p), xc = cos(v * p))
    colnames(m) = paste0(colnames(m), p)
```

```
    X <- cbind(X, m)
    models[[p]] <- lm(y ~ X, data = dados)
}
```

2. Calcula o erro quadrático para cada observação retirada do ajuste e posteriormente predita pelos p modelos. Mantém apenas os erros quadráticos para avaliação de seu comportamento, uma vez que resumos, como a média podem ser obtidos facilmente. Os valores preditos de cada observação também são armazenados.

```
## Calcula o erro quadrático de cada observação para cada modelo via
## cross-validation leave-one-out
results <- lapply(models, function(modelo) {
    n <- nrow(modelo$model)
    eqs <- vector("numeric", length = n)
    pred <- vector("numeric", length = n)
    for (i in 1:n) {
        ## obs <- as.data.frame(modelo$model[i, ])
        obs <- modelo$model$y[i]
        ## pred[i] <- predict(mod, newdata = obs)
        mod <- update(modelo, data = modelo$model[-i, ])
        pred[i] <- cbind(1, modelo$model$X)[i, ] %*% coef(mod)
        eqs[i] <- (obs - pred[i])^2
    }
    list(eqs = eqs, pred = pred)
})

## Extraindo os erros quadráticos médios
eqms.mean <- sapply(results, function(x) mean(x[["eqs"]]))
```

Caso prefiro funções prontas, existe um pacote no R, `cvTools` que realiza a validação cruzada do tipo leave-one-out, além de outras estratégias.

```
##-----------------------------------------
## Pra quem gosta de pacotes ...
library(cvTools)
teste <- sapply(models, function(x) cvLm(x, cost = mspe, K = 211)$cv)
teste == eqms.mean
##-----------------------------------------
```

---

c) *Plote o gráfico do risco estimado vs p. Qual o valor de p escolhido? Denotaremos ele por* p_{esc}

Primeiramente para a escolha do p decidiu-se por avaliar o comportamento dos erros em cada modelo ajustado. Essa avaliação é exibida na Figure 3, onde apresenta-se os erros quadráticos para cada observação em cada modelo na escala logarítmica com a indicação da média, mediana e quartis. Note que o comportamento dos erros é bastante assimétrico, manifestando essa assimetria também na escala logarítmica. Além da assimetria muitos pontos discrepantes são observados, principalmente nos modelos mais complexos ($p > 12$). Devido a isso decidiu-se por apresentar, além do erro quadrático médio, erro quadrático mediano, pela mediana ser uma medida resumo menos sensível a dados extremos.

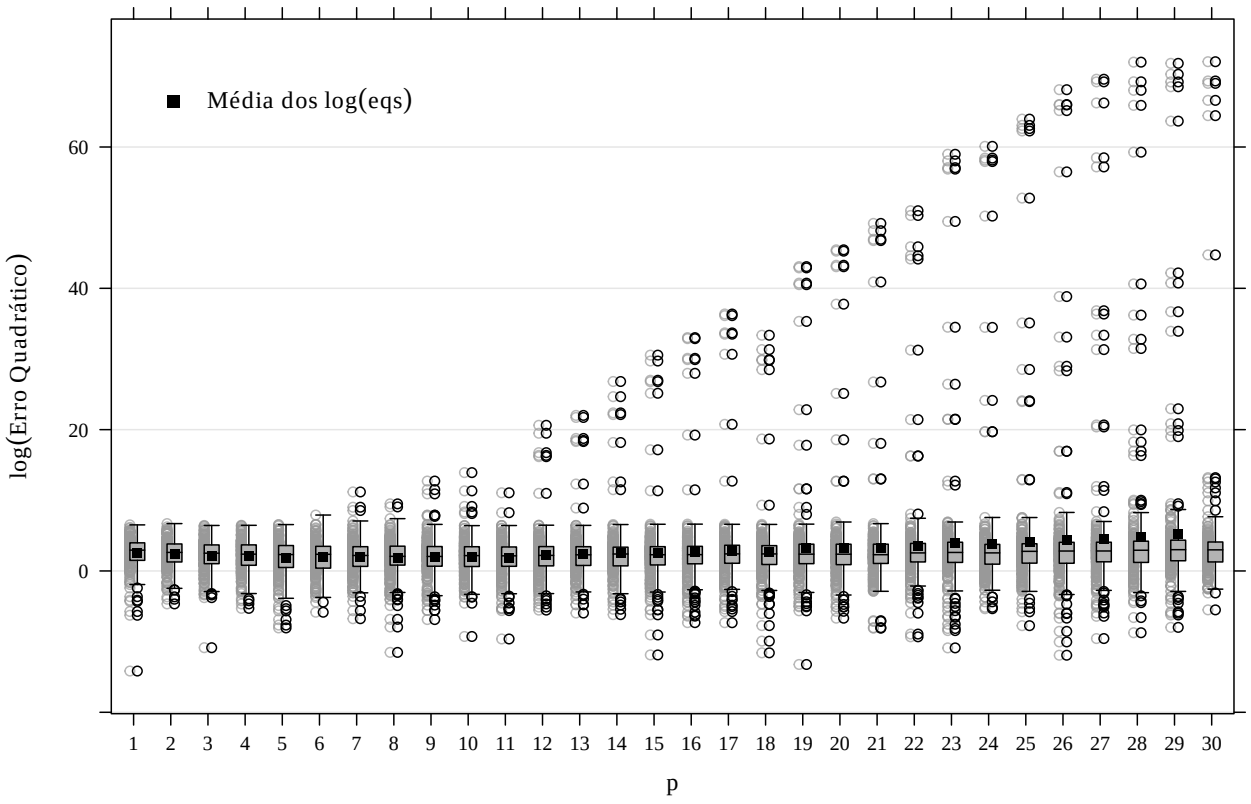

Figura 3: Distribuição empírica dos logarítmos dos erros quadráticos médios para cada um dos p modelos

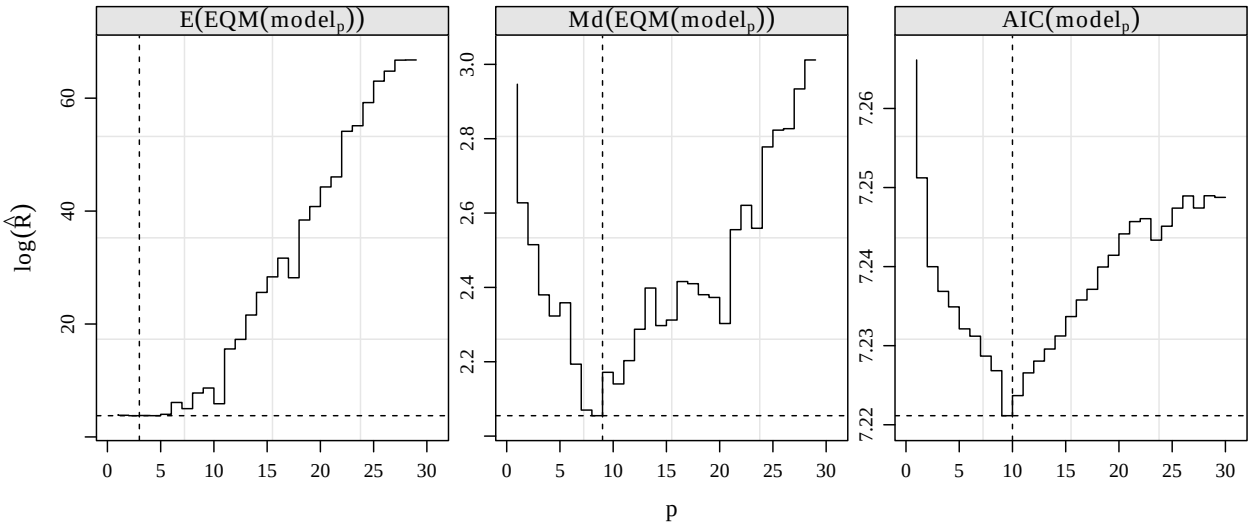

Figura 4: Medidas de qualidade de predição na escala logarítimica (linha pontilhada representa a indicação do melhor modelo)

Na Figure 4 são apresentadas as medidas de qualidade de predição e ajuste para todos os p modelos. São apresentados os erros quadráticos médios e medianos e também o AIC (Critério de Informação de Akaike) como uma medida mais estatística que mensura a parcimônia do modelo. As três medidas indicaram diferentes modelos sendo p = 3, 9, 10 para as medidas erro quadrático médio, erro quadrático mediano e

AIC, respectivamente. Desta forma, nas análises subsequentes serão apresentados os resultados para os modelos com p = 3, 9.

---

d) *Plote os das curvas ajustadas para p = 1 ; p = p_{esc} e p = 30 sob o gráfico de dispersão de X por Y. Qual curva parece mais razoável? Use um grid de valores entre 0 e 1 para isso. Como estes ajustes se comparam com o visto em aula via polinômios? Discuta.*

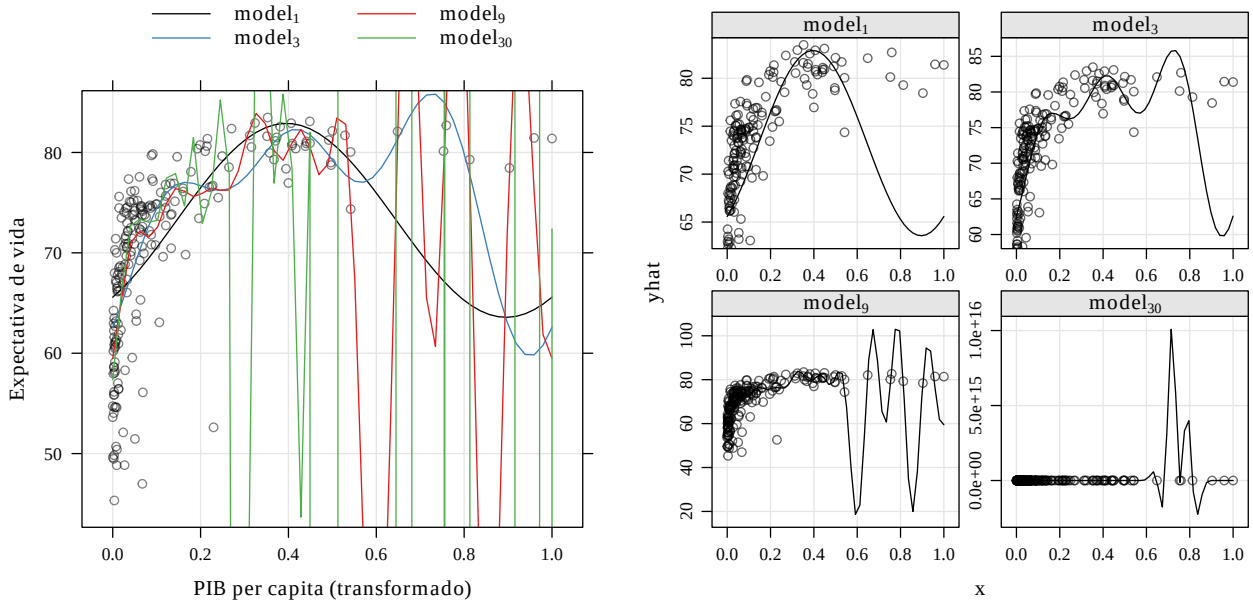

Figura 5: Curvas ajustadas. Conjuntamente no intervalo dos dados (esquerda) e individuais no interrvalo de predição (direita).

Séries de Fourier são comumente utilizadas na análise de séries temporais, pois tem flexibilidade para ajustar sazonalidade, sua aplicação aos dados indicados no exemplo parece não ser adequada conforme pode ser visto na Figure 5, em que os ajustes dos modelos com $p = 1, 3, 9, 30$ são exibidos. Como não é claro um comportamento cíclico nos dados, todos os modelos ajustam ciclos de forma incorreta, sendo nos modelos para $p = 9$ e 30 os ajustes mais discrepantes com ajuste muito incorreto, principalmente para intervalos com menos observações. Para $p = 1$ a imposição de forma da Série de Fourier produz uma curva fora do padrão dos dados para PIB's padronizados maiores que 0,5.

Assim como o ajuste por polinômios, bastante comum em modelos de regressão linear aplicados em problemas de predição, a forma do preditor linear se mantém quando não se tem muitas observações para ajuste, proporcionando assim, um péssimo poder preditivo para problemas em que a forma do preditor linear não é adequada, como é o caso deste exemplo.

---

e) *Plote o gráfico de valores preditos versus ajustados para p = 1; p = p_{esc} e p = 30 (não se esqueça de usar o leave-one-out para calcular os valores preditos! Caso contrário você terá problemas de overfitting novamente). Qual p parece ser o mais razoável?*

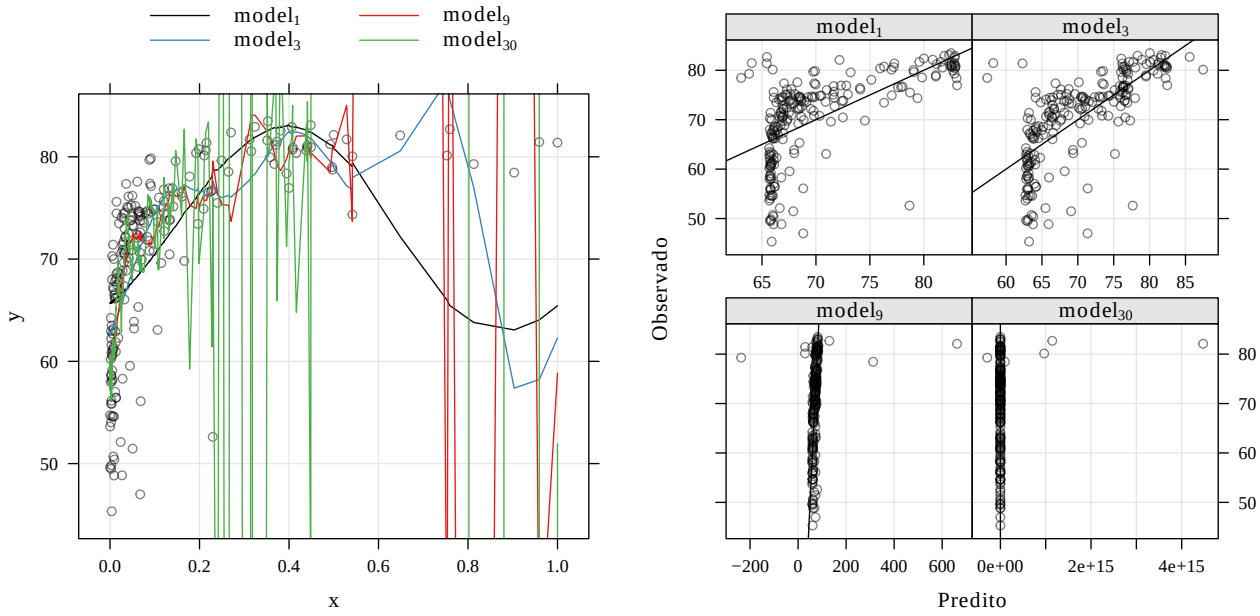

Figura 6: Valores observados versus valores preditos. Traço de valores preditos (esquerda) e observados versus preditos (direita)

Observa-se nos gráficos dos valores preditos versus valores observados, Figure 6 à esquerda, que os modelos mais complexos ($p = 9$ e $p = 30$) parecem produzir melhores resultados. Todavia, para alguns valores, cuja a predição é incorreta, a discrepância com relação ao valor observado é enorme, devido a imposição de sazonalidade do modelo. Isto posto, podemos interpretar estes bons resultados de predição para os modelos com $p = 9$ e $p = 30$ como sobreajuste, uma vez que a forma do modelo é incorreta para predição de novas observações, e o ajuste tentar interpolar as observações. Para os modelos com $p = 1$ e $p = 3$ observa-se claramente que há uma característica não explicada pelo modelo que prejudica as predições, novamente imposta pela forma das séries de Fourier.

---

f) *Quais vantagens e desvantagens de se usar validação cruzada do tipo leave-one-out versus o data-splitting?*

Para a avaliação dos modelos ajustados utilizou-se medidas da qualidade de predição baseadas em validação cruzada do tipo leave-one-out. Essa estratégia de validação cruzada é computacionalmente intensiva, pois cada modelo é reajustado n vezes (neste caso $n = 211$), tornando o procedimento mais lento. Porém ao utilizar a estratégia leave-one-out todos os dados são utilizados tanto na validação quanto no ajuste do modelo tornando os resultados gerais. Isso não ocorre quando utilizado estratégias do tipo data-splitting (holdoult ou k-fold), pois os resultados ficam condicionados a partição da base, ou seja, há a variabilidade nos resultados advinda do procedimento de partição, e.g. ao realizar novamente a avaliação dos modelos via data-splitting os resultados são, com alta probabilidade, distintos.

---

g) *Ajuste a regressão Lasso (Frequentista e Bayesiana) e discuta os resultados encontrados.*

Ajuste via penalização Lasso: Sob essa abordagem minimiza-se a função de soma de quadrados penalizada da forma

$$R(\beta) = \sum_{i=1}^{n}(y_i - \underline{x}_i^t\beta)^2 - \lambda \sum_{j=1}^{p} |\beta_j|$$

em que $\underline{x}_i^t = (x_{i1}, x_{i2}, \ldots, x_{ip})$ são as covariáveis de cada observação. A escolha do λ ótima, em geral, é realizada via validação cruzada.

Ajuste via especificação Bayesiana: Sob o paradigma Bayesiano temos a penalização da função de verossimilhança realizada por meio de prioris dos parâmetros que são assumidas distribuições *Double Exponential* (ou Laplace) de parâmetros 0 e λ.

$$Y \mid X \sim \text{Normal}(\underline{x}_i^t \beta, \sigma^2)$$
$$\beta_j \sim \text{Laplace}(0, \lambda), \quad j = 1, 2, \ldots, p$$

em que $\underline{x}_i^t = (x_{i1}, x_{i2}, \ldots, x_{ip})$ são as covariáveis de cada observação. Mantendo as inferências sob o paradigma bayesiano são dadas prioris para σ^2 e também, em segundo nível para λ. Assim o tunning do parâmetro λ é realizado via simulações MCMC, por exemplo.

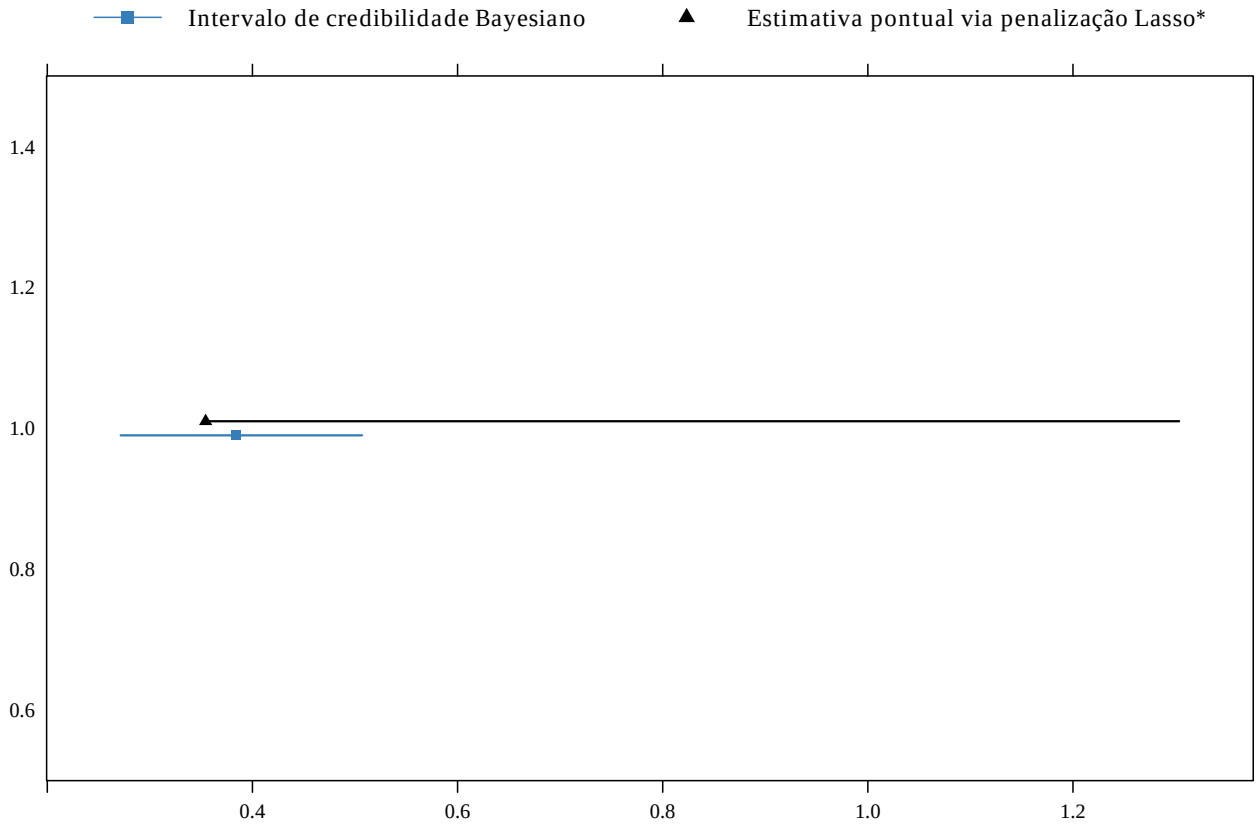

Figura 7: Estimativas do parâmetro lambda pela abordagem de validação cruzada e bayesiana

Na Figure 7 são exibidos as estimativas do parâmetro λ note que as estimativas pontuais foram próximas. Para os intervalos exibidos a comparação não é direta uma vez que na abordagem bayesiana os intervalos são HPD (Highest Posterior Density) e representam a credibilidade do valor e na abordagem Lasso o limite superior é dado pelo maior valor de λ testado que confere um erro quadrático médio menor que o limite superior do erro quadrático médio do λ ótimo.

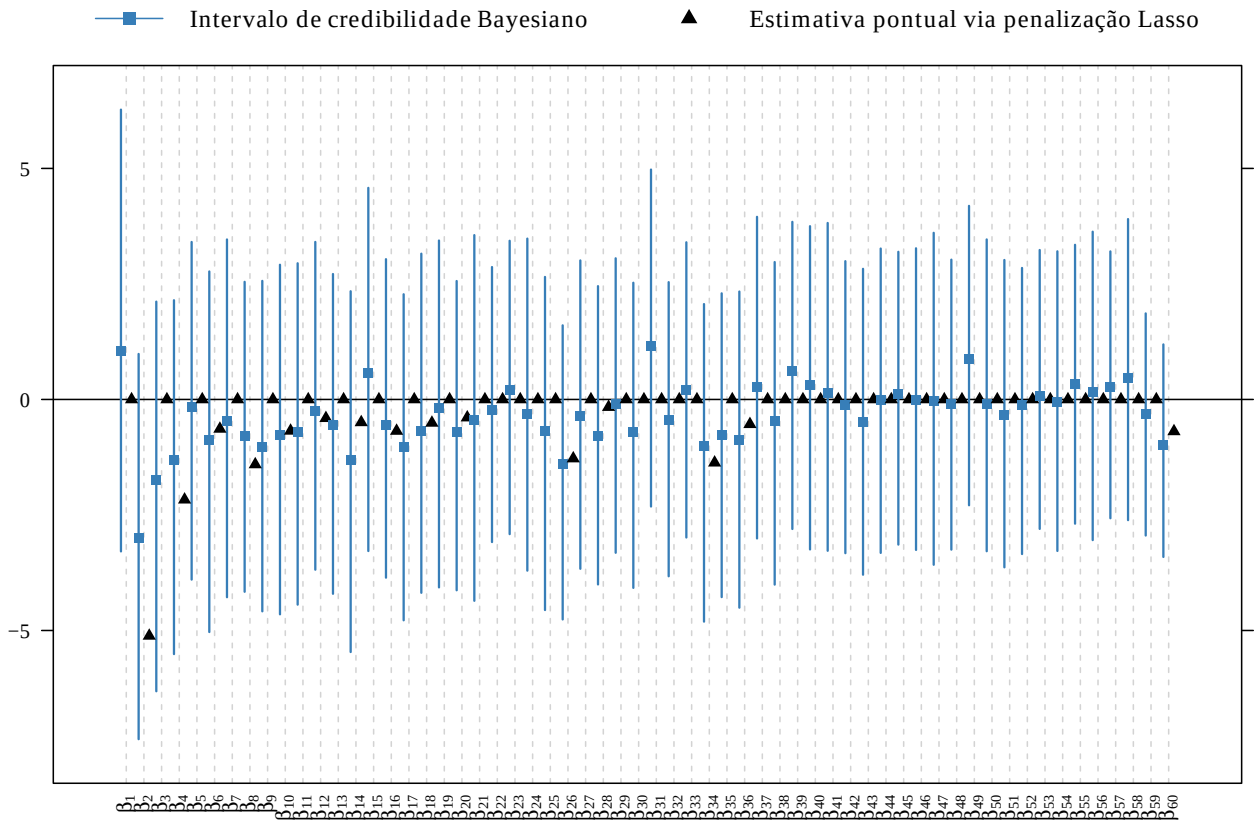

Figura 8: Coeficientes estimado pela abordagem de penalização Lasso frequentista e bayesiana

Na Figure 8 são exibidos os coeficientes estimados por ambas as abordagens com estimativas pontuais para a regressão sob penalização Lasso e intervalos de credibilidade para o modelo Bayesiano. Note que na abordagens Bayesiana todos os intervalos compreendem o valor 0 o que indica que nenhum coeficiente seria necessário em discordância com a regressão Lasso que indicou 14 coeficientes diferentes de zero.

A vantagem da abordagem Bayesiana é que há um modelo probabilístico adjacente ao método que fornece inferências mais completas, em contraste da abordagem frequentista em que o foco consiste somente em predição.

Exercício 2

Neste exercício você irá implementar algumas técnicas vistas em aula para o banco de dados das faces. O objetivo aqui é conseguir criar uma função que consiga predizer para onde uma pessoa está olhando com base em uma foto. Iremos aplicar o KNN para esses dados, assim como uma regressão linear. Como não é possível usar o método dos mínimos quadrados quando o número de covariáveis é maior que o número de observações, para esta segunda etapa iremos usar o lasso.

---

a) *Leia o banco dadosFacesAltaResolucao.txt. A primeira coluna deste banco contém a variável que indica a direção para a qual o indivíduo na imagem está olhando. As outras covariáveis contém os pixels relativos a essa imagem, que possui dimensão 64 por 64. Utilizando os comandos fornecidos, plote 5 imagens deste banco.*
Divida o conjunto fornecido em treinamento (aproximadamente 60% das observações), validação (aproximadamente 20% das observações) e teste (aproximadamente 20% das observações). Utilizaremos o conjunto de treinamento e validação para ajustar os modelos. O conjunto de teste será utilizado para testar sua performance.

A leitura do conjunto de dados para manipulação no software R é realizada conforme código abaixo:

```
## Leitura dos dados
dados <- read.table("./data/dadosFacesAltaResolucao.txt",
                    header = TRUE, sep = " ")

## Verificando
## str(dados)
dim(dados)
```

```
## [1]  698 4097
```

Na Figure 9 são exibidas as seis primeiras imagens do conjunto da dados para ilustrar como o conjunto é constituído.

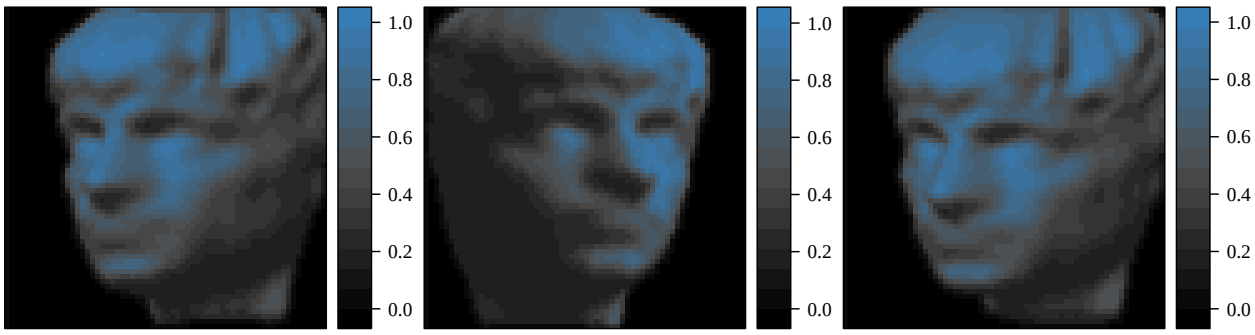

Figura 1 (Direção −32.119) Figura 2 (Direção 29.044) Figura 3 (Direção −28.113)

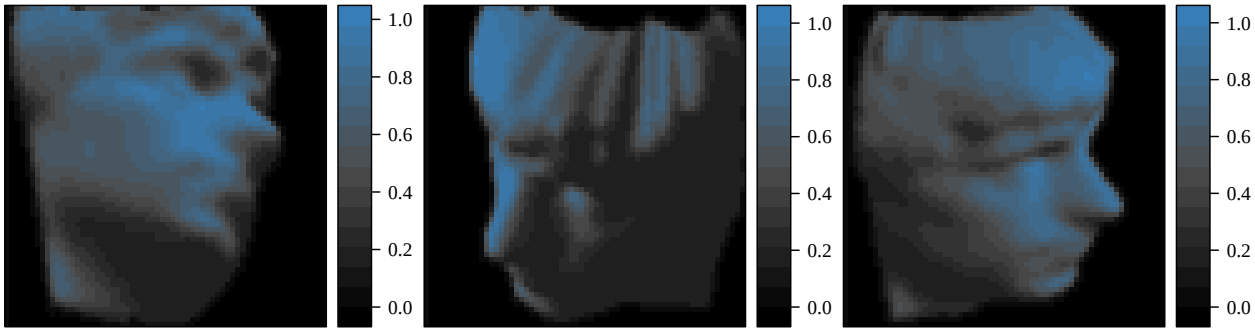

Figura 4 (Direção 59.263) Figura 5 (Direção −42.01) Figura 6 (Direção 61.916)

Figura 9: Seis primeiras imagens representadas no conjunto de dados

Para a partição do conjunto de dados foi implementada uma rotina que realiza a divisão da base conforme proporções informadas, a implementação pode ser vista no complemento online do trabalho.

```
## Particionando o conjunto de dados
dasplit <- mysplit(dados, percent = c(0.6, 0.2, 0.2),
                   seed = 1994)

## Número de observações em cada partição
sapply(dasplit, dim)
```

```
##      [,1] [,2] [,3]
## [1,]  418  140  140
## [2,] 4097 4097 4097
```

```
## Atribuindo as partições em objetos de nome sujestivo
da.train <- dasplit[[1]]
da.valid <- dasplit[[2]]
da.teste <- dasplit[[3]]
```

---

b) *Qual o número de observações? Qual o número de covariáveis? O que representa cada covariável?*

Neste conjunto de dados são 698 observações, que representam imagens, faces humanas, com a indicação da direção para a qual a face está virada e da intensidade da coloração em cada pixel. A imagem tem resolução 64px *x* 64px, portanto têm-se 4096 covariáveis que representam a intensidade de coloração em cada pixel.

c) *Para cada observação do conjunto de teste, calcule o estimador da função de regressão $r()$ dado pelo método dos k vizinhos mais próximos com $k = 5$. Você pode usar as funções vistas em aula.*

Para esta tarefa utilizou-se o pacote FNN (Fast Nearest Neighbor) do R. Na Figure 10 são exibidos as predição provenientes da aplicação do método KNN com cinco vizinhos. Vale destacar que os valores exibidos na figura são predições da base de teste utilizando para treinamento o conjunto de treino e validação.

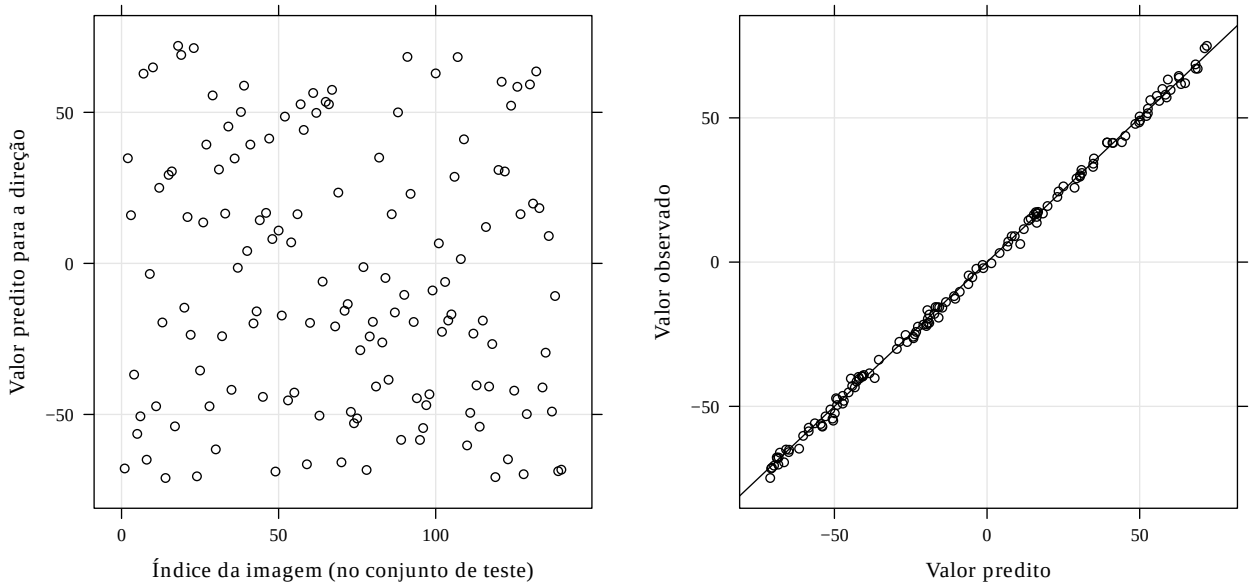

Figura 10: Valores preditos pelo método KNN com 5 vizinhos. Valor preditos na ordem do conjunto de teste (esquerda) e preditos versus observados (direita)

d) *Utilize validação cruzada (data splitting) para escolher o melhor k. Plote k vs Risco estimado.*

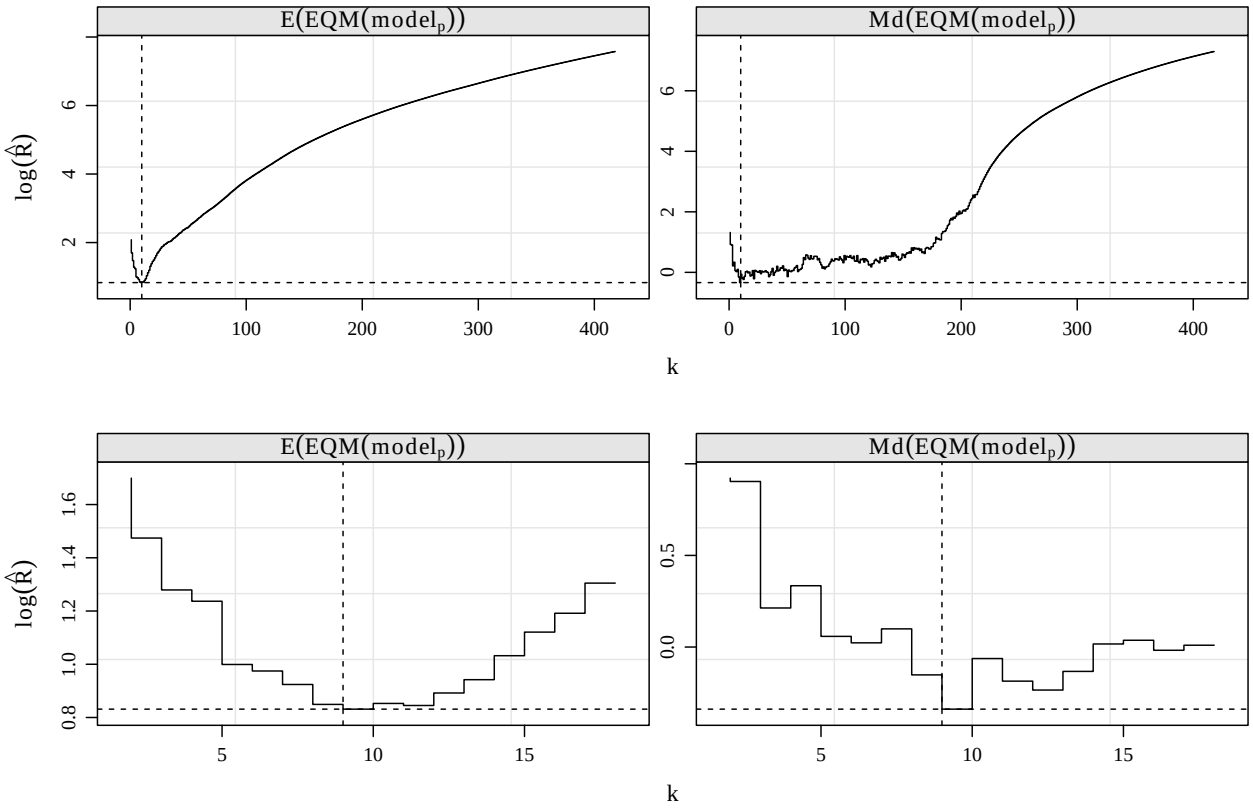

Figura 11: Erros quadráticos médios e medianos na escala logarítimica (linha pontilhada representa a indicação do melhor modelo). Para todos o vizinhos (acima) e apenas para os k's próximos do ótimo (abaixo).

---

e) *Utilizando o conjunto de teste, estime o risco do KNN para o melhor k. Plote os valores preditos versus os valores observados para o conjunto de teste. Inclua a reta identidade.*

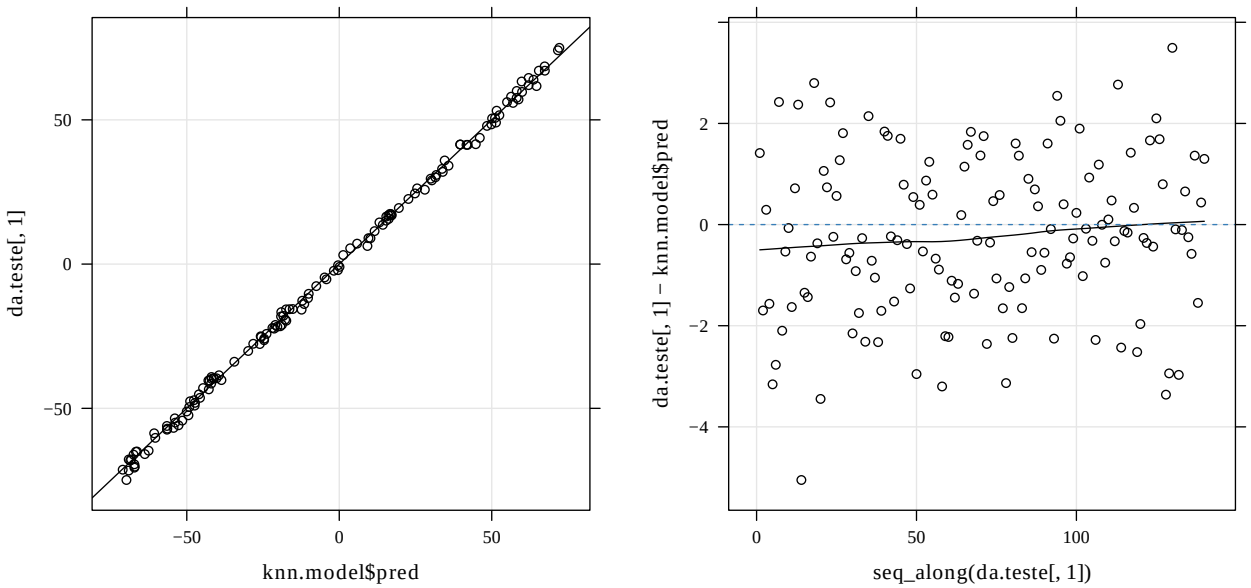

Figura 12: Valores preditos pelo método KNN com 10 vizinhos. Preditos versus observados (esquerda) e desvios (direita).

---

f) *Ajuste uma regressão linear para os dados usando o conjunto de treinamento mais o de validação via lasso (lembre-se que a função que ajusta o lasso no R já faz validação cruzada automaticamente: ao contrário do KNN, neste caso não é necessário separar os dados em treinamento e validação). Qual o lambda escolhido? Plote λ vs Risco estimado.*

Para ajuste da regressão linear sob penalização do tipo Lasso utilizou-se a validação cruzada do tipo leave-one-out para estudo do parâmetro λ de penalização. Os resultados da validação cruzada são exibidos na **??** onde têm-se os erros quadráticos médios em função dos λ's na escala logarítmica. O λ que minimiza o erro quadrático médio é 0.339318 e o maior λ, cujo erro quadrático médio é menor que o limite superior do erro sob o λ ótimo é 0.4699169.

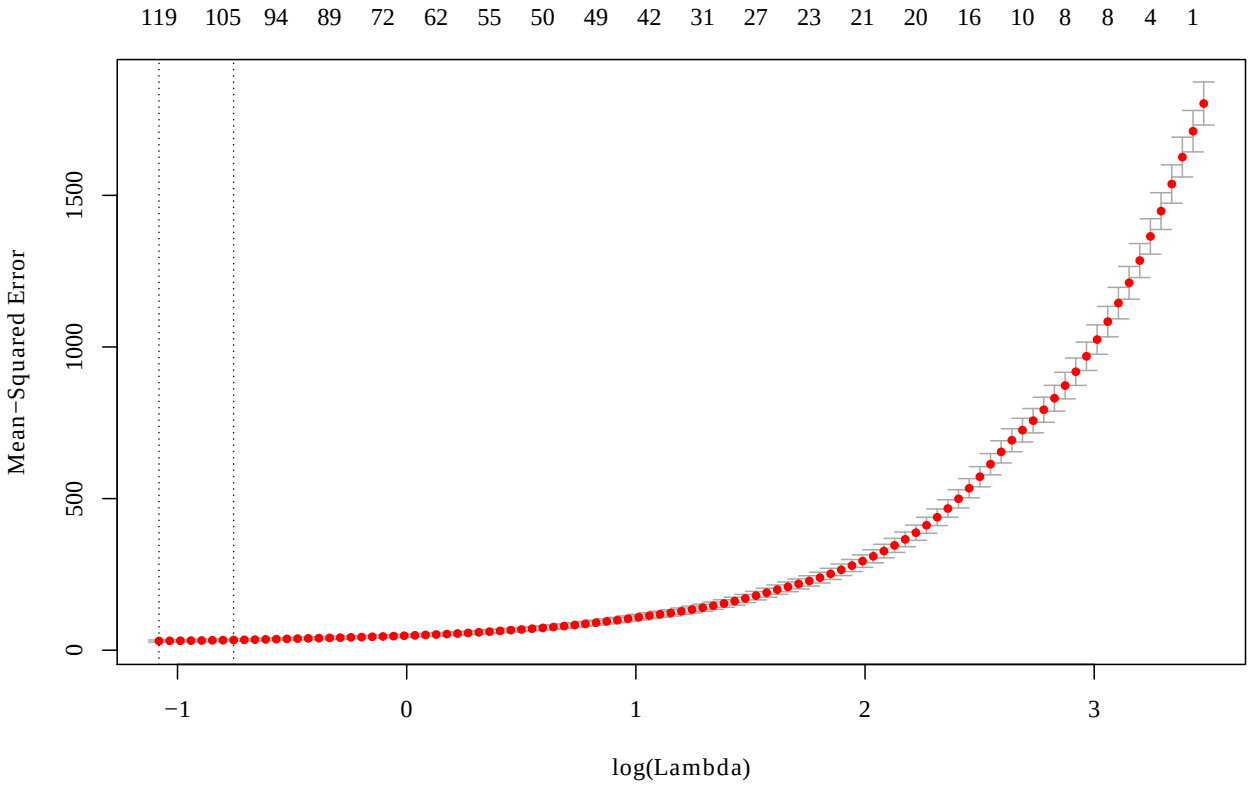

Figura 13: Lambdas versus erros quadráticos médios

---

g) *Utilizando o conjunto de teste, estime o risco do lasso para o melhor λ. Plote os valores preditos versus os valores observados para o conjunto de teste. Inclua a reta identidade.*

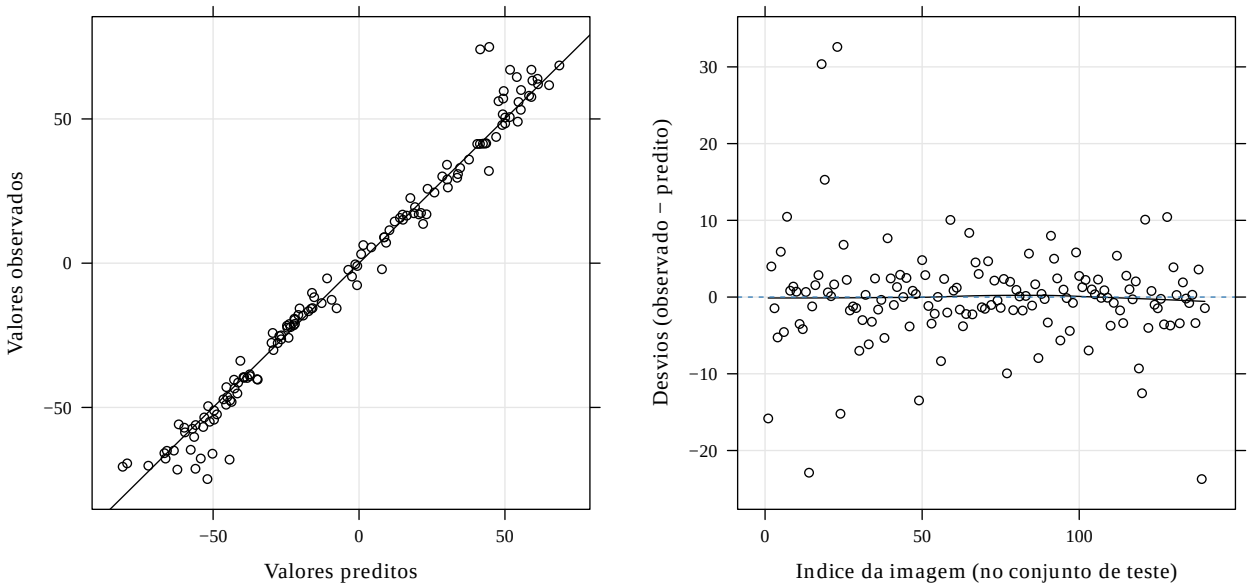

Figura 14: Valores preditos pelo método Lasso. Preditos versus observados (esquerda) e desvios (direita).

---

h) *Quantos coeficientes foram estimados como sendo zero?*

Via penalização Lasso com λ= 0.339318 foram apenas 120 coeficientes de 4096, cujo valor não foi zerado.

---

i) *Qual modelo teve melhores resultados: regressão linear via lasso ou KNN?*

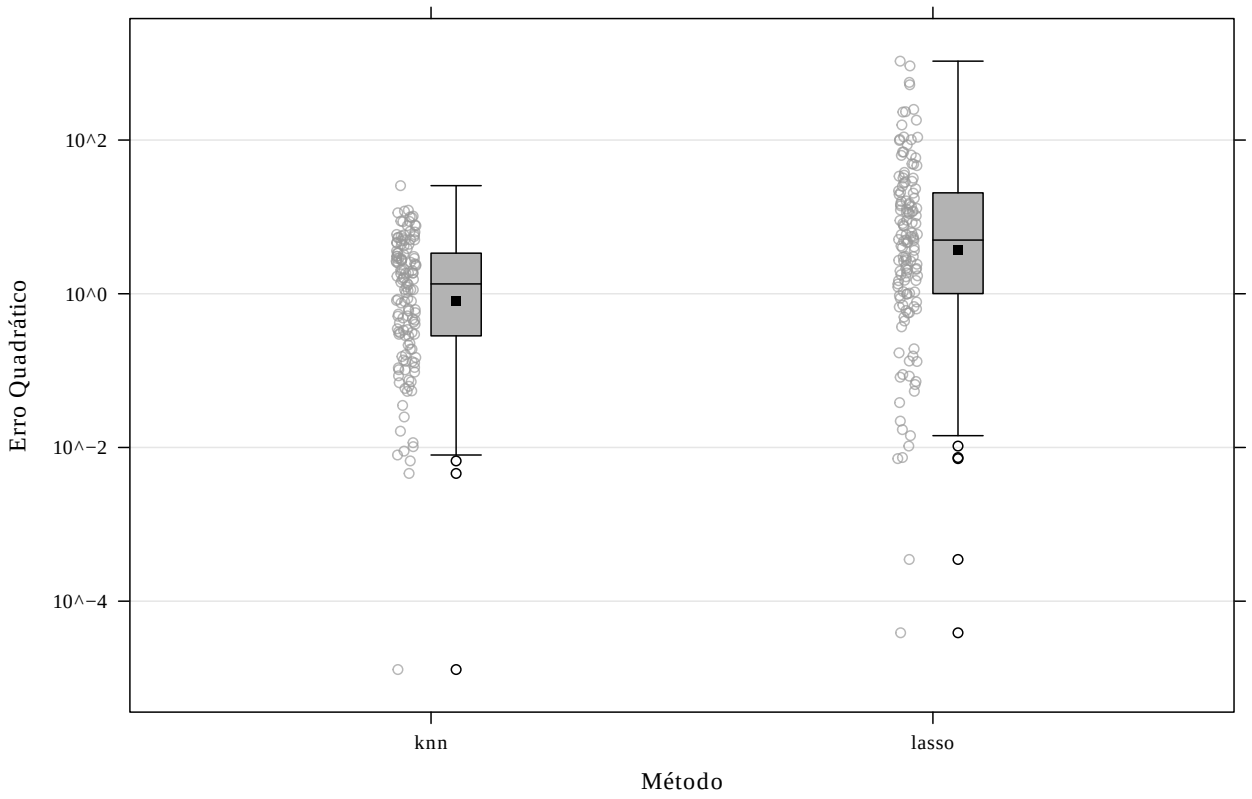

Figura 15: Comparação dos métodos preditivos KNN e regressão Lasso via erros quadráticos

Ambos os métodos são meramente preditivos, ou seja, são algoritmos numéricos em que não se faz inferências a não ser a predição, pois não assume-se modelo, verossimilhança, etc. Portanto, para a comparação dos métodos utilizou-se o erro quadrático de predição no conjunto de teste. Os resultados para comparação são exibidos na Figure 15. Note que os erros são, em média, menores para o KNN favorecendo esse método em detrimento da regressão Lasso. Outra característica observada na figura que favorece o KNN é a dispersão dos erros, que é menor para o KNN. Isso mostra que o método KNN, aplicado a esse conjunto de dados, proporcionou melhores resultados que o método de regressão sob penalização Lasso.

Material suplementar

Todos os códigos (para manipulação, ajustes e gráficos) exibidos neste trabalho estão disponíveis no endereço https://jreduardo.github.io/est171-ml/.