1 Exercício 1

Considere o banco `houses_to_rent_v2` que contém o valor (em reais) do aluguel de imóveis no Brasil. Você pode trabalhar apenas com os imóveis localizados em São Paulo, Rio de Janeiro e Belo Horizonte.

```
library(readr)
library(dplyr)
library(tidyverse)
library(magrittr)
library(ggplot2)
library(glmnet)
library(xtable)

colnames <- c("city", "area", "rooms", "bathroom", "park_spaces", "floor", "animal",
"furniture", "rent_am")
data <- read_csv("D:/__Mestrado/Lista 1/houses_to_rent_v2.csv",
                 na = "-", skip=1,
                 col_types = "cdddddffd",
                 col_names = colnames)
#Filtrar as cidades de São Paulo, Rio de Janeiro e Belo Horizonte
data %<>% filter(city %in% c("São Paulo", "Rio de Janeiro", "Belo Horizonte"))
data$city <- as.factor(data$city)
```

(a) Divida o conjunto de dados em treinamento e teste. Explique como decidiu qual porcentagem deixar para cada um.

A princípio, foi escolhida a divisão de 80% e 20%. Entretanto, posteriormente, foi feita uma análise com os intervalos de confiança a fim de entender se essa porcentagem escolhida está coerente com o intervalo almejado.

A Tabela 1 apresenta os intervalos de confiança para três divisões quando aplicadas sobre o banco todo, pode-se concluir que a divisão escolhida é adequada ao banco de dados. Isso porque por mais que a amplitude seja menor na divisão $70\% - 30\%$, obtemo um risco estimado maior. Decidimos então por continuar com a escolha da divisão entre treino e teste de $80\% - 20\%$.

Tabela 1: IC risco para diferentes porcentagens

Divisão	EQM	Erro padrão	IC	Amplitude do IC
$70\% - 30\%$	8.555.619	57,61	7.823.215 - 9.288.023	1.464.809
$80\% - 20\%$	6.607.061	61,06	5.679.209 - 7.534.912	1.855.703
$90\% - 10\%$	5.863.003	82,76	4.937.471 - 6.788.536	1.851.065

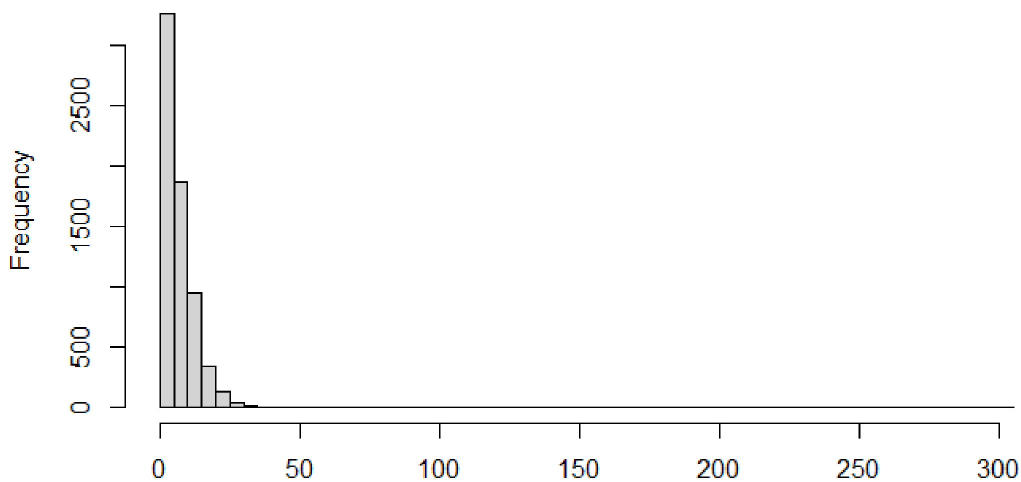
```
set.seed(57)
#Criando vetor para data splitting
data_split <- sample(c("Treino", "Teste"),
                     size = nrow(data),
                     prob = c(0.8, 0.2),
                     replace = TRUE)
```

```
table(data_split)
# Teste Treino
# 1772   6874
data_split70_30 <- sample(c("Treino", "Teste"),
                     size = nrow(data),
                     prob = c(0.7, 0.3),
                     replace = TRUE)
table(data_split70_30)
# Teste Treino
# 2578   6068
data_split90_10 <- sample(c("Treino", "Teste"),
                          size = nrow(data),
                          prob = c(0.9, 0.1),
                          replace = TRUE)
table(data_split90_10)
# Teste Treino
# 856   7790

#) Ajustando o LASSO - 70%-30%
fitLinear.cv70_30 <- cv.glmnet(x = X[data_split70_30 == "Treino",],
                          y = y[data_split70_30 == "Treino"],
                          alpha = 1)
#Ajustando o lasso com o lambda obtido pelo CV
lasso70_30 = glmnet(x = X[data_split70_30 == "Treino",],
               y = y[data_split70_30 == "Treino"],
               alpha = 1, lambda = fitLinear.cv70_30$lambda.min)
pred70_30 <- predict(lasso70_30, newx =  X[data_split70_30 == "Teste", ])
mse_lasso70_30 = mean((y[data_split70_30=="Teste"]-pred70_30)^2)
erro_padrao_lasso70_30 = sqrt(mse_lasso70_30/length(y[data_split70_30=="Teste"]))

#) Ajustando o LASSO - 90%-10%
fitLinear.cv90_10 <- cv.glmnet(x = X[data_split90_10 == "Treino",],
                               y = y[data_split90_10 == "Treino"],
                               alpha = 1)
#Ajustando o lasso com o lambda obtido pelo CV
lasso90_10 = glmnet(x = X[data_split90_10 == "Treino",],
                    y = y[data_split90_10 == "Treino"],
                    alpha = 1, lambda = fitLinear.cv90_10$lambda.min)
pred90_10 <- predict(lasso90_10, newx =  X[data_split90_10 == "Teste", ])
mse_lasso90_10 = mean((y[data_split90_10=="Teste"]-pred90_10)^2)
erro_padrao_lasso90_10 = sqrt(mse_lasso90_10/length(y[data_split90_10=="Teste"]))

####IC
IC_Risco_70_30 = IC_risco(m = 2578,yteste = y[data_split70_30=="Teste"], ypredito = pred70_
IC_Risco_80_20 = IC_risco(yteste = y[data_split=="Teste"], ypredito = pred)
IC_Risco_90_10 = IC_risco(m = 856,yteste = y[data_split90_10=="Teste"], ypredito = pred90_
rbind(unlist(IC_Risco_70_30), unlist(IC_Risco_80_20), unlist(IC_Risco_90_10))
```

(b) Utilizando o conjunto de treinamento, ajuste uma regressão (i) via mínimos quadrados, (ii) via lasso (usando validação-cruzada no treinamento para escolher λ) e (iii) Regressão ridge. Qual o melhor valor de λ encontrado para o lasso?

Tendo em vista que a variável *floor* apresentava o caracter - para indicar se o imóvel era uma casa, plotou-se o histograma dessa variável, a fim de entender sua distribuição, e este é apresentado pela figura 1.

Figura 1: Histograma da variável *floor*

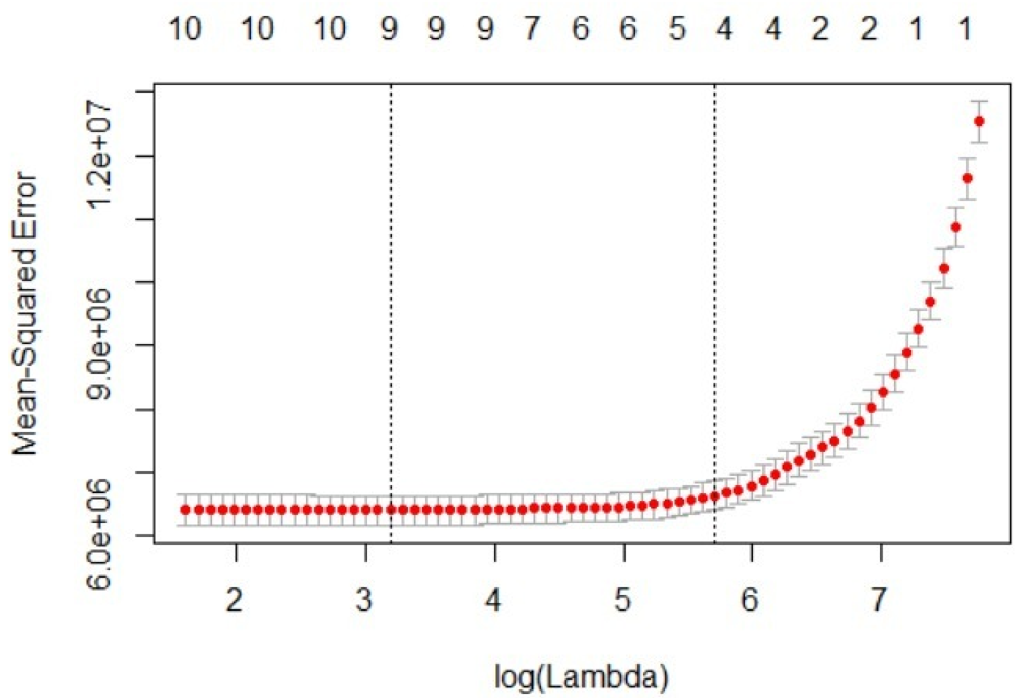



```
data$floor %>% hist(., breaks = 50)
data$floor %>% as.factor() %>% summary()
#Temos 2063 casas (NA)
#aptos com até 4 andares (não necessita de elevador por lei)
#aptos com 5 ou mais andares (necessita de elevador por lei)
#Criando nova coluna para essa nova divisão
data %<>% mutate(floor2 = data$floor)
data$floor2[data$floor < 5] <- "apto_4"
data$floor2[data$floor >= 5] <- "apto_5+"
data$floor2[data$floor %in% NA] <- "casa"
data$floor2 %<>% as.factor()
summary(data$floor2)
# apto_4   apto_5+    casa
# 2765     3818       2063
```

Uma vez que a maior parte das observações ou eram apartamentos cujos prédios apresentavam menos do que 6 andares ou eram casas, criou-se uma nova variável categórica *floor2*, com os valores *casa*, quando o imóvel fosse uma casa; *apto_4*, para os apartamentos em prédios com menos de cinco andares; *apto_5+*, para os apartamentos em prédio com cinco ou mais andares, optou-se por essa divisão pelo fato de que é obrigatório, por lei, que prédios com 5 ou mais andares tenham elevador. Ademais, as variáveis *animal* e *furniture* foram transformadas em fator.

Feitas essas mudanças, foram estimadas as três regressões, sendo que, para o lasso, o melhor valor de λ, encontrado pela validação-cruzada no conjunto de treinamento, foi o de 24,41898. A figura 2 apresenta o processo de escolha desse *tuning parameter*.

Figura 2: Processo de escolha de λ via validação cruzada




```
#Criando a matriz de planejamento X
X <- data[,c(2:5,7,8,10,1)]
X = model.matrix(~.,data = X)
X <- X[,-1]
#Criando o vetor da variável resposta
y = data %>% dplyr::select(rent_am) %>% as.matrix()

#i) Via mínimos quadrados
mq = glmnet(x = X[data_split == "Treino",],
            y = y[data_split == "Treino"],
            alpha =0, lambda = 0)
y_pred_mq = predict(mq, newx =  X[data_split == "Teste", ])
mse_mq = mean((y[data_split=="Teste"]-y_pred_mq)^2)
erro_padrao_mq = sqrt(mse_mq/length(y[data_split=="Teste"]))

#ii) Ajustando o LASSO
fitLinear.cv <- cv.glmnet(x = X[data_split == "Treino",],
                          y = y[data_split == "Treino"],
                          alpha = 1)
#Lambda mínimo pelo CV
lambda = fitLinear.cv$lambda.min
lambda #Melhor valor encontrado para lambda
```

```
plot(fitLinear.cv)
#Ajustando o lasso com o lambda obtido pelo CV
lasso = glmnet(x = X[data_split == "Treino",],
               y = y[data_split == "Treino"],
               alpha = 1, lambda = lambda)
coefficients(lasso)
pred <- predict(lasso, newx =  X[data_split == "Teste", ])

mse_lasso = mean((y[data_split=="Teste"]-pred)^2)
erro_padrao_lasso = sqrt(mse_lasso/length(y[data_split=="Teste"]))

#iii) Via regressão Ridge
cv_ridge <- cv.glmnet(x = X[data_split == "Treino",],
                      y = y[data_split == "Treino"],
                      alpha = 0)
ajuste_ridge <- glmnet(x = X[data_split == "Treino",],
                       y = y[data_split == "Treino"],
                       alpha = 0)
round(coefficients(ajuste_ridge, s = cv_ridge$lambda.min), 4)
predito_ridge <- predict(ajuste_ridge,
                         s = cv_ridge$lambda.min,
                         newx = X[data_split == "Teste", ])
mse_ridge = mean((y[data_split=="Teste"]-predito_ridge)^2)
erro_padrao_ridge = sqrt(mse_ridge/length(y[data_split=="Teste"]))
```

c) Qual dos métodos acima apresentou melhores resultados? Responda essa pergunta utilizando o conjunto de teste e o melhor valor de λ encontrado. Inclua os intervalos de confiança para o risco preditivo nos seus resultados.

A tabela 2 apresenta os valores de algumas métricas a fim de comparar o desempenho preditivo dos três tipos de métodos.

Tabela 2: Desempenho Preditivo

Modelo	EQM	Erro padrão	IC	Amplitude do IC
Mínimos quadrados	6.592.061	60,99282	5.667.292 - 7.516.829	1.849.537
Lasso	6.607.061	61,06218	5.679.209 - 7.534.912	1.855.703
Ridge	6.576.622	60,92136	5.648.370 - 7.504.875	1.856.504

É possível perceber que os erros padrão dos três tipos de modelo foram muito próximos, nesse sentido, é difícil afirmar qual dos modelos teve desempenho preditivo melhor, até porque tanto o EQM quanto a amplitude dos intervalos de confiança foram altos para todos os três tipos de regressão. Uma das possíveis razões para que o EQM de todos os modelos estimados tenha sido alto é a pequena quantidade de covariáveis disponívies, isso porque um dos componentes do risco esperado é a variância intrínseca da variável resposta, e esta só pode ser reduzida se aumentada a quantidade de covariáveis observadas. Embora a regressão ridge tenha apresentado o menor valor para o EQM e para o erro padrão, entende-se que o lasso seja preferível, uma vez que este apresenta a vantagem de não utilizar uma das covariáveis.

A tabela 3 apresenta os coeficientes estimados por cada um dos modelos.

Tabela 3: Coeficientes estimados

Variável	Mínimos Quadrados	Lasso	Ridge
Intercepto	162.1321	419.3566	343.3912
area	0.0818	0.0506	0.1024
rooms	406.9624	377.3532	444.6198
bathroom	1005.7621	1016.1796	909.2152
park_spaces	430.5769	421.3117	433.2273
animalnot acept	77.1648	-	51.8355
furniturenot furnished	-1365.2272	-1318.1696	-1296.1311
floor2apto_5+	420.7835	346.7505	413.6254
floor2casa	238.1823	137.2758	231.7944
cityRio de Janeiro	756.4777	529.1204	580.1689
citySão Paulo	892.7128	738.1093	787.9075

```
### IC para o Risco ###
IC_risco <- function(m = 1772, yteste, ypredito){
  R_hat = mean((yteste - ypredito)^2)
  W_k = (yteste - ypredito)^2
  W_medio = mean(W_k)
  sigma2_hat = mean((W_k - W_medio)^2)
  l = 1.96 * sqrt(sigma2_hat/m)
  LI = R_hat - l
  LS = R_hat + l
  return(list(R_hat = R_hat, sigma2_hat = sigma2_hat,
  amplitude = 2*l, LI = LI, LS = LS))
}
IC_Risco_MQ = IC_risco(yteste = y[data_split=="Teste"], ypredito = y_pred_mq)
IC_Risco_Lasso = IC_risco(yteste = y[data_split=="Teste"], ypredito = pred)
IC_Risco_Ridge = IC_risco(yteste = y[data_split=="Teste"], ypredito = predito_ridge)
rbind(unlist(IC_Risco_MQ), unlist(IC_Risco_Lasso), unlist(IC_Risco_Ridge))
coef <- cbind(round(coefficients(mq), 4),
      round(coefficients(lasso),4),
      round(coefficients(ajuste_ridge, s = cv_ridge$lambda.min), 4))
```

d) Interprete os resultados do melhor modelo encontrado (via coeficientes). Ele faz sentido? Analisando os coeficientes do lasso, temos que:

- Intercepto: Estima-se que o valor esperado do aluguel assumindo que todas as covariáveis sejam iguais a zero é de R$ 419,36, o que neste estudo não faz sentido para algumas covariáveis como a área do imóvel.

- area: Estima-se que o valor esperado para o aluguel aumente em R$ 0,05 quando se aumenta em uma unidade a área do imóvel, mantidas constantes na média as demais cováriaveis.

- rooms: Estima-se que o valor esperado para o aluguel aumente em R$ 377,35 quando se aumenta em uma unidade o número de quartos do imóvel, mantidas constantes na média as demais cováriaveis.

- bathroom: Estima-se que o valor esperado para o aluguel aumente em R$ 1.016,18 quando se aumenta em uma unidade o número de banheiros do imóvel, mantidas constantes na média as demais covári aveis.

- park_spaces: Estima-se que o valor esperado para o aluguel aumente em R$ 421,31 quando se aumenta em uma unidade o número de vagas de estacionamento do imóvel, mantidas constantes na média as demais covári aveis.

- furniture (not furnished): Estima-se que o valor esperado para o aluguel em um imóvel não mobiliado seja, em média, R$ 1.318,17 menor do que o valor de um imóvel mobiliado.

- floor2apto_5+ : Estima-se que o valor esperado para o aluguel em um imóvel de um prédio com 5 andares ou mais seja, em média, R$ 346,75 maior do que o valor de um imóvel em um prédio com menos de 5 andares.

- floor2casa : Estima-se que o valor esperado para o aluguel de uma casa seja, em média, R$ 137,28 maior do que o valor de um imóvel em um prédio com menos de 5 andares.

- cityRio de Janeiro: Estima-se que o valor esperado para o aluguel em um imóvel no Rio de Janeiro seja, em média, R$ 529,12 maior do que o valor de um imóvel em Belo Horizonte.

- citySão Paulo: Estima-se que o valor esperado para o aluguel em um imóvel em São Paulo seja, em média, R$ 738,11 maior do que o valor de um imóvel em Belo Horizonte.

  Podemos perceber que a interpretação dos coeficientes faz sentido para o ajuste em foco. Espera-se que imóveis maiores, com mais cômodos, banheiros, vagas na garagem ou mobiliados tenham um valor de aluguel mais alto. Percebemos, ainda, que o aluguel de apartamentos em prédios com 5 andares ou mais é maior em média do que aqueles com até 4 andares. Geralmente isso se dá porque prédios residenciais maiores costumam também ter uma maior estrutura nas áreas comuns, como áreas de lazer e outras benfeitorias, além da obrigatoriedade de elevadores. Com relação às cidades, como já era esperado, o aluguel é em média mais caro na cidade de são Paulo, seguido de Rio de Janeiro e Belo Horizonte.

e) Inclua todas as iterações entre as variáveis observadas e repita o ajuste do método de mínimos quadrados e do lasso. Como esses ajustes se comparam em relação aos anteriores? Qual foi o melhor modelo encontrado? Esses resultados são esperados?

A tabela 4 apresenta as métricas do desempenho preditivo de cada um dos modelos (mínimos quadrados e lasso) quando incluídas todas as iterações duas a duas entre as variáveis. Nesse novo ajuste, o melhor valor de λ, encontrado pela validação-cruzada no conjunto de treinamento, foi o de 763,268.

Tabela 4: Desempenho Preditivo dos modelos com as iterações duas a duas

Modelo	EQM	Erro padrão	IC	Amplitude do IC
Mínimos quadrados	5.781.091	57,11801	4.963.005 - 6.599.178	1.636.172
Lasso	7.503.856	65,07443	6.510.600 - 8.497.113	1.986.513

É possível perceber que enquanto a adição de mais covariáveis, para o modelo de mínimos quadrados, possibilita uma redução tanto do EQM quanto do erro padrão, o que já era esperado uma vez que o viés no MQO é reduzido pelo aumento de covariáveis, uma situação oposta ocorre ao lasso.

Nesse caso, talvez o aumento da complexidade do modelo não tenha significado um aumento de covariáveis que de fato contribuam para reduzir a variância intrínseca da variável resposta (que é componente do risco esperado) e isso, se por um lado possibilitou que o EQM e o erro padrão do lasso tenham aumentado, por outro não contribuiu para que uma redução ainda maior dessas mesmas métricas para o MQO tivesse sido verificada. Além disso, no caso do lasso, a adição das iterações também fez com que apenas dois coeficientes estimados fossem diferentes de zero (os coeficientes associados às variáveis *bathroom* e *park_ spaces*).

Haja vista esses aspectos, entende-se que o melhor modelo considerando todas as iterações duas a duas é o modelo MQO.

```
#Criando a matriz de planejamento X2 com interações
X2 <- data[,c(2:5,7,8,10,1)]
X2 = model.matrix(~ (.)^2  -1,data = X2)
#Criando o vetor da variável resposta
y = data %>% dplyr::select(rent_am) %>% as.matrix()

#i) Via mínimos quadrados - com interação
mq2 = glmnet(x = X2[data_split == "Treino",],
            y = y[data_split == "Treino"],
            alpha =0, lambda = 0)
round(coefficients(mq2), 4)
y_pred_mq2 = predict(mq2, newx =  X2[data_split == "Teste", ])
mse_mq2 = mean((y[data_split=="Teste"]-y_pred_mq2)^2)
erro_padrao_mq2 = sqrt(mse_mq2/length(y[data_split=="Teste"]))

#ii) Ajustando o LASSO - com interação
fitLinear.cv2 <- cv.glmnet(x = X2[data_split == "Treino",],
                          y = y[data_split == "Treino"],
                          alpha = 1)
#Lambda mínimo pelo CV
lambda2 = fitLinear.cv2$lambda.min
lambda2 #Melhor valor encontrado para lambda
plot(fitLinear.cv2)
#Ajustando o lasso com o lambda obtido pelo CV
lasso2 = glmnet(x = X2[data_split == "Treino",],
               y = y[data_split == "Treino"],
               alpha = 1, lambda = lambda2)
coefficients(lasso2)
pred2 <- predict(lasso2, newx =  X2[data_split == "Teste", ])
mse_lasso2 = mean((y[data_split=="Teste"]-pred2)^2)
erro_padrao_lasso2 = sqrt(mse_lasso2/length(y[data_split=="Teste"]))

IC_Risco_MQ2 = IC_risco(yteste = y[data_split=="Teste"], ypredito = y_pred_mq2)
IC_Risco_Lasso2 = IC_risco(yteste = y[data_split=="Teste"], ypredito = pred2)
rbind(unlist(IC_Risco_MQ2), unlist(IC_Risco_Lasso2))
```

```
coef2 <- cbind(round(coefficients(mq2), 4),
              round(coefficients(lasso2),4))
```

2 Exercício 2

Mostre que

$$\mathbb{E}[(Y - g(\mathbf{X}))^2|\mathbf{X} = \mathbf{x}] = \mathbb{V}[Y|\mathbf{X} = \mathbf{x}] + (r(\mathbf{x}) - \mathbb{E}[g(\mathbf{x})])^2 + \mathbb{V}[g(\mathbf{x})]$$

Tendo em vista que $r(\mathbf{x}) = \mathbb{E}[Y|\mathbf{X}]$ e assumindo a independência entre Y e $g(\mathbf{X})$, dado que obtemos função g a partir do conjunto de treinamento, temos que

$$\mathbb{E}[(Y - g(\mathbf{X}))^2|\mathbf{X} = \mathbf{x}] =$$

$$\begin{aligned}
&= \mathbb{E}[(Y - r(\mathbf{x}) + r(\mathbf{x}) - g(\mathbf{X}))^2|\mathbf{X} = \mathbf{x}] = \\
&= \mathbb{E}[(((Y - r(\mathbf{x})) - (g(\mathbf{X}) - r(\mathbf{x})))^2|\mathbf{X} = \mathbf{x}] = \\
&= \mathbb{E}[(Y - r(\mathbf{x}))^2 - 2(Y - r(\mathbf{x}))(g(\mathbf{X}) - r(\mathbf{x})) + (g(\mathbf{X}) - r(\mathbf{x}))^2|\mathbf{X} = \mathbf{x}] = \\
&= \mathbb{E}[(Y - r(\mathbf{x}))^2|\mathbf{X} = \mathbf{x}] - 2\mathbb{E}[(Y - r(\mathbf{x}))(g(\mathbf{X}) - r(\mathbf{x}))|\mathbf{X} = \mathbf{x}] + \mathbb{E}[(g(\mathbf{X}) - r(\mathbf{x}))^2|\mathbf{X} = \mathbf{x}] = \\
&= \mathbb{E}[(Y - r(\mathbf{x}))^2|\mathbf{X} = \mathbf{x}] + 0 + \mathbb{E}[(g(\mathbf{X}) - r(\mathbf{x}))^2|\mathbf{X} = \mathbf{x}] = \\
&= \mathbb{E}[(Y^2 - 2Yr(\mathbf{x}) + r(\mathbf{x})^2|\mathbf{X} = \mathbf{x}] + \mathbb{E}[(g(\mathbf{X})^2 - 2g(\mathbf{X})r(\mathbf{x}) + r(\mathbf{x})^2)|\mathbf{X} = \mathbf{x}] = \\
&= \mathbb{E}[Y^2|\mathbf{X} = \mathbf{x}] - 2(\mathbb{E}[Y|\mathbf{X} = \mathbf{x}])^2 + (\mathbb{E}[Y|\mathbf{X} = \mathbf{x}])^2 + \mathbb{E}[g(\mathbf{X})^2 - 2g(\mathbf{X})r(\mathbf{x}) + r(\mathbf{x})^2|\mathbf{X} = \mathbf{x}] = \\
&= \mathbb{V}[Y|\mathbf{X} = \mathbf{x}] + \mathbb{E}[g(\mathbf{X})^2|\mathbf{X} = \mathbf{x}] - 2\mathbb{E}[g(\mathbf{X})r(\mathbf{x})|\mathbf{X} = \mathbf{x}] + \mathbb{E}[r(\mathbf{x})^2|\mathbf{X} = \mathbf{x}] = \\
&= \mathbb{V}[Y|\mathbf{X} = \mathbf{x}] + \mathbb{E}[g(\mathbf{x})^2] - 2r(\mathbf{x})\mathbb{E}[g(\mathbf{x})] + r(\mathbf{x})^2 + \mathbb{E}[g(\mathbf{x})]^2 - \mathbb{E}[g(\mathbf{x})]^2 = \\
&= \mathbb{V}[Y|\mathbf{X} = \mathbf{x}] + \mathbb{E}[g(\mathbf{x})^2] - \mathbb{E}[g(\mathbf{x})]^2 + r(\mathbf{x})^2 - 2r(\mathbf{x})\mathbb{E}[g(\mathbf{x})] + \mathbb{E}[g(\mathbf{x})]^2 = \\
&= \mathbb{V}[Y|\mathbf{X} = \mathbf{x}] + \mathbb{V}[g(\mathbf{x})] + r(\mathbf{x})^2 - 2r(\mathbf{x})\mathbb{E}[g(\mathbf{x})] + \mathbb{E}[g(\mathbf{x})]^2 = \\
&= \mathbb{V}[Y|\mathbf{X} = \mathbf{x}] + \mathbb{V}[g(\mathbf{x})] + (r(\mathbf{x}) - \mathbb{E}[g(\mathbf{x})])^2 = \\
&= \mathbb{V}[Y|\mathbf{X} = \mathbf{x}] + (r(\mathbf{x}) - \mathbb{E}[g(\mathbf{x})])^2 + \mathbb{V}[g(\mathbf{x})]
\end{aligned}$$

Obs:

$$\begin{aligned}
\mathbb{E}[(Y - r(\mathbf{x}))(g(\mathbf{X}) - r(\mathbf{x}))|\mathbf{X} = \mathbf{x}] &= \\
&= \mathbb{E}[Yg(\mathbf{X}) - Yr(\mathbf{x}) - r(\mathbf{x})g(\mathbf{X}) + r(\mathbf{x})^2|\mathbf{X} = \mathbf{x}] \\
&= \mathbb{E}[Y|\mathbf{X} = \mathbf{x}]\mathbb{E}[g(\mathbf{x})] - r(\mathbf{x})\mathbb{E}[Y|\mathbf{X} = \mathbf{x}] - r(\mathbf{x})\mathbb{E}[g(\mathbf{x})] + r(\mathbf{x})^2 \\
&= r(\mathbf{x})\mathbb{E}[g(\mathbf{x})] - r(\mathbf{x})^2 - r(\mathbf{x})\mathbb{E}[g(\mathbf{x})] + r(\mathbf{x})^2 \\
&= 0
\end{aligned}$$

3 Exercício 3

Seja $0 < \alpha < 1$ fixo e considere a função de perda

$$L(g; (\mathbf{X}, Y)) = (g(\mathbf{X}) - Y)(\mathbb{I}(Y \leq g(\mathbf{X})) - \alpha)$$

Qual a função g que minimiza a função de risco (aleatório apenas em $(\mathbf{X}, Y)$) correspondente? Interprete e justifique.

Sabemos que a função de risco é a esperança da função de perda. Portanto, temos que

$$R_{pred}(g) = \mathbb{E}[(g(\mathbf{X}) - Y)(\mathbb{I}(Y \leq g(\mathbf{X})) - \alpha)|\mathbf{X} = \mathbf{x}]$$

Logo, podemos reescrever a função de risco como

$$R_{pred}(g) = \int_{-\infty}^{g(\mathbf{x})} (g(\mathbf{x}) - Y)(1-\alpha) f(Y|\mathbf{X} = \mathbf{x})\, dy + \int_{g(\mathbf{x})}^{+\infty} (Y - g(\mathbf{x}))(\alpha) f(Y|\mathbf{X} = \mathbf{x})\, dy =$$

$$= \int_{-\infty}^{g(\mathbf{x})} g(\mathbf{x})(1-\alpha) f(Y|\mathbf{X} = \mathbf{x})\, dy - \int_{-\infty}^{g(\mathbf{x})} Y(1-\alpha) f(Y|\mathbf{X} = \mathbf{x})\, dy$$

$$+ \int_{g(\mathbf{x})}^{+\infty} Y(\alpha) f(Y|\mathbf{X} = \mathbf{x})\, dy - \int_{g(\mathbf{x})}^{+\infty} g(\mathbf{x})(\alpha) f(Y|\mathbf{X} = \mathbf{x})\, dy =$$

$$= g(\mathbf{x})(1-\alpha) F(g(\mathbf{x})|\mathbf{X} = \mathbf{x}) - (1-\alpha) \int_{-\infty}^{g(\mathbf{x})} Y f(Y|\mathbf{X} = \mathbf{x})\, dy$$

$$+ \alpha \int_{g(\mathbf{x})}^{+\infty} Y f(Y|\mathbf{X} = \mathbf{x})\, dy - g(\mathbf{x})\alpha(1 - F(g(\mathbf{x})|\mathbf{X} = \mathbf{x})) =$$

$$= g(\mathbf{x}) F(g(\mathbf{x})|\mathbf{X} = \mathbf{x}) - g(\mathbf{x})\alpha F(g(\mathbf{x})|\mathbf{X} = \mathbf{x}) - (1-\alpha) \int_{-\infty}^{g(\mathbf{x})} Y f(Y|\mathbf{X} = \mathbf{x})\, dy$$

$$+ \alpha \int_{g(\mathbf{x})}^{+\infty} Y f(Y|\mathbf{X} = \mathbf{x})\, dy + g(\mathbf{x})\alpha F(g(\mathbf{x})|\mathbf{X} = \mathbf{x}) - g(\mathbf{x})\alpha =$$

$$= g(\mathbf{x}) F(g(\mathbf{x})|\mathbf{X} = \mathbf{x}) - (1-\alpha) \int_{-\infty}^{g(\mathbf{x})} Y f(Y|\mathbf{X} = \mathbf{x})\, dy + \alpha \int_{g(\mathbf{x})}^{+\infty} Y f(Y|\mathbf{X} = \mathbf{x})\, dy - g(\mathbf{x})\alpha$$

Assim,

$$R_{pred}(g) = g(\mathbf{x}) F(g(\mathbf{x})|\mathbf{X} = \mathbf{x}) - (1-\alpha) \int_{-\infty}^{g(\mathbf{x})} Y f(Y|\mathbf{X} = \mathbf{x})\, dy + \alpha \int_{g(\mathbf{x})}^{+\infty} Y f(Y|\mathbf{X} = \mathbf{x})\, dy - g(\mathbf{x})\alpha$$

Como queremos encontrar a função g que minimiza essa função de risco, precisamos derivar $R_{pred}(g)$ em relação a $g(\mathbf{x})$, que é o nosso estimador de interesse, e igualar esse resultado a zero para encontrar qual o valor de g que faz com que

$$\frac{\partial R_{pred}(g)}{\partial g(\mathbf{x})} = 0$$

$$F(g(\mathbf{x})|\mathbf{X}=\mathbf{x}) + g(\mathbf{x})f(g(\mathbf{x})|\mathbf{X}=\mathbf{x}) - (1-\alpha)g(\mathbf{x})f(g(\mathbf{x})|\mathbf{X}=\mathbf{x})$$
$$+ \alpha(-1)g(\mathbf{x})f(g(\mathbf{x})|\mathbf{X}=\mathbf{x}) - \alpha = 0$$

$$F(g(\mathbf{x})|\mathbf{X}=\mathbf{x}) + g(\mathbf{x})f(g(\mathbf{x})|\mathbf{X}=\mathbf{x}) - g(\mathbf{x})f(g(\mathbf{x})|\mathbf{X}=\mathbf{x}) + \alpha g(\mathbf{x})f(g(\mathbf{x})|\mathbf{X}=\mathbf{x})$$
$$- \alpha g(\mathbf{x})f(g(\mathbf{x})|\mathbf{X}=\mathbf{x}) - \alpha = 0$$

$$F(g(\mathbf{x})|\mathbf{X}=\mathbf{x}) - \alpha = 0$$

$$F(g(\mathbf{x})|\mathbf{X}=\mathbf{x}) = \alpha$$

$$g(\mathbf{x}) = F^{-1}(\alpha|\mathbf{X}=\mathbf{x})$$

Dessa forma, o $\alpha-$ésimo quantil é o estimador que minimiza a função de risco com função de perda dada por $L(g;(\mathbf{X},Y)) = (g(\mathbf{X})-Y)(\mathbb{I}(Y \leq g(\mathbf{X})) - \alpha)$.