Lista 01 - Aprendizado de Máquinas

13 junho, 2022

Exercício 01

Faça uma pesquisa sobre aplicações práticas de aprendizado de máquinas e descreva aplicações de cada um dos problemas abaixo, incluindo a descrição dos vetores de características, rótulos ou respostas quando apropriado para cada um dos problemas:

a) Problema de classificação.

b) Problema de regressão.

c) Problema de Agrupamento.

Exercício 02

Descreva com suas próprias palavras o que é a "maldição da dimensionalidade".

Exercício 03

Implemente o método dos vizinhos mais próximos para um problema de classificação. Sua função deve receber como argumento:

1) Um número k representando o número de vizinhos para usar no kNN;

2) Um vetor $x = (x_1, x_2)$ com duas componentes;

3) Uma *dataframe* D com colunas x_1, x_2 e y, em que x_1 e x_2 são valores numéricos representando duas componentes do vetor de características x (i.e., $x = (x_1, x_2)$) e y é um fator representando os rótulos dos pontos. Abaixo tem-se um exemplo dessa *dataframe* com duas classes:

```
D <- tibble( x_1 = rnorm(100,1,1),
             x_2 = rnorm(100,-1,2),
             y   = factor(sample(c("one","two","three"),100,replace = T)))
head(D)
```

x_1	x_2	y
1.1285962	0.5696290	two
0.3996574	-0.3337484	one
0.8739806	2.0674261	one
1.0546384	-1.8910249	two
0.5064259	1.1835884	three
1.7283965	2.0934309	two

A função deve ter a assinatura `function(k,x,D)` e deve retornar a classe mais provável associada ao ponto x.

dica: Você pode fazer o kNN usando uma sequencia de comandos encadeados pelo operador pipe `%>%`. Por exemplo, teste a seguinte sequencia de comandos com a *dataframe* D anterior:

```
x = c(1,2)
k = 10
D2 <- D %>%
  mutate( dist = (x[1] - x_1)^2 + (x[2] - x_2)^2 ) %>%
  arrange( dist ) %>% head(k) %>% count(y)
```

Exercício 04

Usando o banco de dados `iris` e sua implementação do kNN do exercício anterior, calcule quantos pontos são classificados corretamente de acordo com o rótulo `Species` usando as colunas `Petal.length` e `Sepal.length` com $k = 10$ e com $k = 1$.

dica 1: Você pode carregar o banco Iris no R da seguinte forma:

```
library(tidyverse)
data("iris") # Carrega o banco no ambiente global
iris <- as_tibble(iris) %>%  # Converte para a dataframe tibble
  select(Petal.Length,Sepal.Length,Species) %>% # Seleciona colunas da dataframe
  rename( x_1 = Petal.Length, x_2 = Sepal.Length, y = Species) # Renomeia as colunas

head(iris)
```

x_1	x_2	y
1.4	5.1	setosa
1.4	4.9	setosa
1.3	4.7	setosa
1.5	4.6	setosa
1.4	5.0	setosa
1.7	5.4	setosa

dica 2: As funções `map` da biblioteca `purrr` do pacote *tidyverse* são muito úteis! Por exemplo, a função `pmap_lgl` aplica uma função à argumentos fornecidos por uma lista e retorna os resultados da função como um vetor de booleanos (neste caso, a função deve retornar valores booleanos). Rode o exemplo abaixo:

```
l_iris <- as.list(iris) # converte a dataframe em uma lista
                        # que possui elementos com nomes x_1, x_2 e y

# Aplica a função usando os valores dados pelos valores na lista,
#  concatena os resultados da função em um vetor booleano.
v_bool <- pmap_lgl(l_iris, function(x_1,x_2,y){
          print(str_c(x_1,", ",x_2,", ",y))
          return( y == "setosa" )
})
```

A função `sum` pode também ser útil. Lembre-se de usar o comando `?` para ajuda.

Exercício 5 (opcional)

Em aula vimos como calcular a função de regressão $f : \mathcal{X} \to \mathcal{Y}$ ótima que minimiza o risco esperado:

$$\mathcal{R}(f) = \mathbb{E}_{XY}[\ell(Y, f(X))]$$

quando a função de perda é dada por $\ell(y, y') := (y - y')^2$. Essa função de perda é geralmente usada por possuir derivada contínua. Mas existem outras funções de perda, como a função de perda do erro absoluto,

que é dada por: $\ell_a(y, y') := |y - y'|$. Mostre que a função de f ótima, que minimiza o risco esperado com essa função de perda, é dada por $f(x) := \text{Mediana}(Y|X = x)$.

dica 1: A mediana de uma variável aleatória contínua tomando valor em $\mathbb{R}$ é definida como sendo o valor real m tal que $P(Y > m) = P(Y < m) = 1/2$.

dica 2: A derivada de $\ell_a(y, y')$ em relação a y' quando $y' \neq y$ existe e é limitada. Nestes casos, sabe-se que

$$\frac{\partial}{\partial z}\mathbb{E}[\ell_a(Y, z)|X = x] = \mathbb{E}\Big[\frac{\partial}{\partial z}\ell_a(Y, z)\Big|X = x\Big].$$

Exercício 6 (opcional)

Considere que m pontos são espalhados uniformemente em uma hiperesfera de raio unitário e dimensão d. Mostre que a mediana da distância do ponto mais próximo à origem é dada por: $(1 - 0.5^{1/m})^{1/d}$.