TP1: Clasificación binaria - Predicción de la comestibilidad de hongos

Micaela Oliva, Camila Bernardez

2024-08-15

Ejercicio 1: Introducción al problema

El dataset elegido tiene como origen:

https://www.kaggle.com/datasets/vishalpnaik/mushroom-classification-edible-or-poisonous?resource=download&select=mushroom.csv

```
mushrooms <- read.csv("mushrooms_v2.csv")
```

Es un dataset que busca clasificar si un hongo es comestible (edible) o no (poisonous). Para considerarlo, el dataset esta compuesto de 16 variables, de las cuáles:

- `class` (variable categórica binaria): indica si un hongo es comestible o no, y es lo que buscamos predecir.
  - edible (1)
  - poisonous (0)

```
unique(mushrooms$class)
```

```
## [1] 0 1
```

- `cap-diameter` (variable numérica): indica el diametro del sombrero del hongo en cm.
- `cap-shape` (variable categórica): indica el forma del sombrero del hongo.
  - 'bell'
  - 'conical'
  - 'convex'
  - 'flat'
  - 'sunken'
  - 'spherical'
  - 'others'

```
unique(mushrooms$cap.shape)
```

```
## [1] "convex"    "flat"      "spherical" "bell"      "conical"   "sunken"
## [7] "others"
```

- `cap-surface` (variable categórica): indica la textura de la superficie del sombrero del hongo.
  - 'fibrous'
  - 'grooves'
  - 'scaly'
  - 'smooth'
  - 'dry'
  - 'shiny'

  - 'leathery'
  - 'silky'
  - 'sticky'
  - 'wrinkled'
  - 'fleshy'
  - '' #

```
unique(mushrooms$cap.surface)
```

```
##  [1] "grooves"  "shiny"    ""         "sticky"   "scaly"    "fleshy"
##  [7] "smooth"   "leathery" "dry"      "wrinkled" "fibrous"  "silky"
```

- `cap-color` (variable categórica): indica el color del sombrero del hongo.
  - 'brown'
  - 'orange'
  - 'buff'
  - 'gray'
  - 'green'
  - 'pink'
  - 'purple'
  - 'red'
  - 'white'
  - 'yellow'
  - 'blue'
  - 'black'

```
unique(mushrooms$cap.color)
```

```
##  [1] "orange" "red"    "brown"  "gray"   "green"  "white"  "yellow" "pink"
##  [9] "purple" "buff"   "blue"   "black"
```

- `does-bruise-or-bleed` (variable categórica binaria -> true/false): indica si el hongo al lesionarse presenta moratones o sangrado.
  - 'true'
  - 'false'

```
unique(mushrooms$does.bruise.or.bleed)
```

```
## [1] 0 1
```

- `gill-attachment` (variable categórica): indica cómo las láminas del hongo se adhieren al pie.
  - 'adnate'
  - 'adnexed'
  - 'decurrent'
  - 'free'
  - 'sinuate'
  - 'pores'
  - 'none' #
  - '' #

```
unique(mushrooms$gill.attachment)
```

```
## [1] "free"      ""          "adnate"    "decurrent" "sinuate"   "adnexed"
## [7] "pores"     "none"
```

- `gill-spacing` (variable categórica): indica la separación entre las láminas del hongo.
  - 'close'

  - 'distant'
  - 'none' #
  - '' #

```
unique(mushrooms$gill.spacing)
```

```
## [1] ""        "close"   "distant" "none"
```

- `stem-heigh` (variable numérica): indica la altura del pie del hongo en cm.
- `stem-width` (variable numérica): indica el ancho del pie del hongo en mm.
- `stem-root` (variable categórica): indica la estructura de la raíz del pie del hongo.
  - 'bulbous'
  - 'swollen'
  - 'club'
  - 'cup' #extra
  - 'equal' #extra
  - 'rhizomorphs'#extra
  - 'rooted'
  - '' #falta es f, supongo que es none
  - '' #

```
unique(mushrooms$stem.root)
```

```
## [1] "swollen" ""        "bulbous" "rooted"  "club"    "f"
```

- `veil-type` (variable categórica): indica el tipo de velo que cubre las láminas del hongo.
  - 'partial' #extra
  - 'universal'
  - '' #

```
unique(mushrooms$veil.type)
```

```
## [1] "universal" ""
```

- `has-ring` (variable categórica binaria -> true/false): indica si esta presente un anillo en el hongo o no.
  - 'true'
  - 'false'

```
unique(mushrooms$has.ring)
```

```
## [1] 1 0
```

- `ring-type` (variable categórica): indica el tipo del anillo presente en el hongo.
  - 'cobwebby' #extra
  - 'evanescent'
  - 'flaring'
  - 'grooved'
  - 'large'
  - 'pendant'
  - 'sheathing', #extra
  - 'zone'
  - 'scaly' #extra
  - 'movable'
  - 'none'
  - 'unknown' #extra
  - ''

```r
unique(mushrooms$ring.type)
```

```
## [1] "grooved"    "pendant"    "evanescent" "large"      "none"
## [6] "movable"    ""           "flaring"    "zone"
```

- `habitat` (variable categórica): indica el ambiente en el cual el hongo fue encontrado.
  - 'grasses'
  - 'leaves'
  - 'meadows'
  - 'paths'
  - 'heaths'
  - 'urban'
  - 'waste'
  - 'woods'

```r
unique(mushrooms$habitat)
```

```
## [1] "woods"   "meadows" "grasses" "heaths"  "leaves"  "paths"   "waste"
## [8] "urban"
```

- `season` (variable categórica): indica la estación en la cual el hongo es comunmente observado.
  - 'spring'
  - 'summer'
  - 'autumn'
  - 'winter'

```r
unique(mushrooms$season)
```

```
## [1] "winter" "summer" "autumn" "spring"
```

Decidimos elegir este conjunto de datos en particular para el uso de árboles de decisión, ya que contiene variables númericas y categóricas, tanto binarias como multiclase.

Ejercicio 2: Preparación de los datos

Carga del conjunto de datos y realizamiento del preprocesamiento necesario

Análisis exploratorio de datos: Estadísticas descriptivas y visualizaciones de las variables principales

Para empezar, veamos cuantos valores tenemos que son NA, none o vacíos/missing para cada una de las variables:

```r
# class
missing_count_class <- sum(
  is.na(mushrooms$class) |
    mushrooms$class == "None" |
    mushrooms$class == "",
    na.rm = TRUE
)
print(paste("class: ", missing_count_class))
```

```
## [1] "class:  0"
```

```r
# cap-diameter
missing_count_cap_diameter <- sum(
  is.na(mushrooms$cap.diameter) |
    mushrooms$cap.diameter == "None" |
```

```
    mushrooms$cap.diameter == "",
    na.rm = TRUE
)
print(paste("cap-diameter: ", missing_count_cap_diameter))
```

```
## [1] "cap-diameter:  0"
```

```
# cap-shape
missing_count_cap_shape <- sum(
  is.na(mushrooms$cap.shape) |
    mushrooms$cap.shape == "None" |
    mushrooms$cap.shape == "",
    na.rm = TRUE
)
print(paste("cap-shape: ", missing_count_cap_shape))
```

```
## [1] "cap-shape:  0"
```

```
# cap-surface
missing_count_cap_surface <- sum(
  is.na(mushrooms$cap.surface) |
    mushrooms$cap.surface == "None" |
    mushrooms$cap.surface == "",
    na.rm = TRUE
)
print(paste("cap-surface: ", missing_count_cap_surface))
```

```
## [1] "cap-surface:  14120"
```

```
# cap-color
missing_count_cap_color <- sum(
  is.na(mushrooms$cap.color) |
    mushrooms$cap.color == "None" |
    mushrooms$cap.color == "",
    na.rm = TRUE
)
print(paste("cap-color: ", missing_count_cap_color))
```

```
## [1] "cap-color:  0"
```

```
# does-bruise-or-bleed
missing_count_bruise_or_bleed <- sum(
  is.na(mushrooms$does.bruise.or.bleed) |
    mushrooms$does.bruise.or.bleed == "None" |
    mushrooms$does.bruise.or.bleed == "",
    na.rm = TRUE
)
print(paste("does-bruise-or-bleed: ", missing_count_bruise_or_bleed))
```

```
## [1] "does-bruise-or-bleed:  0"
```

```
# gill-attachment

# gill-spacing

# stem-heigh
```

```
# stem-width

# stem-root

# veil-type

# has-ring

# ring-type

# habitat

# season
```

Ejercicio 3: Construcción de un árbol de decisión básico