Laboratorio de Imputación y Normalización

STDT Ing. George Albadr

2024-11-18

```
data_path <- "/Users/georgealbadr/GitHub/Data-Wrangling/Lab9/titanic_MD.csv"
complete_data_path <- "/Users/georgealbadr/GitHub/Data-Wrangling/Lab9/titanic.csv"

titanic_md <- read.csv(data_path, stringsAsFactors = FALSE)
titanic_complete <- read.csv(complete_data_path, stringsAsFactors = FALSE)
```

```
missing_report <- titanic_md %>%
  summarise(across(everything(), ~sum(is.na(.)))) %>%
  gather(key = "Column", value = "MissingCount") %>%
  mutate(MissingPercentage = round((MissingCount / nrow(titanic_md)) * 100, 2))

missing_report
```

```
##         Column MissingCount MissingPercentage
## 1  PassengerId            0              0.00
## 2     Survived            0              0.00
## 3       Pclass            0              0.00
## 4         Name            0              0.00
## 5          Sex            0              0.00
## 6          Age           25             13.66
## 7        SibSp            3              1.64
## 8        Parch           12              6.56
## 9       Ticket            0              0.00
## 10        Fare            8              4.37
## 11       Cabin            0              0.00
## 12    Embarked            0              0.00
```

Métodos para Imputación de Missing Values:

1. Age (13.66% faltantes):

Método: Media (promedio)

Razón: La edad es una variable numérica continua que suele tener una distribución cercana a la normal.

Usar la media asegura que la distribución general de los datos no se distorsione significativamente,

salvo que existan muchos outliers.

2. SibSp (1.64% faltantes):

Método: Moda

Razón: Es una variable discreta que indica el número de hermanos o cónyuges a bordo.

Como el porcentaje de datos faltantes es bajo, imputar por la moda (el valor más frecuente)

es adecuado, ya que es menos probable que cause distorsiones en los datos.

3. Parch (6.56% faltantes):

Método: Moda

Razón: Similar a SibSp, es una variable categórica/discreta que indica el número de padres o hijos a bordo.

La moda es una buena elección para mantener consistencia en los valores más comunes.

4. Fare (4.37% faltantes):

Método: Mediana

Razón: Es una variable numérica continua que puede tener valores extremos (outliers).

La mediana es robusta frente a outliers y representa mejor el valor central de la distribución.

Columnas sin valores faltantes:

```r
complete_rows <- titanic_md[complete.cases(titanic_md), ]
n_complete <- nrow(complete_rows)
paste("Número de filas completas: ", n_complete)
```

```
## [1] "Número de filas completas:  141"
```

```r
imputed_mean <- titanic_md %>%
  mutate(across(where(is.numeric), ~ifelse(is.na(.), mean(., na.rm = TRUE), .)))

imputed_mode <- titanic_md %>%
  mutate(across(where(is.numeric), ~ifelse(is.na(.), as.numeric(names(sort(table(.), decreasing = TRUE)

imputed_median <- titanic_md %>%
  mutate(across(where(is.numeric), ~ifelse(is.na(.), median(., na.rm = TRUE), .)))
```

```r
fit <- lm(Age ~ Pclass + Sex + SibSp + Parch + Fare, data = titanic_md, na.action = na.exclude)
titanic_md$Age[is.na(titanic_md$Age)] <- predict(fit, titanic_md[is.na(titanic_md$Age), ])
```

```r
outliers_sd <- titanic_md %>%
  filter(if_any(where(is.numeric), ~abs(scale(.)) > 3))
nrow(outliers_sd)
```

```
## [1] 16
```

#Comparación contra titanic.csv

```r
numeric_columns <- names(titanic_md)[sapply(titanic_md, is.numeric)]



compare_methods <- function(column_name, method_data) {
  mse <- mean((titanic_complete[[column_name]] - method_data[[column_name]])^2, na.rm = TRUE)
  return(mse)
}


compare_results <- data.frame(
  Column = numeric_columns,
  Mean_Imputation = sapply(numeric_columns, function(col) compare_methods(col, imputed_mean)),
  Mode_Imputation = sapply(numeric_columns, function(col) compare_methods(col, imputed_mode)),
  Median_Imputation = sapply(numeric_columns, function(col) compare_methods(col, imputed_median))
)
options(scipen = 999)

compare_results
```

```
##                  Column Mean_Imputation Mode_Imputation Median_Imputation
## PassengerId PassengerId     0.000000000      0.00000000        0.00000000
## Survived       Survived     0.000000000      0.00000000        0.00000000
## Pclass           Pclass     0.000000000      0.00000000        0.00000000
## Age                 Age    33.514202819     51.76775956       33.51229508
## SibSp             SibSp     0.004335661      0.01092896        0.01092896
## Parch             Parch     0.039176970      0.06557377        0.06557377
## Fare               Fare   155.951612838    247.02139504      164.97544297
```

Resultados de la comparación de imputaciones

PassengerId:

Todas las imputaciones (media, moda y mediana) son 0, ya que no había valores faltantes en esta columna.

Por lo tanto, no se aplicaron cambios.

Survived:

Todas las imputaciones (media, moda y mediana) son 0, ya que no había valores faltantes en esta columna.

Esta columna no requería imputación.

Pclass:

Todas las imputaciones (media, moda y mediana) son 0, ya que no había valores faltantes en esta columna.

No se realizaron ajustes.

Age:

- Mean Imputation: 33.5142

- Mode Imputation: 51.7678

- Median Imputation: 33.5123

Observaciones: La imputación por media y mediana generan valores muy similares y razonables.

Sin embargo, la moda da un valor mucho más alto, probablemente debido a la naturaleza

discreta de la moda.

SibSp:

- Mean Imputation: 0.0043

- Mode Imputation: 0.0109

- Median Imputation: 0.0109

Observaciones: Todos los métodos producen valores bajos y similares, con diferencias mínimas.

```r
titanic_standardized <- titanic_md %>%
  mutate(across(where(is.numeric), scale))
```

#MinMaxScaling

```r
titanic_minmax <- titanic_md %>%
  mutate(across(where(is.numeric), ~(. - min(., na.rm = TRUE)) / (max(., na.rm = TRUE) - min(., na.rm =
```

#MaxAbsScaler

```r
titanic_maxabs <- titanic_md %>%
  mutate(across(where(is.numeric), ~. / max(abs(.), na.rm = TRUE)))
```

```r
stat_comparison_original <- titanic_complete %>%
  summarise(across(where(is.numeric), list(mean = mean, sd = sd), na.rm = TRUE))

stat_comparison_standardized <- titanic_standardized %>%
  summarise(across(where(is.numeric), list(mean = mean, sd = sd), na.rm = TRUE))

list(Original = stat_comparison_original, Standardized = stat_comparison_standardized)
```

```
## $Original
##   PassengerId_mean PassengerId_sd Survived_mean Survived_sd Pclass_mean
## 1         455.3661       247.0525     0.6721311   0.4707247    1.191257
##   Pclass_sd Age_mean   Age_sd SibSp_mean  SibSp_sd Parch_mean  Parch_sd
## 1  0.515187 35.67443 15.64387  0.4644809 0.6441586  0.4754098 0.7546171
##   Fare_mean  Fare_sd
## 1  78.68247 76.34784
##
## $Standardized
##           PassengerId_mean PassengerId_sd             Survived_mean Survived_sd
## 1 0.00000000000000003445938              1 -0.00000000000000001735103           1
##                Pclass_mean Pclass_sd                 Age_mean Age_sd
## 1 0.00000000000000001650168         1 0.00000000000000001582692      1
##                  SibSp_mean SibSp_sd                Parch_mean Parch_sd
## 1 -0.00000000000000001319932        1 0.00000000000000007271636        1
##                  Fare_mean Fare_sd
## 1 0.00000000000000003755726       1
```