10.- Imbalanced Analysis_04_06_turismo_gasto_completo_v_01

June 15, 2023

#

CU55_Modelo agregado de estimación del gasto medio por turista

Citizenlab Data Science Methodology > II - Data Processing Domain *** > # 10.- Imbalanced Analysis

Data Balancing is the process to obtain an adequate data balance if is required, in order to have the adequate amount of data that reflects the intrinsic structure of the problem to be solved.

0.1 Tasks

Imbalanced Analysis

Evaluate Imbalanced Classification Models

Select appopiate metrics

Data Balancing

- Undersampling the Majority Class
- Oversampling the Minority Class
- Mix under-oversampling
- Evaluate a model with random oversampling and undersampling

Cost-Sensitive Algorithms

0.2 File

- Input File: CU_45_06_03_turismo_receptor.csv
- Sampled Input File: CU_45_07_03_turismo_receptor.csv
- Output File: No aplica

0.2.1 Encoding

Con la siguiente expresión se evitan problemas con el encoding al ejecutar el notebook. Es posible que deba ser eliminada o adaptada a la máquina en la que se ejecute el código.

```
In [1]: Sys.setlocale(category = "LC_ALL", locale = "es_ES.UTF-8")
```

```
'LC_COLLATE=es_ES.UTF-8;LC_CTYPE=es_ES.UTF-8;LC_MONETARY=es_ES.UTF-8;LC_NUMERIC=C;LC_TIME=es_ES.UTF-8'
```

0.3 Settings

0.3.1 Libraries to use

```
In [2]: library(readr)
        library(dplyr)
        library(tidyr)
        library(stringr)
```

```
Attaching package: 'dplyr'


The following objects are masked from 'package:stats':

    filter, lag


The following objects are masked from 'package:base':

    intersect, setdiff, setequal, union
```

0.3.2 Paths

```
In [3]: iPath <- "Data/Input/"
        oPath <- "Data/Output/"
```

0.4 Data Load

OPCION A: Seleccionar fichero en ventana para mayor comodidad
Data load using the {tcltk} package. Ucomment the line if using this option

```
In [4]: # file_data <- tcltk::tk_choose.files(multi = FALSE)
```

OPCION B: Especificar el nombre de archivo

```
In [5]: iFile <- "CU_55_06_03_gasto_municipio.csv"
        file_data <- paste0(iPath, iFile)

        if(file.exists(file_data)){
            cat("Se leerán datos del archivo: ", file_data)
        } else{
            warning("Cuidado: el archivo no existe.")
        }
```

```
Se leerán datos del archivo:  Data/Input/CU_55_06_03_gasto_municipio.csv
```

Data file to dataframe Usar la función adecuada según el formato de entrada (xlsx, csv, json, ...)

```
In [6]: data <- read_csv(file_data)
```

```
Rows: 50294 Columns: 9
 Column specification
Delimiter: ","
chr (5): mes, pais_orig_cod, pais_orig, mun_dest, CMUN
dbl (4): mun_dest_cod, turistas, gasto, Target

 Use `spec()` to retrieve the full column specification for this data.
 Specify the column types or set `show_col_types = FALSE` to quiet this message.
```

Visualizo los datos.
Estructura de los datos:

```
In [7]: data |> glimpse()
```

```
Rows: 50,294
Columns: 9
$ mes          <chr> "2019-07", "2019-07", "2019-07", "2019-07", "2019-07", "
$ pais_orig_cod <chr> "000", "010", "011", "030", "110", "121", "123", "126",
$ pais_orig    <chr> "Total", "Total Europa", "Total Unión Europea", "Total A
$ mun_dest_cod <dbl> 28002, 28002, 28002, 28002, 28002, 28002, 28002, 28002,
$ mun_dest     <chr> "Ajalvir", "Ajalvir", "Ajalvir", "Ajalvir", "Ajalvir", "
$ turistas     <dbl> 338, 290, 268, 37, 56, 54, 37, 40, 157, 116, 109, 8461,
$ CMUN         <chr> "002", "002", "002", "002", "002", "002", "002", "002",
$ gasto        <dbl> 86.78, 86.78, 86.78, 86.78, 76.36, 78.92, 93.65, 102.04,
$ Target       <dbl> 86.78, 86.78, 86.78, 86.78, 76.36, 78.92, 93.65, 102.04,
```

Muestra de los primeros datos:

```
In [8]: data |> slice_head(n = 5)
```

A spec_tbl_df: 5 Œ 9

mes <chr>	pais_orig_cod <chr>	pais_orig <chr>	mun_dest_cod <dbl>	mun_dest <chr>	turistas <dbl>
2019-07	000	Total	28002	Ajalvir	338
2019-07	010	Total Europa	28002	Ajalvir	290
2019-07	011	Total Unión Europea	28002	Ajalvir	268
2019-07	030	Total América	28002	Ajalvir	37
2019-07	110	Francia	28002	Ajalvir	56

0.5 Imbalanced Analysis

No aplica al caso de uso estudiado.

```
In [9]: # # If not already installed, install the ggplot2 package
        # if(!require(ggplot2)) install.packages('ggplot2')

        # # Load the ggplot2 package
        # library(ggplot2)

        # # Select the column name
        # column_name <- "Target"  # replace with your column name

        # # Create a histogram of the numeric column
        # ggplot(data, aes_string(x = column_name)) +
        #   geom_histogram(binwidth = 10, fill = "blue", color = "black") +
        #   theme_minimal() +
        #   ggtitle(paste("Histogram of", column_name))

        # # Calculate basic statistical measures
        # summary(data[[column_name]])
```

0.6 Evaluate Imbalanced Classification Models

No aplica

0.7 Undersampling the Majority Class

No aplica

0.8 Oversampling the Minority Class

No aplica

0.9 Combine Data Undersampling and Oversampling with SMOTEENN

No aplica

0.10 Evaluating a model with random oversampling and undersampling

No aplica

0.11 Cost-Sensitive Algorithms

No aplica