Homework 2 Report

Classification of liver malfunction severity (Logistic Regression)

true

September 20, 2021

Contents

Overview

In this assignment we will:

- Extract, load, and transform (*ELT*) the `Liver.txt` dataset.
- Perform exploratory data analysis (*EDA*) on the dataset.
- Fit and analyze a logistic regression model on the dataset.
- Perform multiple cross-validation tasks at different k-fold values ($k = 3$, $k = 10$).

This report was generated as a vignette in a formal R Package structured specifically for data analysis, with the following package requirements:

```
library(dplyr)
library(forcats)
library(ggplot2)
```

ELT

The `Liver.txt` file contains a header-less set of comma-separated fields, with one record per newline.

Extraction

We begin by getting the filepath for `Liver.txt`, which is stored in the package's `inst/extdata` folder:

```r
# Retrieve paths to data files.
liver.filepath <- system.file(
  "extdata",
  "Liver.txt",
  package = "RIT.STAT745.HW2",
  mustWork = TRUE
)
```

Next, we prepare the column names:

```r
# Prepare the variable names.
liver.variables <- c(
  # X1 through X5 are quantitative blood test results. (Unknown tests).
  "blood.1",
  "blood.2",
  "blood.3",
  "blood.4",
  "blood.5",
  # X6: No. of alcoholic beverages consumed.
  "drinks",
  # X7: Liver condition severity.
  "severity"
)
```

Then, we prepare the column types:

```r
# Prepare the variable types.
liver.types <- readr::cols(
  # X1 through X5 - Quantitative blood test results.
  readr::col_integer(),
  readr::col_integer(),
  readr::col_integer(),
  readr::col_integer(),
  readr::col_integer(),
  # X6 - No. of alcoholic beverages.
  readr::col_double(),
  # X7 - Severity group.
  readr::col_factor()
)
```

Now we extract and save the `Liver.txt` file to place in our `data/` folder as a `liver_data.rda` file. This also allows us to refer to the data with `liver_data`.

```
# Read the dataset into memory.
liver.txt <- readr::read_csv(
  # File is located in inst/extdata
  file = liver.filepath,
  # First row is NOT a header row.
  col_names = liver.variables,
  # Column types known in advance.
  col_types = liver.types
)
```

```
# Store as tibble.
liver_data <- dplyr::as_tibble(liver.txt)

# Write *.csv file in data-raw/
readr::write_csv(liver_data, "data-raw/liver_data.csv")

# Save the imported Liver.txt tibble.
usethis::use_data(liver_data, overwrite = TRUE)
# v Saving 'liver_data' to 'data/liver_data.rda'
# * Document your data (see 'https://r-pkgs.org/data.html')
```

Loading

The `*.rda` data files are saved within the `data/` folder. When the package is installed (eg., `devtools::load_all("RIT.STAT745.HW2")` from the project directory), this attaches our `liver_data` tibble to the working environment:

```
library(RIT.STAT745.HW2)

str(liver_data)
# tibble [345 x 7] (S3: tbl_df/tbl/data.frame)
#  $ blood.1 : int [1:345] 85 85 86 91 87 98 88 88 92 90 ...
#  $ blood.2 : int [1:345] 92 64 54 78 70 55 62 67 54 60 ...
#  $ blood.3 : int [1:345] 45 59 33 34 12 13 20 21 22 25 ...
#  $ blood.4 : int [1:345] 27 32 16 24 28 17 17 11 20 19 ...
#  $ blood.5 : int [1:345] 31 23 54 36 10 17 9 11 7 5 ...
#  $ drinks  : num [1:345] 0 0 0 0 0 0 0.5 0.5 0.5 0.5 ...
#  $ severity: Factor w/ 2 levels "1","2": 1 2 2 2 2 2 1 1 1 1 ...
```

Transformation

Transformation of the dataset is minimal. In our case we want to encode our response variable `severity` to properly mention our *positive* and *negative* class labels. We can trivally declare a lookup table that provides us with an adequate data dictionary:

```
severity_map
```

status	value	code
severe	1	1
not severe	2	0

```
liver <- liver_data %>%
  mutate(severity = fct_recode(severity,
    "1" = "1",
    "0" = "2"
  ))
liver %>%  select(severity)
```

severity
1
0
0
0
0
0
1
1
1
1
1
1
1
1
1
1
1
1
1
1
1
1
1
1
1
1
1
1
1
1
1
1
1
1
1
0
0

severity
0
0
0
0
0
0
0
0
0
0
0
0
0
0
0
0
0
0
0
0
0
0
0
0
0
0
0
1
1
0
0
0
0
1
1
1
1
1
1
0
0
0
0
0
0
0
0
0
0
0

severity
0
1
1
1
1
1
1
1
1
0
0
0
0
0
0
1
1
1
1
1
1
1
0
0
0
0
0
0
0
0
0
0
0
0
0
0
0
0
0
0
0
0
0
1
1
0
0
0
0
0

severity
0
0
0
1
1
1
1
1
1
1
1
1
1
0
0
0
0
0
0
0
0
0
0
0
0
0
0
0
0
1
1
0
1
1
1
1
1
1
1
0
0
0
0
0
1
1
0
0
0
0

severity
0
1
1
1
1
0
1
1
1
1
1
1
1
1
1
1
1
1
1
1
1
1
1
1
1
1
1
1
1
1
0
0
0
0
0
0
0
0
0
0
0
0
0
0
0
0
0
0
0
0

severity
0
0
0
0
0
0
1
1
1
1
1
1
0
0
0
0
0
1
1
1
1
1
1
1
1
1
0
0
0
0
0
0
0
0
1
1
1
0
0
0
0
1
0
0
0
0
0
0
0
0

severity
0
0
0
0
0
0
0
0
0
0
0
0
0
0
0
0
0
0
0
0
1
1
1
1
1
1
1
1
1
1
1
0
0
0
0
0
0
1
1
1
1
1
0
0
0
0
0
1
1
0

severity
0
0
0
1
0
1
1
1

EDA

Classification

For this classification problem, we want to predict whether or not a patient's liver malfunction will be severe, given the predictors available in the dataset.

The probability of a test observation having the positive class label outcome ("Group 1") is described by p as such:

$$p = Pr(Y = 1)$$

Likewise, the probability of a test observation having the negative class label outcome ("Group 2") is described by q as such:

$$q = 1 - p = Pr(Y = 0)$$

severity can only take one of two values: 0 or 1. We denote $p = Pr(severity = 1)$ and we will fit a logistic regression model:

$$ln[\frac{p}{(1-p)}] = \beta_0 + \beta_1(blood.1) + \cdots + \beta_5(blood.5) + \beta_6(drinks)$$

We will predict $severity = 1$ if $p >= \pi_0$ and as 0 otherwise. π_0 represents some arbitrary threshold probability we can select, but we'll begin with $\pi_0 = 0.5$.

Session Information

This document was generated from an R Markdown Notebook (See the `vignettes/HW2_report.Rmd` in the package's sub-directory). The setup chunk for this document sets the root directory to the project root directory using the `rprojroot` package; all file paths are relative to the project root.

```
# R version 4.1.1 (2021-08-10)
# Platform: x86_64-w64-mingw32/x64 (64-bit)
# Running under: Windows 10 x64 (build 19042)
#
# Matrix products: default
#
# locale:
# [1] LC_COLLATE=English_United States.1252
# [2] LC_CTYPE=English_United States.1252
# [3] LC_MONETARY=English_United States.1252
# [4] LC_NUMERIC=C
# [5] LC_TIME=English_United States.1252
#
# attached base packages:
# [1] stats     graphics  grDevices datasets  utils
# [6] methods   base
#
# other attached packages:
# [1] RIT.STAT745.HW2_0.1.1 ggplot2_3.3.5
# [3] forcats_0.5.1         dplyr_1.0.7
# [5] rprojroot_2.0.2       knitr_1.36
#
# loaded via a namespace (and not attached):
#  [1] tidyselect_1.1.1 xfun_0.26        bslib_0.3.0
#  [4] purrr_0.3.4      colorspace_2.0-2 vctrs_0.3.8
#  [7] generics_0.1.0   usethis_2.0.1    htmltools_0.5.2
# [10] yaml_2.2.1       utf8_1.2.2       rlang_0.4.11
# [13] pillar_1.6.3     jquerylib_0.1.4  glue_1.4.2
# [16] withr_2.4.2      DBI_1.1.1        bit64_4.0.5
# [19] lifecycle_1.0.1  stringr_1.4.0    munsell_0.5.0
# [22] gtable_0.3.0     codetools_0.2-18 evaluate_0.14
# [25] tzdb_0.1.2       fastmap_1.1.0    parallel_4.1.1
# [28] fansi_0.5.0      highr_0.9        readr_2.0.2
# [31] renv_0.14.0      scales_1.1.1     desc_1.4.0
# [34] vroom_1.5.5      jsonlite_1.7.2   fs_1.5.0
# [37] bit_4.0.4        hms_1.1.1        digest_0.6.28
# [40] stringi_1.7.4    grid_4.1.1       cli_3.0.1
# [43] tools_4.1.1      magrittr_2.0.1   sass_0.4.0
# [46] tibble_3.1.3     crayon_1.4.1     pkgconfig_2.0.3
# [49] ellipsis_0.3.2   assertthat_0.2.1 rmarkdown_2.11
# [52] rstudioapi_0.13  R6_2.5.1         compiler_4.1.1
```