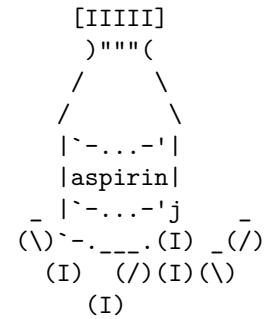
Práctica Machine Learning

Fernández Hernández, Alberto

4/28/2021



1. Introducción y descripción de los datos

El objetivo del presente proyecto consiste en **elaborar un modelo de clasificación binaria que permita predecir si un paciente presentará o será más propenso a padecer una complicación hospitalaria tras una intervención quirúrgica** [1]. Originalmente, el fichero original (extraído de la plataforma Kaggle) contiene tres variables objetivo, dos continuas:

1. *ccsComplicationRate*: incidencia general de complicaciones hospitalarias por cada tipo de intervención quirúrgica.
2. *complication_rsi*: índice de estratificación de riesgo en complicaciones hospitalarias.

Y una binaria:

3. *complication*: **si el paciente ha sufrido una complicación (1) o no (0)**.

Por tanto, de cara a la práctica tendremos únicamente en cuenta, como variable objetivo, la columna *complication*, descartando las dos variables continuas anteriores.

En relación con las posibles variables *input*, nos encontramos con las siguientes:

CONTINUAS

1. *bmi*: **índice de masa de corporal**.
2. *Age*: **edad del paciente**.
3. *baseline_charlson*: **índice de comodidad de Charlson, el cual predice la mortalidad a diez años de un paciente que puede tener una variedad de condiciones comórbidas (como una enfermedad cardíaca, SIDA o cáncer)**.
4. *ahrq_ccs*: **tipo de procedimiento/intervención quirúrgica, etiquetado por la Agencia estadounidense para la Investigación Sanitaria** [2]. Dicha variable contiene un total de 22 valores únicos, por lo que se ha decidido mantener la variable como numérica.

[1] https://www.kaggle.com/omnamahshivai/surgical-dataset-binary-classification
[2] https://www.hcup-us.ahrq.gov/toolssoftware/ccs10/CCSCategoryNames(FullLabels).pdf

5. *ccsMort30Rate*: **incidencia general de mortalidad a los 30 días por cada intervención (dado por el código de la columna ahrq_ccs)**.
6. *hour*: **hora a la que se realizó la intervención**.
7. *mortality_rsi*: **índice de estratificación de riesgo en la mortalidad a los 30 días**.

CATEGÓRICAS

8. *asa_status*: **estado físico del paciente establecido por la Sociedad Americana de Anestesiología** [3]. Contiene tres categorías:

- 0: **estado I-II** (paciente sano / paciente con enfermedad sistémica leve).
- 1: **estado III** (paciente con enfermedad sistémica grave).
- 2: **estado IV-VI** (paciente con enfermedad muy grave / no espera sobrevivir sin la operación / muerte cerebral).

9. *baseline_cancer*: **¿El paciente padece algún cáncer?** (1 = Sí; 0 = No)
10. *baseline_cvd*: **¿El paciente sufre alguna enfermedad cardio o cerebrovascular?** (1 = Sí; 0 = No)
11. *baseline_dementia*: **¿El paciente sufre algún trastorno por demencia?** (1 = Sí; 0 = No)
12. *baseline_diabetes*: **¿El paciente sufre diabetes?** (1 = Sí; 0 = No)
13. *baseline_digestive*: **¿El paciente sufre alguna enfermedad gastro-intestinal?** (1 = Sí; 0 = No)
14. *baseline_osteoart*: **¿El paciente padece osteoartritis**[4]**?** (1 = Sí; 0 = No)
15. *baseline_psych*: **¿El paciente padece algún desorden psiquiátrico?** (1 = Sí; 0 = No)
16. *baseline_pulmonar*: **¿El paciente sufre alguna enfermedad pulmonar?** (1 = Sí; 0 = No)
17. *dow* o *day of week*: **día de la semana en el que se realizó la intervención** (0 = Lunes; 1 = Martes; 2 = Miércoles; 3 = Jueves; 4 = Viernes).
18. *month*: **mes en el que se realizó la intervención**.
19. *moonphase*: **fase lunar que tuvo lugar durante la intervención quirúrgica** (0 = Luna nueva; 1 = Cuarto creciente; 2 = Luna llena; 3 = Cuarto menguante).
20. *mort30*: **¿El paciente presenta algún riesgo de fallecer a los 30 días?** (1 = Sí; 0 = No)
21. *race*: **raza del paciente** (0 = Caucásico; 1 = Afroameriano; 2 = Otro)

2. Librerías empleadas

A continuación, se expone un listado de las librerías empleadas en el desarrollo del proyecto:

1. *caret*: tuneo de hipérparámetros de los diferentes algoritmos de clasificación.
2. *data.table*: estructura de datos, similar al *data.frame*, aunque mucho más eficiente en memoria.
3. *ggplot2*: librería gráfica.
4. *scorecard*: cálculo del valor de información (IV), así como el peso de la evidencia (WOE).
5. *dummies*: transformación de variables categóricas a *dummies*.
6. *forcats*: tratamiento de variables categóricas.
7. *inspectdf*: libreria para inspeccionar las caracteristicas principales de un *dataset*, incluyendo variables categóricas, valores *missing* o distribución de las variables continuas.

[3] https://www.asahq.org/standards-and-guidelines/asa-physical-status-classification-system
[4] https://dicciomed.usal.es/palabra/osteoartritis

8. *dplyr*: manipulación de datos.
9. *psych*: información general de data.frames y/o data.tables (media, asimetría, desviación típica, entre otros).
10. *doParallel* y *parallel*: paralelización de funciones.
11. *readxl*: lectura de ficheros *Excel* (.xlsx).
12. *visualpred*: visualización de predicciones por diferentes algoritmos de clasificación.
13. *h2o*: *auto Machine Learning (autoML)*.
14. **Librerías y funciones proporcionadas por el profesor**.

```
#--- Librerias
suppressPackageStartupMessages({
  library(caret)          # Data partitioning
  library(data.table)     # Lectura de ficheros mucho mas rapido que read.csv
  library(dplyr)          # Manipulacion de datos
  library(ggplotgui)      # EDA manual mediante entorno interactivo (GUI)
  library(ggplot2)        # Libreria grafica
  library(scorecard)      # Woebin + Woebin_plot + Information Value (IV)
  library(bestNormalize) # Transformacion optima variables continuas
  library(VGAM)           # Aplicacion de transformaciones sobre variables
  library(dummies)        # Creacion variables dummy
  library(psych)          # Informacion estadistica de dataframes
  library(ranger)         # Random Forest (+ rapido que caret)
  library(forcats)        # Tratamiento variables categoricas
  library(inspectdf)      # EDA Automatico (II)
  library(purrr)          # Programacion Funcional

  source("./librerias/librerias_propias.R")
})
```

3. Depuración de los datos

Inicialmente, comenzamos con la lectura del fichero:

```
# Lectura del fichero
surgical_dataset <- fread("./data/Surgical-deepnet.csv", data.table = FALSE)
dim(surgical_dataset) # Filas x columnas
```

```
## [1] 14635    25
```

Nos encontramos con 14.635 observaciones, junto con las 25 variables descritas anteriormente. En primer lugar, **codificamos como *factor* tanto la variable objetivo como el resto de variables categóricas**:

```
# Codificamos como factor la variable objetivo...
surgical_dataset$complication <- as.factor(surgical_dataset$complication)
# ...Asi como el resto de variables categoricas mencionadas anteriormente
cat_columns <- c("gender", "race", "asa_status", "baseline_cancer", "baseline_cvd", "baseline_dementia"
                 "baseline_diabetes", "baseline_digestive", "baseline_osteoart", "baseline_psych",
                 "baseline_pulmonary", "dow", "month", "moonphase", "mort30")
surgical_dataset[,cat_columns] <- lapply(surgical_dataset[, cat_columns], factor)
```

A continuación, almacenamos los nombres de cada variable en un vector por separado, **en función de si es continua o categórica**:

```
# Separamos las variables en numericas, categoricas y target
# [-16] => Salvo la variable objetivo
cat_columns <- names(Filter(is.factor, surgical_dataset))[-16]
num_columns <- names(Filter(is.numeric, surgical_dataset))
target      <- "complication"
```

3.1 Valores NA

Como se puede comprobar a continuación, el *dataset* ___no contiene valores *missing* en ninguna de las variables:

```
sum(is.na(surgical_dataset))
```

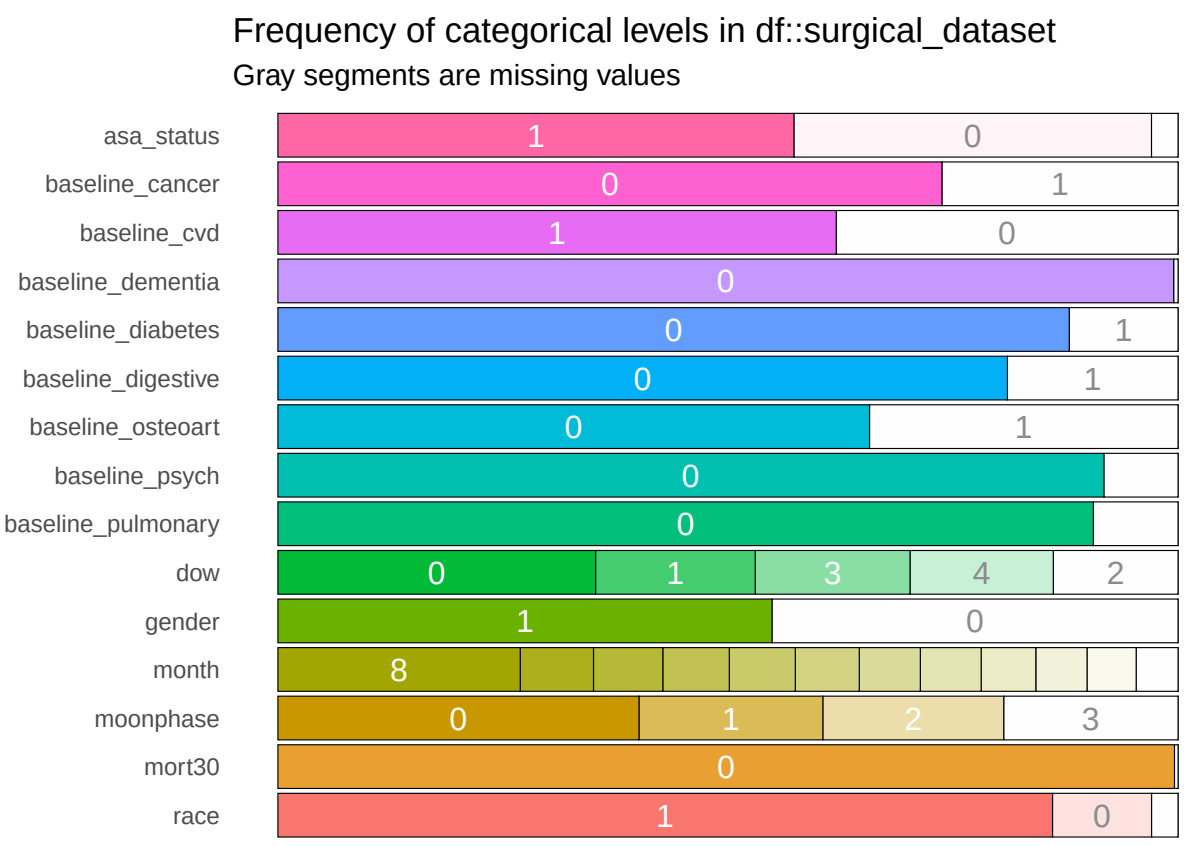
```
## [1] 0
```

3.2 Variables categóricas

Tras almacenar los nombres de cada variable, mediante la librería *inspectdf* se realizó un primer análisis exploratorio de datos automático con el que **analizar el dataset en primera instancia**. Dado que el contenido que el informe es muy extenso, se incluirá en la memoria el contenido esencial (el informe completo se incluye en el anexo *00_EDA_report_with_factors.pdf*).

Sobre dicho informe comenzamos remarcando la frecuencia de aparición de los niveles de cada variable categórica:

```
x <- inspectdf::inspect_cat(surgical_dataset[, cat_columns], include_int = TRUE)
show_plot(x)
```



A simple vista, prácticamente todas las categorías presentan una frecuencia de aparición superior a las 100

observaciones, **salvo por *baseline_dementia* y *mort30***, donde el número de observaciones a 1 es de 71 y 58, respectivamente.

```
surgical_dataset[, c("baseline_dementia", "mort30")] %>% map(table)
```

```
## $baseline_dementia
##
##     0     1
## 14564    71
##
## $mort30
##
##     0     1
## 14577    58
```

Es decir, se tratan de variables con pocas observaciones con valor 1. De hecho, si analizamos el valor de información, haciendo uso del paquete *scorecard*:

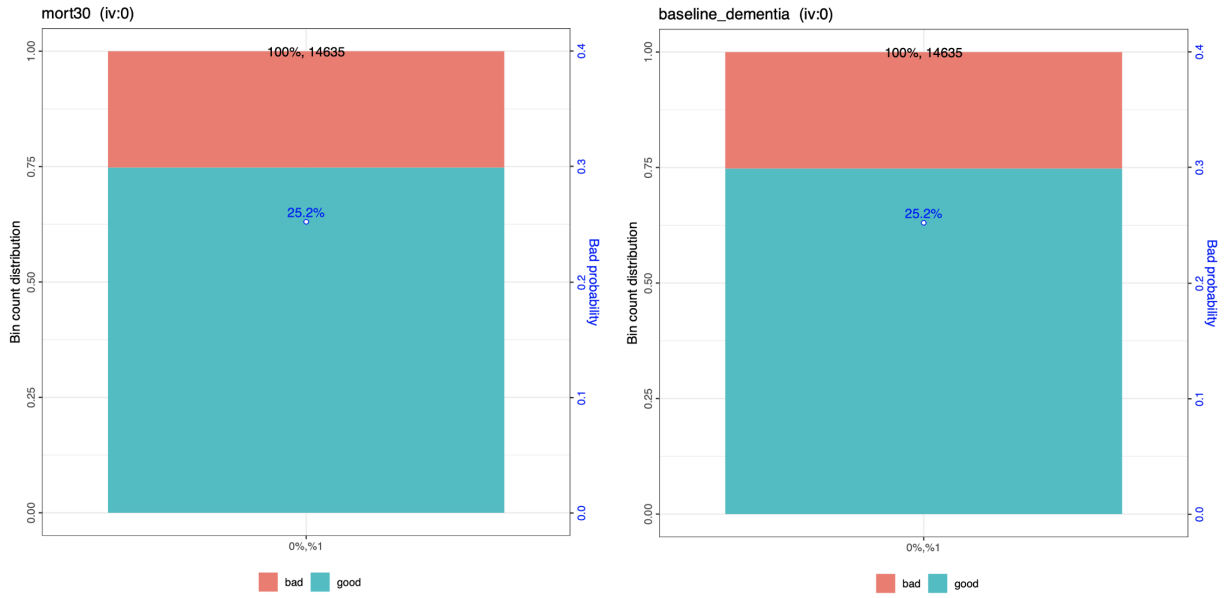


Figure 1: Mort 30 y Baseline dementia (IV)

Observamos que el valor de información es cero, dada la poca representatividad de los valores a 1, de forma que el paquete *scorecard* acaba uniendo ambas categorías, lo que se traduce en un escaso poder predictivo. Por otro lado, si analizamos la proporción de aparición de la variable objetivo sobre cada categoría:

```
#-- baseline_dementia
surgical_dataset %>%
    count(baseline_dementia, complication) %>%
    group_by(complication)
```

```
## # A tibble: 4 x 3
## # Groups:   complication [2]
##   baseline_dementia complication     n
##   <fct>             <fct>        <int>
## 1 0                 0            10913
## 2 0                 1             3651
## 3 1                 0               32
```

```
## 4 1                 1               39
```

```
#-- mort30
surgical_dataset %>%
    count(mort30, complication) %>%
    group_by(complication)
```

```
## # A tibble: 4 x 3
## # Groups:   complication [2]
##   mort30 complication     n
##   <fct>  <fct>        <int>
## 1 0      0            10924
## 2 0      1             3653
## 3 1      0               21
## 4 1      1               37
```

A simple vista, en ambas variables **no existe una clara diferencia entre ambas categorías**. Por tanto, se ha tomado la decisión de descartar ambas columnas del conjunto de datos.

```
surgical_dataset$baseline_dementia <- NULL; surgical_dataset$mort30 <- NULL
```

3.2.1 Agrupación de variables categóricas

Por otro lado, nos encontramos con dos variables cuyas categorías pueden ser agrupadas, según la información proporcionada por el paquete *scorecard*:

DÍA DE LA SEMANA (dow):

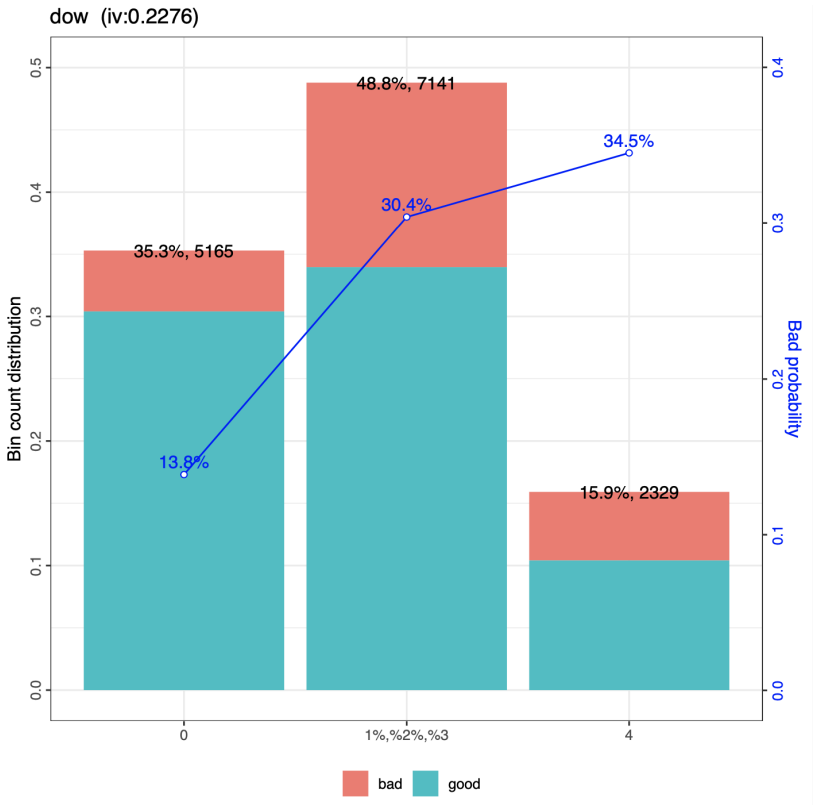
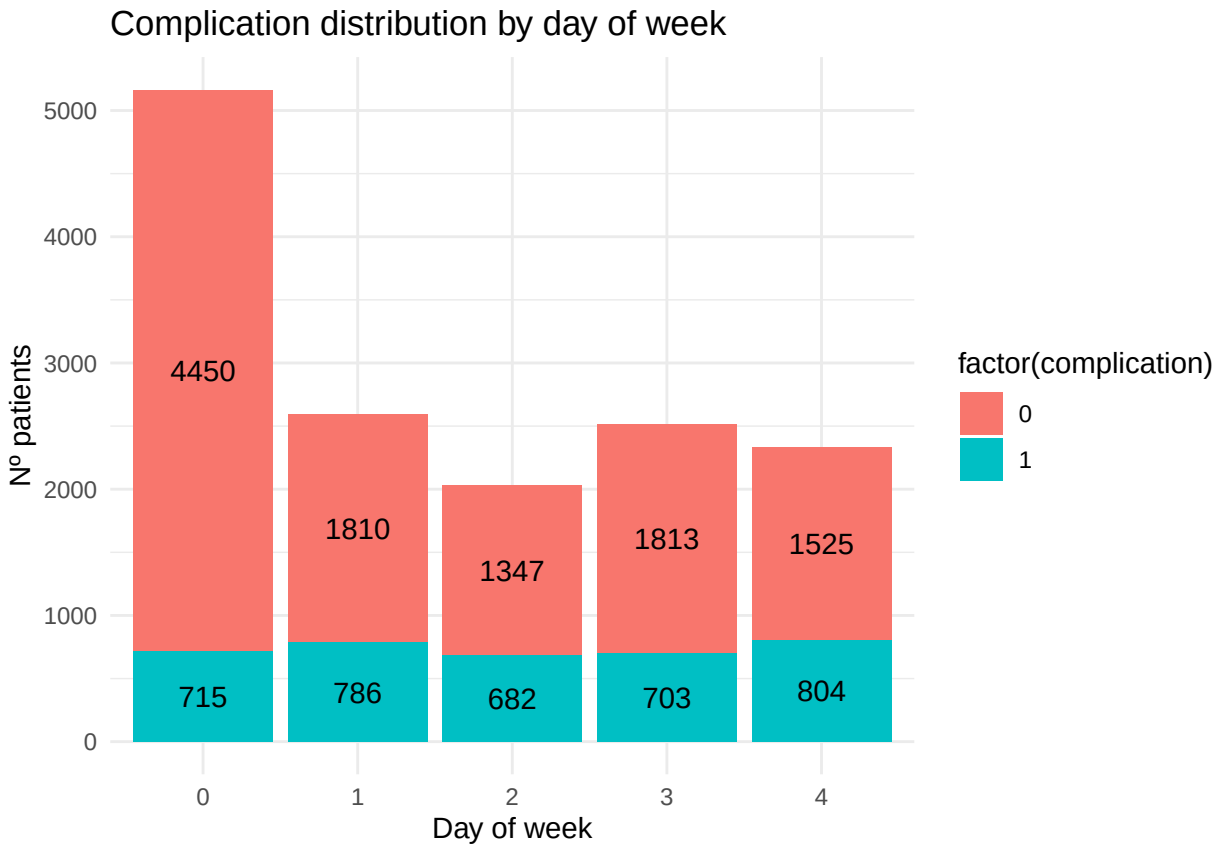


Figure 2: Dia de la semana o dow (IV)

Sobre dicha variable, **observamos una relación "lineal" en la distribución de la variable objetivo a lo largo de los diferentes días de la semana**, comenzando por el Lunes (0), con el menor porcentaje de complicaciones hospitalarias (alrededor del 14 %), seguido de los Martes-Miércoles-Jueves, donde el porcentaje aumenta hasta el 30.4 %, y finalizando con los viernes, donde se alcanza el mayor porcentaje de complicaciones hospitalarias sobre el total: 34.5 %.

Por otro lado, si analizamos detenidamente el gráfico de distribución:



Observamos que la proporción de aparición de pacientes con complicaciones es muy similar entre los martes, miércoles y jueves:

dow	sin.comp	con.comp	total	prop.complicacion
1	4450	715	2596	30.3
2	1810	786	2029	33.6
3	1347	682	2516	27.9
En conjunto (1-2-3)	1813	703	7141	30.4
4	1525	804	2329	34.5

En conjunto, acumulan alrededor del 30.4 % de pacientes con complicaciones, mientras que con tan solo el viernes aumenta hasta alcanzar el 34 %. Por tanto, dado que los martes, miércoles y jueves presentan una proporción de aparición similar, **las agrupamos en torno a una misma categoría**:

1. **Lunes (0)**
2. **Martes-Miercoles-Jueves (1-3)**
3. **Viernes (4)**

MES (month):

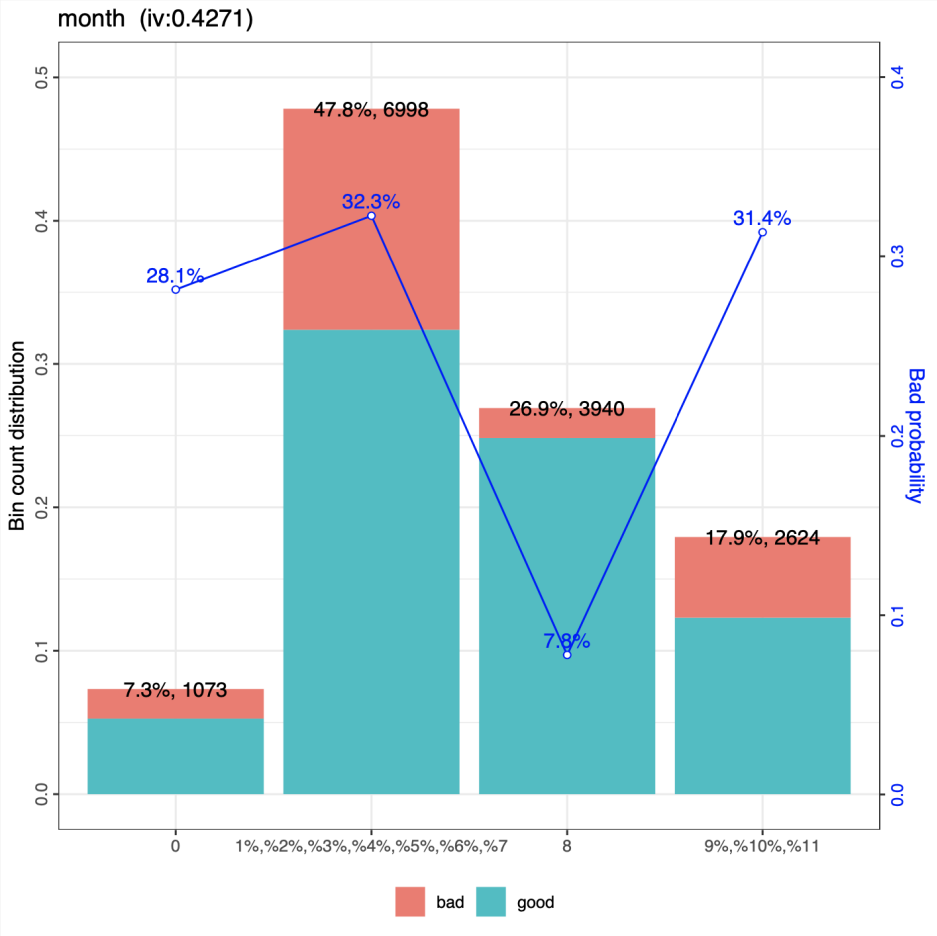
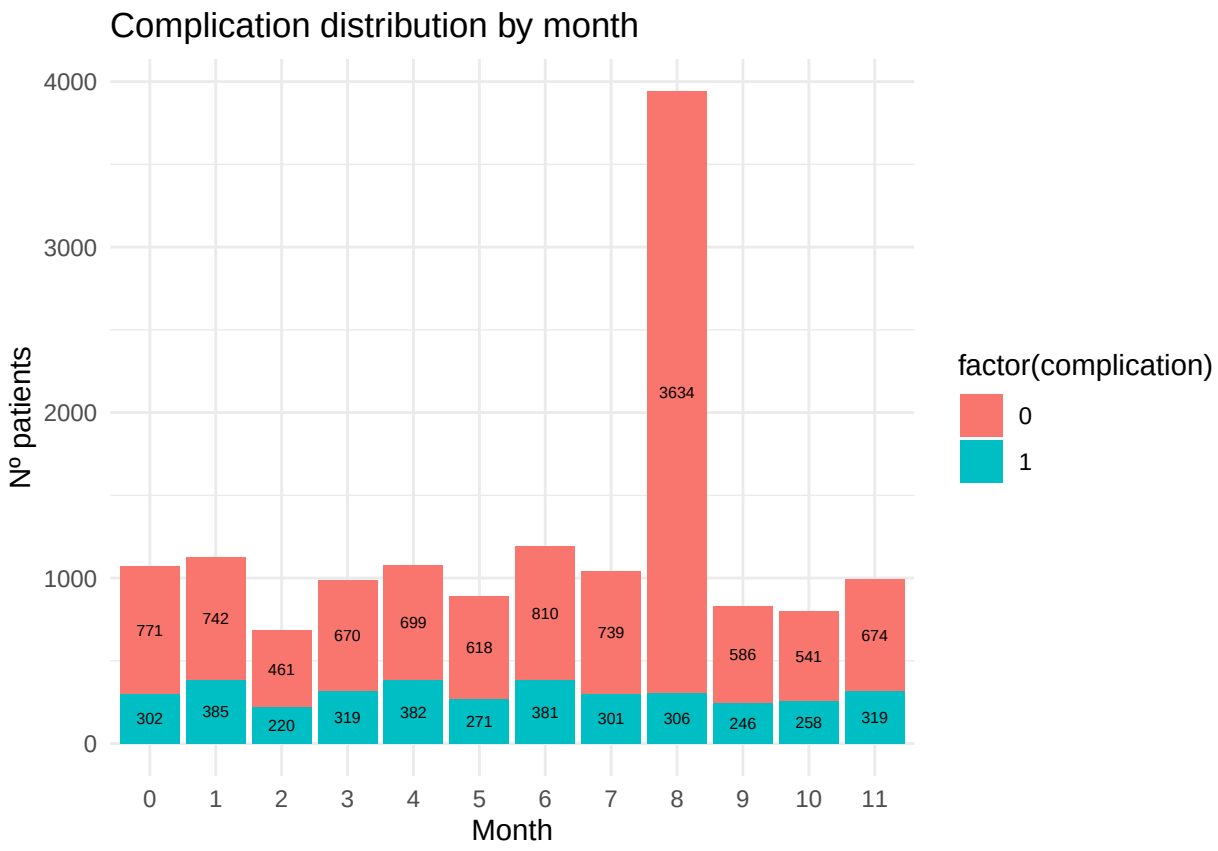


Figure 3: Month (IV)

En este caso, llaman la atención tres principales grupos: en primer lugar el mes de enero (0), con un 28.1 % de las complicaciones hospitalarias, **seguido de los meses de febrero (1) hasta agosto (7) con un total acumulado del 32.3 % de los pacientes con complicaciones**, es decir, el mes de enero tiene un porcentaje similar de pacientes con complicaciones que los siguientes 7 meses en conjunto. Por el contrario, **durante el mes de septiembre (8) el porcentaje se desploma hasta el 7.8 %**, porcentaje que vuelve a aumentar en los tres meses siguientes (octubre, noviembre y diciembre), hasta el 31.4 %.

Por otro lado, si analizamos el gráfico de distribución:



Sucede un comportamiento similar al de la variable *dow*: salvo el mes de septiembre, **la distribución de la variable objetivo sobre cada mes es muy similar, de forma que podemos agrupar varios de los meses en una misma categoría, tal y como hemos comprobado anteriormente**:

1. **Enero (0)**
2. **Febrero a Agosto (1-7)**
3. **Septiembre (8)**
4. **Octubre, Noviembre y Diciembre (9-10-11)**

En relación con el resto de variables categóricas, si analizamos el valor de información de obtenido: