

Bank Marketing Data Set

Author: Ledvir Flores

Introducción:

The Bank marketing dataset tiene 41188 ejemplos con 20 inputs y 1 output variable. La data es de Portugal específicamente de Portuguese Banking Institute. Tiene valores numéricos, así como objetos. La meta es construir un modelo de Machine Learning para predecir si un cliente realiza un depósito a plazo o no. Se realizó un análisis tanto en Python como en R.

Dataset:

El Dataset fue obtenido de UCI Machine Learning Repository y este a su vez lo obtuvo de la dirección de marketing de Portuguese Banking Institution. El dataset está disponible en <http://archive.ics.uci.edu/ml/Bank+Marketing>

La información de este dataset se fue recolectando desde Mayo de 2008 hasta Noviembre de 2010. A continuación se detalla las variables que contiene el dataset:

1. Age - Edad del Cliente - (int64)
2. job - Tipo de trabajo - (object)
3. marital - Estado marital del cliente - (object)
4. education - Nivel de educación - (object)
5. default - ¿Tiene crédito en mora? - (object)
6. housing - ¿Tiene préstamo de vivienda? - (object)
7. loan - ¿Tiene Préstamo personal? - (object)
8. contact - Contacto celular o teléfono - (object)
9. month - Mes de último contacto con el cliente - (object)

Modelo Logístico y Modelo de Selección:

El conjunto de datos del dataset esta limpio, y la variable de respuesta es sí o no según el cliente si se suscribió al deposito a plazo o no. Entonces, se utilizara la Validación cruzada y los árboles de regresión (CART).

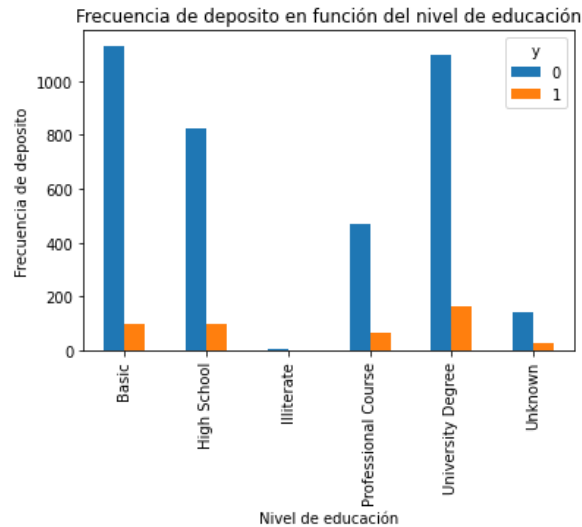


Figura N° 1

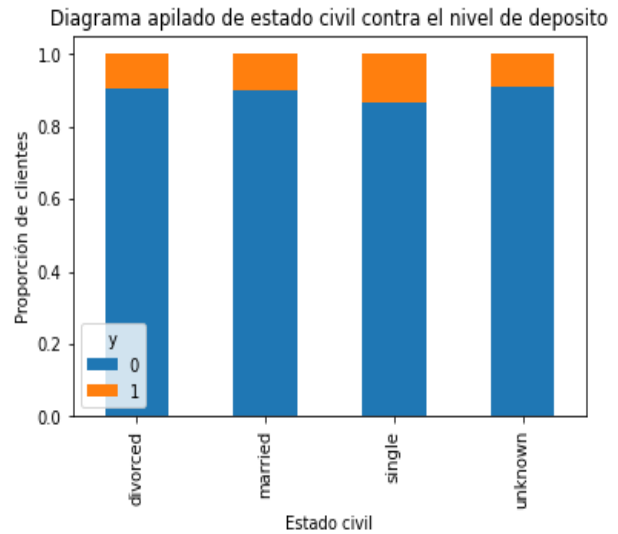


Figura N° 2

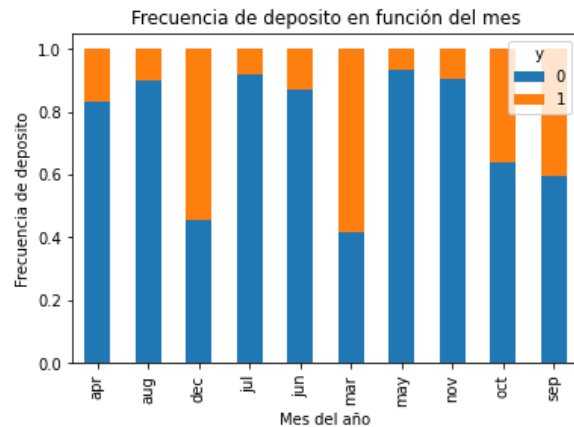


Figura N° 3

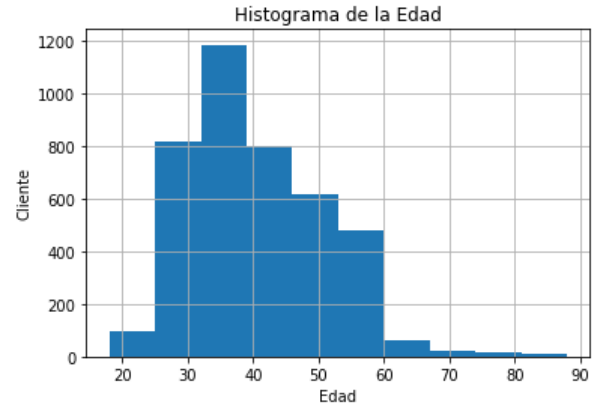


Figura N° 4

Resultados en Python:

Modelo logístico y validación:

Siendo:

- 0 = No
- 1 = Si

Out[45]:

col_0	count
prediction	
0	846
1	390

Personas que NO están dispuestas a realizar un depósito a **plazo fijo 68.45%**

Personas que SI están dispuestas a realizar un depósito a **plazo fijo 31.55 %**

Validación cruzada:

Validando el modelo nos da que está muy próximo a 1 y esta dentro del rango aceptado para un modelo de regresión logística.

```
In [50]: scores
```

```
Out[50]: array([0.90533981, 0.88834951, 0.90533981, 0.89563107, 0.90048544,  
                0.8907767 , 0.88349515, 0.89320388, 0.89320388, 0.88807786])
```

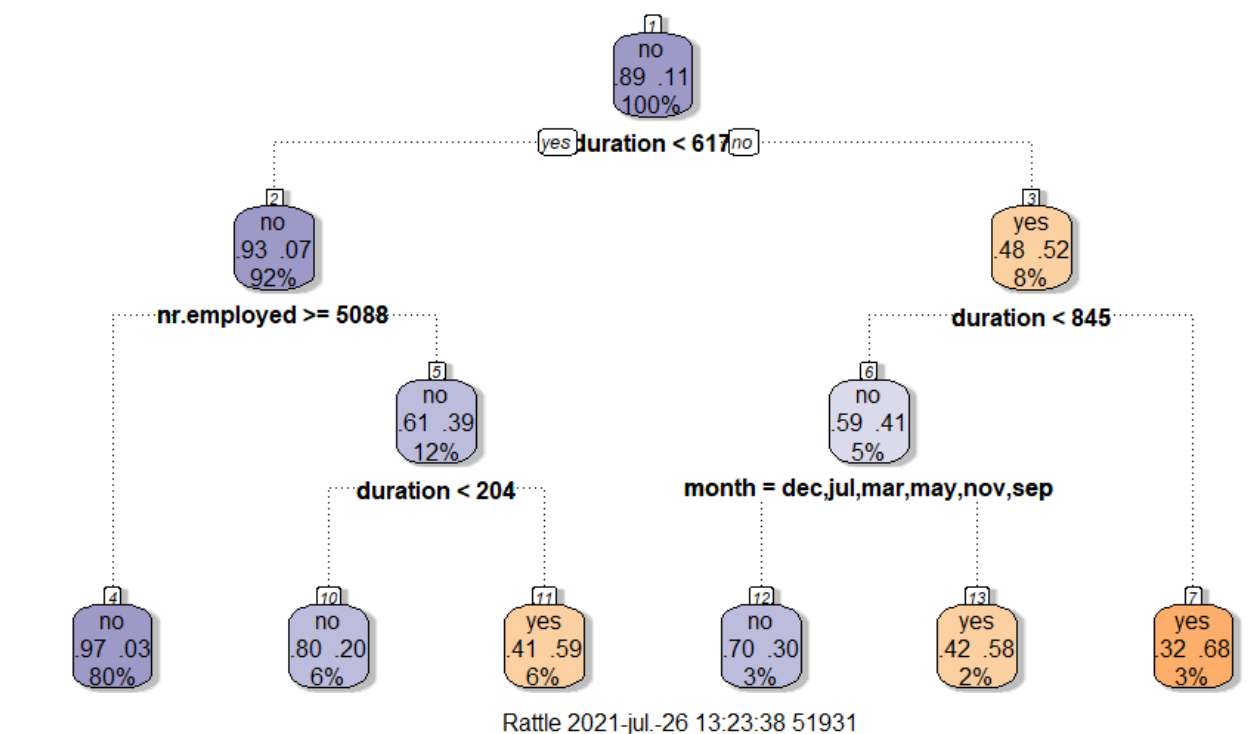
```
In [51]: scores.mean()
```

```
Out[51]: 0.8943903101599225
```

Resultados en R:

Árbol de Decisión

- Accuracy (Exactitud) : 0.9125
- Misclassification Rate (Tasa de error) : 0.0875
- Sensibility (Sensibilidad) : 0.9625
- Specificity (Especificidad) : 0.5222



	Predicho	
Actual	no	yes
no	1045	55
yes	56	79

El porcentaje de exactitud es de 91.25 % y la tasa de error es de 8.75%

Conclusiones:

- Para realizar alguna campaña de marketing el público objetivo sería entre 25 a 60 años siendo los 32 a 39 años el público de más cantidad (Ver Figura N°4).
- En cuanto al mes del año en el que se debería realizar las campañas serían en el mes de Marzo y Diciembre, siendo el público objetivo entre las edades de 32 a 39 años.(Ver Figura N° 3)

- Según la Figura N° 1 y N° 2 las personas que mas depósitos a plazo fijo realizan son las que cuentan con estudios Universitarios y las personas solteras.
- Del modelo logístico se puede concluir que las personas que quieren poner su dinero a plazo fijo tienen un porcentaje de 31.55 %. Este porcentaje se vería incrementado si se realiza una adecuada campaña de marketing bien segmentado como se detalla en los puntos anteriores.
- En cuanto al árbol de decisión se tiene una exactitud del 91.25% y una tasa de error del 8.75% con el modelo. De eso se puede concluir que 9 de cada 10 predicciones que nos da el modelo respecto a quienes realizaran un depósito a plazo fijo se acertadas teniendo solo 1 persona que no se le podrá categorizar de forma adecuada.

Referencias:

Si desean ver mas detallado todos los cálculos los invito a revisar mi github www.github.com/ledvir26 donde esta más detallado los modelos de regresión logística la validación cruzada que se utilizo para verificar si el modelo es adecuado. Estan los 2 codigos tanto en R como en Python.