Evaluación 2 - Minería de Datos

Pablo Herrera, Felipe Sánchez

2022-07-23

Contexto

El objetivo de este trabajo es utilizar dos algoritmos de clasificación: Árboles de Clasificación y K Vecino más Cercano (KNN). Se cuenta con una base de datos que contiene 1.000 registros con la evaluación de riesgo crediticio de un banco alemán.

Se busca generar modelos predictivos para la evaluación de riesgo (contenida en la variable binaria *credit*).

División de la base de datos.

El total de 1.000 observaciones es dividido entre dos sets de datos, una parte mayoritaria será utilizada para el entrenamiento de los modelos con el 75% de las observaciones, la otra parte será utilizada para la validación de los modelos entrenados con el 25% restante. Para obtener un análisis replicable, se utiliza una semilla de valor 123.

Con el objetivo de chequear que ambos sets de datos estén balanceados en sus características, se calculan las medias de algunas variables en los dos sets de datos creados. Se observa que no existen mayores diferencias (ver Tabla 1).

Table 1: Ambos sets de datos están balanceados en las medias de sus variables.

Variable	Entrenamiento	Testeo
credit	0.7	0.7
amount	3197.0	3494.1
property	2.3	2.4
age	35.3	36.2
employed	3.4	3.4

Árboles de Clasificación.

El primer paso en la estimación de los modelos de árboles de clasificación fue la elección del set de variables. Como primer criterio en la elección se define considerar al menos una variable por cada una de las categorías mencionadas en el Anexo 1 (de esta manera el modelo contendrá al menos una variable con información bancaria, patrimonial, personal y laboral). Luego de esto se estimaron las correlaciones entre la variable *credit* y cada grupo de variables explicativas, de manera que se escogieron las que presentaban mayor correlación dentro de cada categoría. Adicionalmente la variable *foreign* fue ignorada debido a que tiene muy pocas observaciones con $foreign = 1$ (tan solo 37 de las 1.000) lo que podría ser un problema debido a las restricciones utilizadas para la cantidad de observaciones en los nodos.

Como resultado, se define considerar las siguientes variables: *amount, history, status, property, savings, age, personal* y *employed*. Posteriormente se realizan transformaciones a las 8 variables mencionadas. En algunos

casos las variables son transformadas a variables binarias y en otros casos simplemente se reducen los posibles valores y/o son reordenadas (ver detalle en Anexo 2).

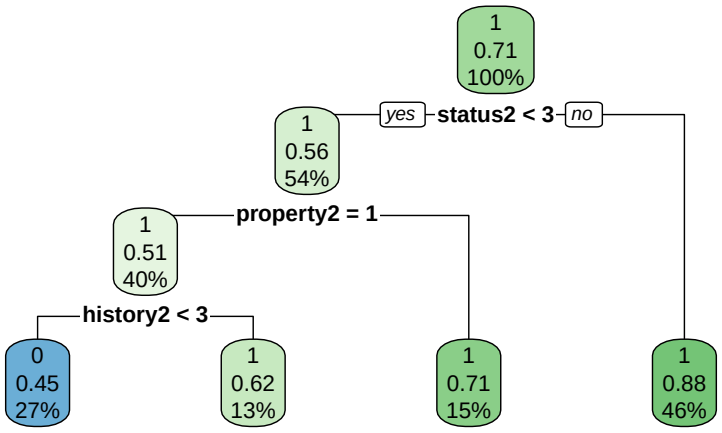
A continuación se estiman distintos modelos con distintas definiciones y posteriormente se analizan los cambios que estas modificaciones causan en los resultados.

Modelo 1. Modelo base.

El primer modelo considera las siguientes definiciones:

- **Árbol de clasificación.** Dado que la variable dependiente es la evaluación crediticia y esta es binariam el modelo en ningún caso será un Árbol de Regresión.
- **Método de división.** En este primer modelo se utiliza el de Criterio de Ganancia de Información. Más adelante se analizarán los resultados al utilizar el Índice de Impuridad de Gini.
- **Parámetro de complejidad.** El cp (por sus siglas en inglés) inicialmente tendrá valor 0.009. Este valor se obtuvo estimando un modelo preliminar con $cp = 0$, para luego analizar distintos valores cp y sus respectivos errores usando la función ***printcp***. Con esta se observa que el menor error de validación cruzada (o *xerror*) se obtiene con un $cp = 0.009$. Cabe mencionar que el paso de un cp=0 a un cp=0.009, tiene como consecuencia la disminución de niveles de profundidad (de 5 a 3). Más adelante se analizarán los resultados al utilizar otros valores del parámetro.
- **Controles adicionales.** Más adelante se analizarán los cambios en los resultados producto de utilizar otros valores para estos controles.
  - **Min. de obs. para dividir un nodo**: 40 observaciones.
  - **Min. de obs. para tener en un nodo terminal**: 30 observaciones.
  - **Profundidad max. que tendrá el modelo**: 5 niveles.

Árbol Modelo 1.



Notar que los árboles graficados, contienen información en cada uno de sus nodos. Estos indican la categoría mayoritaria en cada uno (si la evaluación crediticia fue negativa, es decir 0, o positiva, es decir 1), la fracción de personas que tienen una evaluación positiva según la muestra en ese nodo, y el porcentaje de la muestra de 750 personas que forman parte del nodo.

Table 2: Matriz de confusión en base de entrenamiento, Modelo 1.

	Predicho 0	Predicho 1
Real 0	111	109
Real 1	91	439

Table 3: Matriz de confusión en base de validación, Modelo 1.

	Predicho 0	Predicho 1
Real 0	44	36
Real 1	29	141

La precisión dentro de la base de entrenamiento es de: **73.33**%. La precisión dentro de la base de validación es de: **74**%.

Modelo 2. Se modifica el método de división.

El segundo modelo considera las siguientes definiciones:

- Se modifica el método de división pasando desde el Criterio de Ganancia de Información del Modelo 1, al uso del **Índice de Impuridad de Gini** en este segundo modelo.
- El resto de los parámetros no son modificados.

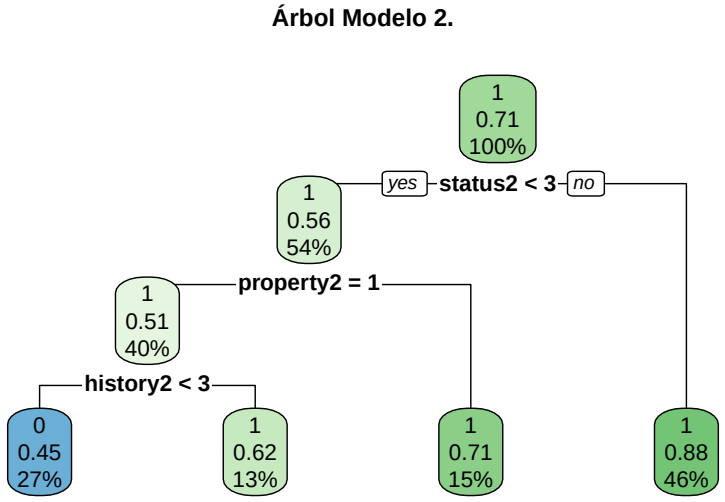


Table 4: Matriz de confusión en base de entrenamiento, Modelo 2.

	Predicho 0	Predicho 1
Real 0	111	109
Real 1	91	439

Table 5: Matriz de confusión en base de validación, Modelo 2.

	Predicho 0	Predicho 1
Real 0	44	36
Real 1	29	141

La precisión dentro de la base de entrenamiento es de: **73.33**%. La precisión dentro de la base de validación es de: **74**%.

Modelo 3. Se modifica el parámetro del costo de complejidad.

El tercer modelo considera las siguientes definiciones:

- Con respecto al Modelo 1, se modifica el valor del parámetro de complejidad. Se pasa de un cp 0.009 (utilizado en los modelos 1 y 2), a un valor de 0.002.
- El resto de los parámetros del Modelo 1 no son modificados.

Árbol Modelo 3.

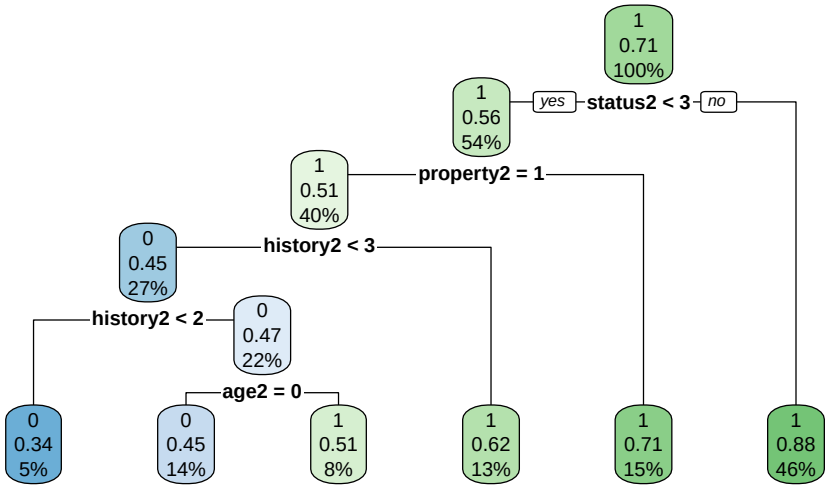


Table 6: Matriz de confusión en base de entrenamiento, Modelo 3.

	Predicho 0	Predicho 1
Real 0	81	139
Real 1	60	470

Table 7: Matriz de confusión en base de validación, Modelo 3.

	Predicho 0	Predicho 1
Real 0	34	46
Real 1	19	151

La precisión dentro de la base de entrenamiento es de: **73.47**%. La precisión dentro de la base de validación es de: **74**%.

Modelo 4. Se modifican los valores de los controles adicionales.

El cuarto modelo considera las siguientes definiciones:

- Con respecto al Modelo 1, se modifican los Controles Adicionales.
  - **Min. de obs. para dividir un nodo**: se pasa de 40 a 30 observaciones.
  - **Min. de obs. para tener en un nodo terminal**: se pasa de 30 a 15 observaciones.
  - **Profundidad max. que tendrá el modelo**: se pasa de 5 a 6 niveles.

Árbol Modelo 4.

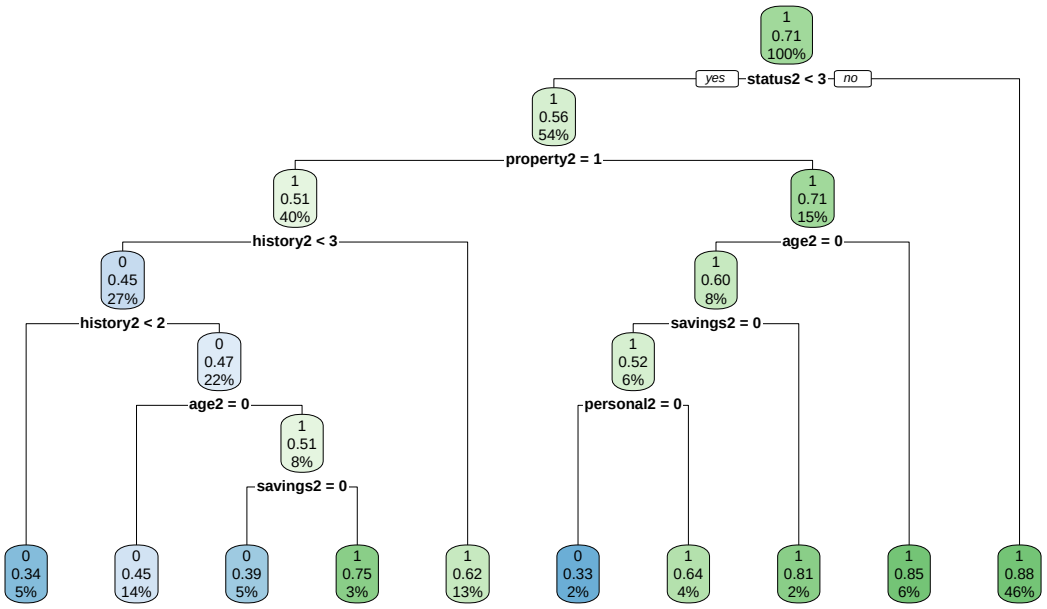


Table 8: Matriz de confusión en base de entrenamiento, Modelo 4.

	Predicho 0	Predicho 1
Real 0	118	102
Real 1	82	448

Table 9: Matriz de confusión en base de validación, Modelo 4.

	Predicho 0	Predicho 1
Real 0	44	36
Real 1	27	143

La precisión dentro de la base de entrenamiento es de: **75.47**%. La precisión dentro de la base de validación es de: **74.8**%.

Interpretación y comparación de modelos.

Modelo 1.

En primer lugar cabe destacar que el cp de 0.009 limita la cantidad de divisiones (tan sólo se generan 3 nodos, cuando máximo tolerado es de 5). En cuanto a la capacidad predictiva de este primer modelo los resultados son de un 73% en la base de entrenamiento, y de 74% en la base de validación. Dentro de las variables utilizadas, el primer nodo toma la variable transformada *status2*. Dicho nodo identifica a las personas que tienen algún tipo de ahorro, abarcando un 46% de la muestra de entrenamiento. En este primer nodo final un 88% de los casos efectivamente es evaluado positivamente. El resto de los nodos finales logran menor precisión con un 71, 62 y 55% cada uno. Además de *status2*, los otros nodos utilizados son *property* en el segundo nivel que separa a las personas que tienen alguna propiedad y *history2* que hace la diferencia entre las personas que tienen retrasos en el pasado, cuenta crítica o créditos en otros bancos, o que no tienen créditos anteriores, y las personas que tienen créditos vigentes al día o todos sus créditos pagados.

Modelo 2.

En este modelo, el único cambio realizado es pasar del método de definición por ganacia de información al índice de impuridad de gini. Sin embargo, no se observan cambios y los resultados son prácticamente los mismos que los del Modelo 1.

Modelo 3.

En este modelo, en único cambio con respecto al Modelo 1 es la disminución a menos de un tercio del parámetro de complejidad pasando de un 0.009 a un 0.002. Es posible ver que en este caso, la restricción de los niveles aceptados (5) se activa. Los primeros 4 nodos finales del Modelo 1 se mantienen y el nodo que en el Modelo 1 acumulaba el 27 de los casos, genera dos niveles adicionales. En cuanto a su precisión, el Modelo funciona levemente mejor en el set de entrenamiento (pasando de un 73 a un 74% de precisión), y funciona igual en el set de validación.

Modelo 4.

En este último modelo, se generan cambios en los controles adicionales mencionados en el Modelo 1. Se disminuye la cantidad mínima de observaciones para dividir un nodo (de 40 a 30), se disminuye la cantidad mínima de observaciones en un nodo terminal (de 30 a 15) y se tolera una mayor profundidad (pasando de un máximo de 5 a 6). De la misma manera que en el Modelo 3, este cuarto modelo sigue ahondando en el último nodo final del Modelo 1, el cual acumulaba 27% de las observaciones. Además se alcanza el máximo tolerado de profundidad. La precisión del modelo aumenta levemente, alcanzado una precisión de 76 y 75% en el set de entrenamiento y validación, respectivamente.

K vecinos más cercanos (KNN).

Método conocido como KNN por sus siglas en inglés (K nearest neighbours).

Para la confección de este modelo, no se tendrá en consideración las transformaciones de las variables mencionadas anteriormente. Esto porque ahora estamos interesados en relaciones de distancias bajo alguna métrica establecida, y no en decisiones binarias como lo hacen los árboles de desición (La reducción de categorías establecidas conservaban la "distribución" de 1 y 0 en la variable target *credit*).

La selección de variables para la elaboración del modelo respetará la misma lógica utilizada en la sección anterior, optando por usar las variables clasificadoas:

- **Información Bancaria:** amount, history, status.
- **Patrimonial:** property, savings.
- **Personal:** Age, personal.
- **Laboral:** employed.

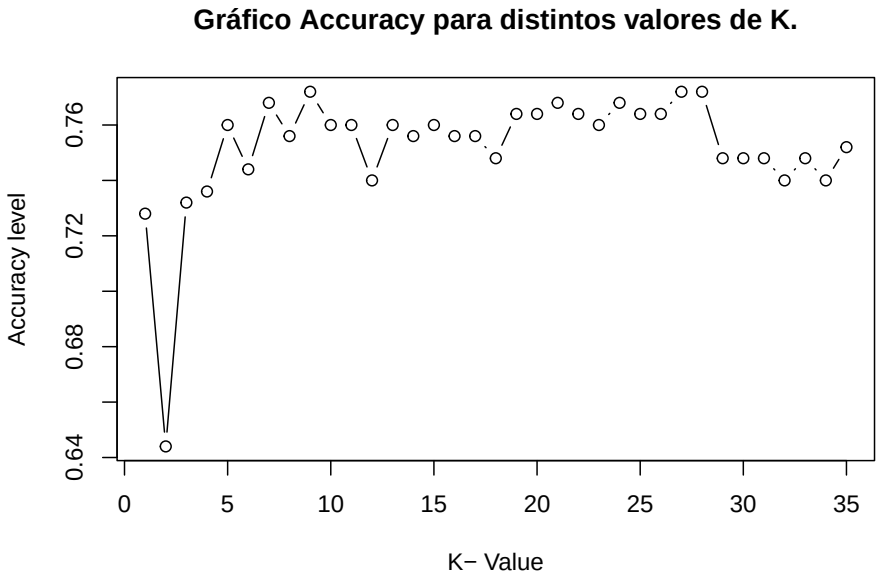
Dentro de las cuales adaptaremos *history*, *status* y *personal* por ser de naturaleza categórica a variables dummy bajo el siguiente esquema:

1) *status*: Se crean dos nuevas columnas, "cuenta con deuda" y "cuenta con ahorros", identificando si la persona se encuentra en estado de deuda o de ahorros (Sin cuenta en su defecto).

2) *history*: Colapsaremos las categorías 0 y 1, junto con 3 y 4 para generar dos nuevas columnas "Retrasos" y "NoRetraso" respectivamente (Sin créditos en su defecto categ. 2).

3) *personal*: En función de la cantidad de registros por categorías, se juntan las categorías 1 y 4 para crear dos nuevas columnas "Hs/Mns" (hombre soltero/ mujer no soltera :categ. 2) y "Hc" (Hombre casado o viudo). Hombre divorciado o mujer soltera pars los otros casos.

Luego de modificar nuestras variables para poder aplicar el algoritmo KNN, normalizamos nuestras variables para que las distancias no se vean afectadas por las escalas de las variables y separamos la data en 75% destinada a entrenamiento y 25% como datos de validación del modelo con una semilla pseudo-aleatoria de 123.



Tras varias repeticiones del ejercicio anterior, se identifica que los k = 9, 25 y 27 entregan consistemente altos niveles de precisión. Por lo tanto se estimarán tres modelos con estos valores. Notar que estos valores al ser impares evitan decisiones al azar.

Modelo 1. Utilizando 27 vecinos.

```
## Confusion Matrix and Statistics
##
##           Reference
## Prediction   0   1
##          0  38  15
##          1  42 155
##
##                Accuracy : 0.772
##                  95% CI : (0.7149, 0.8225)
##     No Information Rate : 0.68
##     P-Value [Acc > NIR] : 0.0008644
##
```

```
##                   Kappa : 0.4247
##
##  Mcnemar's Test P-Value : 0.0005736
##
##             Sensitivity : 0.4750
##             Specificity : 0.9118
##          Pos Pred Value : 0.7170
##          Neg Pred Value : 0.7868
##              Prevalence : 0.3200
##          Detection Rate : 0.1520
##    Detection Prevalence : 0.2120
##       Balanced Accuracy : 0.6934
##
##        'Positive' Class : 0
##
```

Modelo 2. Utilizando 25 vecinos.

```
## Confusion Matrix and Statistics
##
##           Reference
## Prediction   0   1
##          0  39  18
##          1  41 152
##
##                Accuracy : 0.764
##                  95% CI : (0.7064, 0.8152)
##     No Information Rate : 0.68
##     P-Value [Acc > NIR] : 0.002220
##
##                   Kappa : 0.4131
##
##  Mcnemar's Test P-Value : 0.004181
##
##             Sensitivity : 0.4875
##             Specificity : 0.8941
##          Pos Pred Value : 0.6842
##          Neg Pred Value : 0.7876
##              Prevalence : 0.3200
##          Detection Rate : 0.1560
##    Detection Prevalence : 0.2280
##       Balanced Accuracy : 0.6908
##
##        'Positive' Class : 0
##
```

Modelo 3. Utilizando 9 vecinos.

```
## Confusion Matrix and Statistics
##
##           Reference
## Prediction   0   1
##          0  39  16
##          1  41 154
```

```
##
##                Accuracy : 0.772
##                  95% CI : (0.7149, 0.8225)
##     No Information Rate : 0.68
##     P-Value [Acc > NIR] : 0.0008644
##
##                   Kappa : 0.4289
##
##  Mcnemar's Test P-Value : 0.0014785
##
##             Sensitivity : 0.4875
##             Specificity : 0.9059
##          Pos Pred Value : 0.7091
##          Neg Pred Value : 0.7897
##              Prevalence : 0.3200
##          Detection Rate : 0.1560
##    Detection Prevalence : 0.2200
##       Balanced Accuracy : 0.6967
##
##        'Positive' Class : 0
##
```

Interpretación y comparación de modelos.

Considerando que la clase positiva corresponde a una evaluación negativa del riesgo crediticio, observamos en todos los modelos mejores resultados para la clase negativa, lo cual es concordante con la mayor presencia de observaciones con evaluación de riesgo crediticio positiva ($credit = 1$). Es por esto que el criterio utilizado para la selección del modelo final será Balanced Accuracy, con el objetivo de "maximizar" la precisión de la clasificación de ambas clases.

Ante este análisis, se escoge el modelo con 9 vecinos al tener el mayor valor tanto en Balanced Accuracy como en Accuracy.

Comparación entre los resultados de ambos algoritmos.

Table 10: Mat. confusión (set de validación). Árbol, Modelo 4.

	Predicho 0	Predicho 1
Real 0	44	36
Real 1	27	143

Table 11: Mat. confusión (set de validación). KNN, Modelo 3.

	Predicho 0	Predicho 1
Real 0	39	41
Real 1	16	154

Ya identificando a simple vista que el valor de accuracy del modelo de k vecinos cercanos es mayor al otorgado por el árbol de clasificación en un 1,73%, eligiríamos el algoritmo KNN para la clasificación de crédito sobre esta base.

Además, es posible observar que la "accuracy" entregada por el algoritmo de árbol de clasificación se mantiene por debajo de los modelos de K vecinos cercanos. Una posible causa de este fenómeno es la no-homogeneidad de la clase a predecir (700 1-registros y 300 0-registros) ya que mientras el árbol se encarga de realizar segmentaciones a las variables, KNN mide solo distancias entre el punto de interés y los vecinos especificados por nuestro K.

Anexos.

Anexo 1. Variables disponibles.

Se cuenta con un conjunto de 17 variables. Una es la variable *credit*, que será la variable endógena de los modelos y otras 16 son variables explicativas. Estas son organizadas dependiendo si corresponden a información bancaria, patrimonial, personal o laboral.

Evaluación de riesgo:

- ***credit***: evaluación de riesgo (binaria).

Información bancaria:

- ***amount***: monto del crédito.
- ***credits***: cantidad de créditos vigentes (ordinal).
- ***history***: historial de créditos (categórica).
- ***rate***: tasa del crédito en proporción del ingreso (ordinal).
- ***status***: historial de cuenta corriente (categórica).

Información patrimonial:

- ***housing***: relación de propiedad con su vivienda (categórica).
- ***property***: bien más valioso (ordinal).
- ***savings***: historial de ahorros (ordinal).

Información personal:

- ***age***: edad de la persona.
- ***foreign***: extranjero (binaria).
- ***personal***: estado civil y sexo (categórica).
- ***persons***: carga familiar (binaria).
- ***residence***: tiempo en la residencia (ordinal).
- ***telephone***: tiene teléfono fijo (binaria).

Información laboral:

- ***employed***: tiempo empleado (ordinal).
- ***job***: nivel de empleabilidad (categórica).

Anexo 2. Transformación de variables utilizadas en Árboles de Decisión.

Las 8 variables consideradas en los modelos de Árboles de Decisión son transformadas con las siguientes consideraciones:

- ***amount***: se genera una variable binaria que reconoce quiénes están sobre la mediana de la muestra (considerando las 1.000 observaciones).
- ***history***: se reducen los posibles valores, pasando de 5 a 3 niveles (niveles 0 y 1 se concentran en uno, los niveles 3 y 4 también y el nivel 2 se mantiene solo).
- ***status***: se reducen los posibles valores, pasando de 4 a 3 niveles. Los dos niveles superiores se concentran en uno. Además, se reordenan las categorías: con deuda, sin cuenta, con ahorros.
- ***property***: se genera una variable binaria que reconoce si la persona tiene alguna propiedad.
- ***savings***: se genera una variable binaria que reconoce si la persona tiene ahorros.
- ***age***: se genera una variable binaria que reconoce quiénes están sobre la mediana de la edad.
- ***personal***: se identifica que los grupos 3 y 4 tiene en promedio mejor evaluación crediticia que los grupos 1 y 2. Se genera una variable binaria que reconoce quiénes pertenecen a los grupos 3 y 4.
- ***employed***: se genera una variable binaria que reconoce llevan más de un año trabajando.