Benemérita Universidad Autónoma de Puebla
Facultad de Ciencias de la Computación

Máquinas de aprendizaje

Reporte: Problema 2 con k-NN

Docente: Abraham Sánchez López

Alumno	Matrícula
Taisen Romero Bañuelos	202055209

K-NN y la odisea chupasangre de los bancos

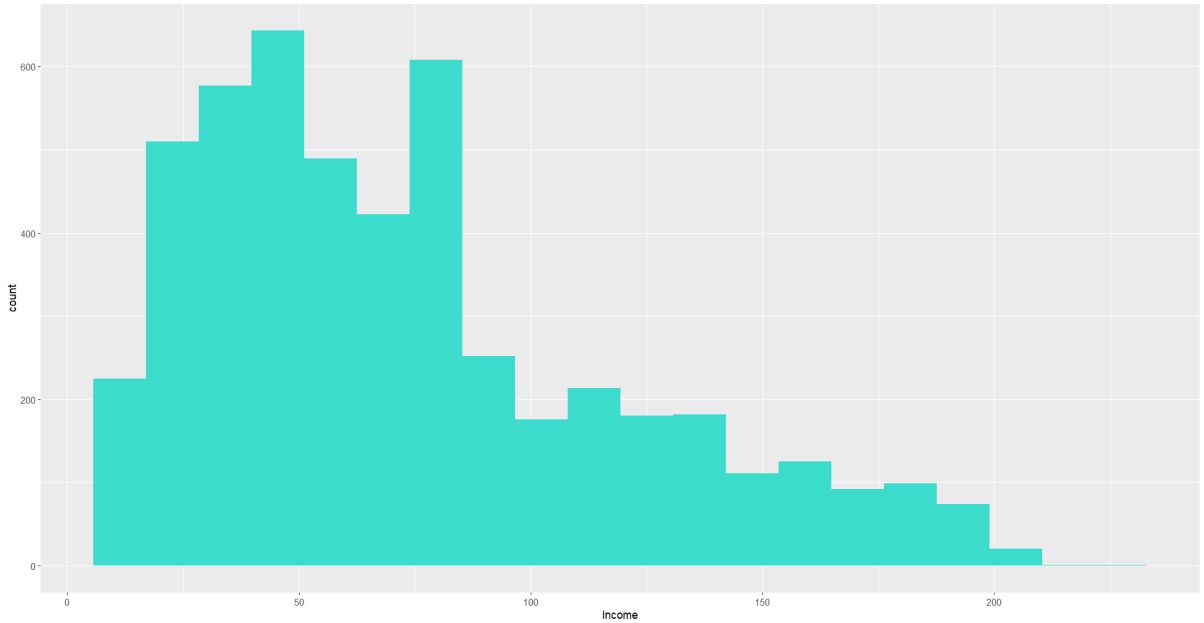
Bien, nuestro objetivo es utilizar k-NN para predecir si un nuevo cliente aceptará una oferta de préstamo. Pero primero hay que explorar los datos para hacernos una idea del material con el que hay que trabajar.

Empecemos viendo qué nos dice la función summary() y de paso veamos si hay valores nulos en el dataset.

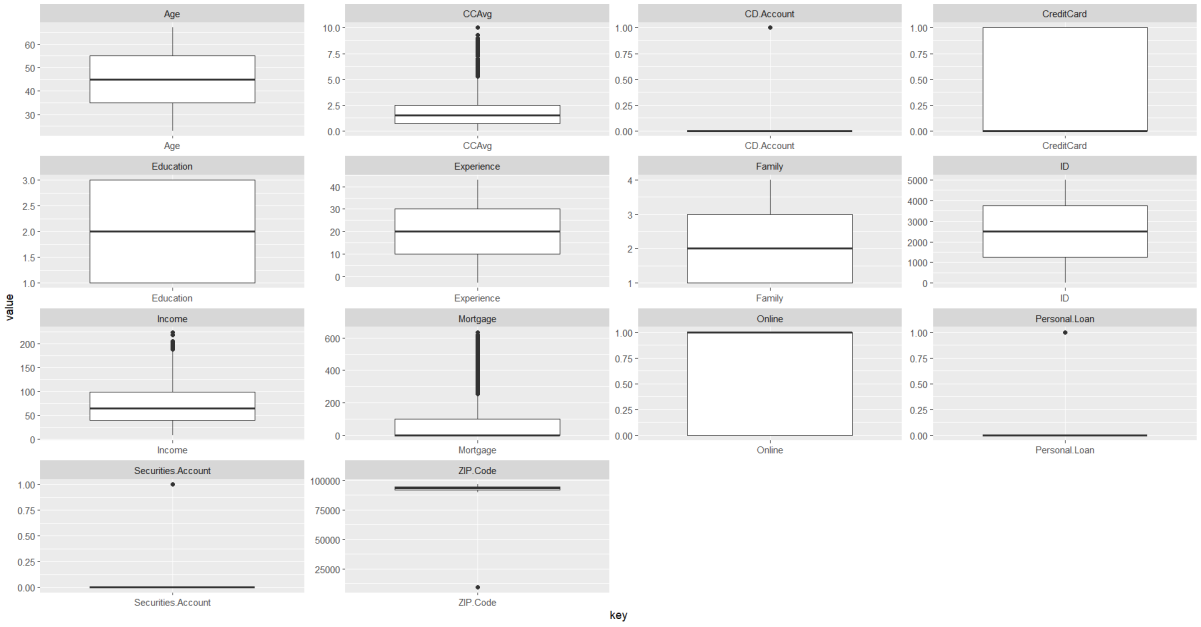
```
> summary(data)
       ID              Age            Experience       Income          ZIP.Code         Family          CCAvg            Education     
 Min.   :   1   Min.   :23.00   Min.   :-3.0   Min.   :  8.00   Min.   : 9307   Min.   :1.000   Min.   : 0.000   Min.   :1.000  
 1st Qu.:1251   1st Qu.:35.00   1st Qu.:10.0   1st Qu.: 39.00   1st Qu.:91911   1st Qu.:1.000   1st Qu.: 0.700   1st Qu.:1.000  
 Median :2500   Median :45.00   Median :20.0   Median : 64.00   Median :93437   Median :2.000   Median : 1.500   Median :2.000  
 Mean   :2500   Mean   :45.34   Mean   :20.1   Mean   : 73.77   Mean   :93153   Mean   :2.396   Mean   : 1.938   Mean   :1.881  
 3rd Qu.:3750   3rd Qu.:55.00   3rd Qu.:30.0   3rd Qu.: 98.00   3rd Qu.:94608   3rd Qu.:3.000   3rd Qu.: 2.500   3rd Qu.:3.000  
 Max.   :5000   Max.   :67.00   Max.   :43.0   Max.   :224.00   Max.   :96651   Max.   :4.000   Max.   :10.000   Max.   :3.000  
    Mortgage      Personal.Loan    Securities.Account   CD.Account        Online         CreditCard     
 Min.   :  0.0   Min.   :0.000   Min.   :0.0000     Min.   :0.0000   Min.   :0.0000   Min.   :0.000  
 1st Qu.:  0.0   1st Qu.:0.000   1st Qu.:0.0000     1st Qu.:0.0000   1st Qu.:0.0000   1st Qu.:0.000  
 Median :  0.0   Median :0.000   Median :0.0000     Median :0.0000   Median :1.0000   Median :0.000  
 Mean   : 56.5   Mean   :0.096   Mean   :0.1044     Mean   :0.0604   Mean   :0.5968   Mean   :0.294  
 3rd Qu.:101.0   3rd Qu.:0.000   3rd Qu.:0.0000     3rd Qu.:0.0000   3rd Qu.:1.0000   3rd Qu.:1.000  
 Max.   :635.0   Max.   :1.000   Max.   :1.0000     Max.   :1.0000   Max.   :1.0000   Max.   :1.000  
> 
```

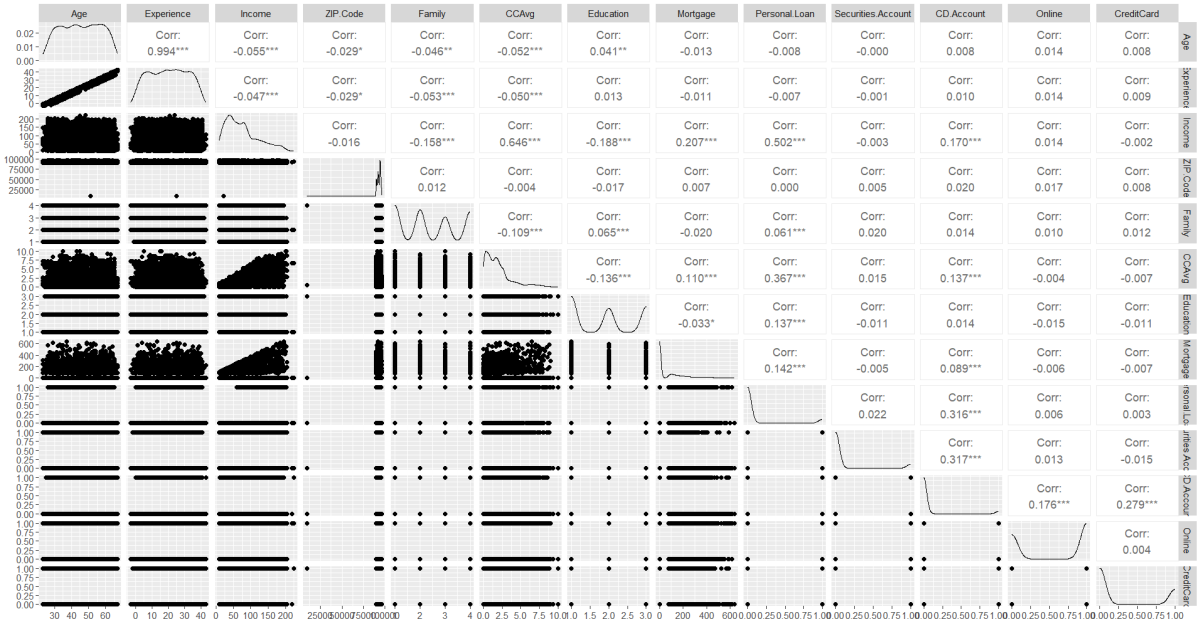
```
> #Búsqueda de datos nulos
> colSums(is.na(data))
                ID                Age         Experience             Income           ZIP.Code             Family 
                 0                  0                  0                  0                  0                  0 
             CCAvg          Education           Mortgage      Personal.Loan Securities.Account        CD.Account 
                 0                  0                  0                  0                  0                  0 
            Online         CreditCard 
                 0                  0 
```

Las variables Income y ZIP.Code son de interés ya que en ambas variables la diferencia entre el mínimo y el máximo es considerable, además la diferencia entre la media y mediana de Income no es tan grande pero si provoca mi atención, así que habrá que observar que nos dice el histograma de esta variable. También, cabe mencionar que hay inconsistencias en la variable Experiencie, ya que tiene un **mínimo de -3**.



Al parecer los datos están sesgados hacia la izquierda. Miremos si sucede lo mismo con las demás variables y de paso veamos qué correlación hay entre variables haciendo diagramas de caja y posteriormente una matriz de correlación.





Bueno, ambos gráficos nos revelan mucha información. Por ejemplo, Income y Personal.Loan tienen una correlación positiva significativa, al igual que Education y Personal.Loan, lo que sugiere que los ingresos y un mayor nivel educativo influyen considerablemente en la aceptación de préstamos. Pero hay un problema, Age y Experiencie tienen una correlación casi perfecta (de 0.994, casi 1), por lo que habría que **eliminar** una de las dos columnas para evitar redundancia de cara al modelo predictivo que vamos a hacer con K-NN. Además, los gráficos de caja nos muestran

desbalances en la distribución de los datos. Por ejemplo, Income y Mortgage tienen valores atípicos, CCAbg tiene una distribución sesgada (pues muchos clientes gastan más en tarjetas de crédito) y también Personal.Loan tiene un desequilibrio ya que la mayoría de los clientes no aceptaron el préstamo (pero esto ya lo sabíamos de antemano, así que no es sorpresa).

Como Experiencie tenía un mínimo inconsistente de -3 sería conveniente eliminarla para evitar redundancia, pero como necesito la columna para el inciso A la dejaré. Entonces, voy a normalizar los datos y a proceder con el inciso A.

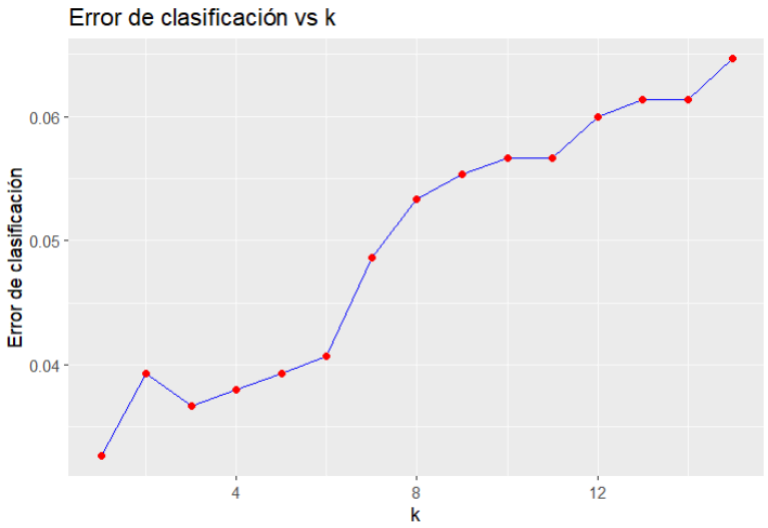
A) Para el proceso del inciso A se eliminaron las columnas solicitadas y se hizo el ajuste de variables ficticias. Se separaron las variables predictoras de la variable objetivo (Personal.Loan), igualmente se separó el cliente nuevo del conjunto de datos de entrenamiento. Este fue el resultado de la predicción con k=1.

```
> pred #0=no aceptó || 1=aceptó
[1] 0
Levels: 0 1
```

B) ¿Cuál es una opción de k que equilibre entre el sobreajuste y la ignorancia de la información del predictor?

La elección de un k adecuado siempre dependerá de los datos que se tengan, pero existe la recomendación de elegir un k entre 5 y 15 pero para no andar probando uno por uno se puede hacer una validación cruzada.

C) Siguiendo la metodología del ejercicio de K-NN de Iris obtuve el siguiente gráfico de los errores de clasificación



O si se prefiere se pueden ver los valores precisos llamando al objeto error.

```
> print(error)
 [1] 0.03266667 0.03933333 0.03666667 0.03800000 0.03933333 0.04066667 0.04866667 0.05333333 0.05533333 0.05666667 0.05666667
[12] 0.06000000 0.06133333 0.06133333 0.06466667
```

Los resultados fueron ciertamente insatisfactorios ya que podría pensarse que como k=1 fue el k con un error más bajo entonces es el mejor k, pero considerando el incremento de errores entre un k y otro creo que la mejor opción es k=3 o k=5, ya que k=1 puede ser muy sensible al ruido, y la diferencia de error entre 3 y 4, y 5 y 6 parece estable.

D) Luego considerar el cliente dado probé con k=3 y k=5, en ambos casos el resultado fue el mismo.

```
> pred
[1] 0
Levels: 0 1
>
```

E) Para dividir los datos en 50, 30 y 20 por ciento primero dividí los datos en 50 y 50 (y luego en 30 y 20 ya que 30+20=50). Estos fueron los resultados para k=3

```
> conf_train
        knn_train
y_train    0    1
      0 2264   12
      1   96  130
> conf_val
     knn_val
y_val    0    1
    0 1333   19
    1   97   53
> conf_test
       knn_test
y_test   0   1
     0 879  16
     1  75  30
```

Al igual que con k=5 (del que se hablará más adelante), el modelo presenta una tendencia a manejar bien los datos conocidos (de entrenamiento), pero no es tan bueno con datos nuevos. Esto se ve reflejado en que según bajamos en las matrices de confusión los resultados empeoran levemente con respecto a la matriz anterior. Esto podría significar un sobreajuste o falta de generalización, por lo que sería bueno explorar alternativas para mejorar el modelo.

Sobre k=5, ¿Qué podemos decir?

```
> conf_train
       knn_train
y_train    0    1
      0 2260   16
      1  122  104
> conf_val
     knn_val
y_val    0    1
    0 1335   17
    1  104   46
> conf_test
      knn_test
y_test   0   1
     0 885  10
     1  78  27
```

Los resultados parecen mejorar en los verdaderos positivos y falsos, pero también un poco en los errores con respecto a k=3, lo que podría significar un sobreajuste. También podemos mencionar que el conjunto de pruebas es un poco peor respecto al de validación, lo que podría indicar que el modelo no generaliza bien los datos nuevos. Y si hablamos del conjunto de pruebas sigue empeorando el asunto, por lo que podemos pensar de nuevo que el modelo no generaliza bien.

Pese a que me provoca cierta resistencia mental usar un k=3, creo que es lo más indicado. Normalmente me gusta empezar con k=7, pero dados los errores de clasificación y la comparación con las matrices de confusión me veo resignado a usar un k=3, aunque bien se podrían explorar alternativas para mejorar el modelo, como lo que comenté al inicio de excluir la columna Experiencie, ya que eso podría provocar que el modelo esté agarrando características inadecuadas para la predicción. También podría probarse el no normalizar los datos ya que al normalizarlos quizá la poca variación entre datos afecte a que no detecte bien los patrones. O simplemente elegir un k más grande. Sea como sea, lo que me sirve de consuelo es que el 3 es un número impar como el 7.