**Trabajo Práctico N°1. Aprendizaje Automatico**

**Maestría en Data Mining y Descubrimiento de conocimiento**

**Autores:** Diego Dell’era - Miguel Angel Barros

**Objetivo**

Analizar los resultados obtenidos mediante la utilización del algoritmo j48 y la variación de sus parámetros

**Materiales y Métodos**

*Data Set*

El data set utilizado se obtuvo del repositorio GESIS, correspondiente al Libniz Institute for the Social Sciences. Específicamente el data set corresponde al relevamiento de los votantes en las elecciones democráticas llevadas a cabo 1994 y su comportamiento en dichos comicios. La dimensión corresponde a 13 atributos por 2360 instancias, de los cuales uno de ellos fue numérico (“age”) y 7 de los atributos nominales fueron de tipo binario.

Software

Para las tareas de discretización se utilizaron programas ejecutados en Python.

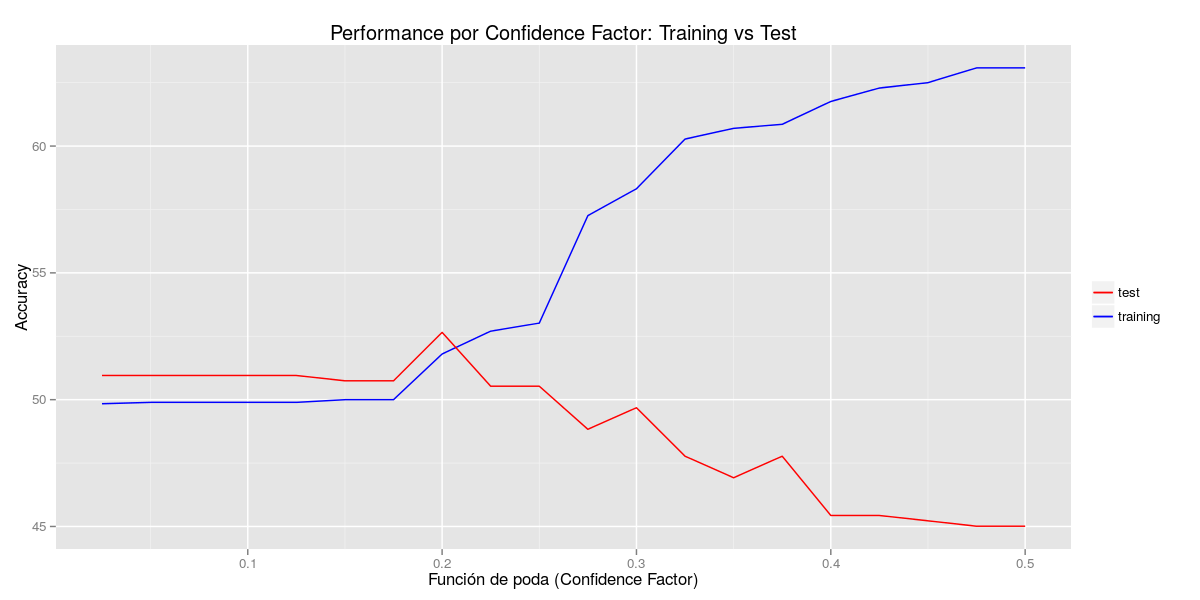
Para la realización de gráficos y sumarizacion de la información se uso R

El algoritmo de clasificación utilizado se corrió bajo el software Weka.

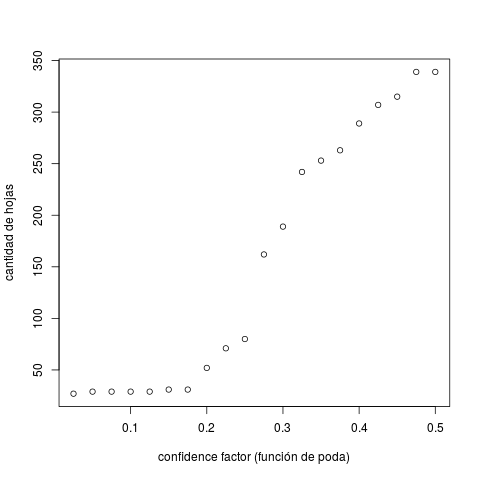
**Resultados**

Sobreajuste y poda

La figura 1 describe la performance del árbol de decisión como clasificador tanto en el set de datos reservados para la validación (20%) como en el conjunto usado para entrenamiento. El clasificador se mantuvo de manera regular hasta alcanzar un pico máximo de 52% de accuracy, y luego a partir de ese punto el algoritmo exhibe el efecto conocido como overfitting. (Figura 1) Como se nota en el gráfico, este efecto está representado por una disminución en la precisión de clasificación a un confidence factor por encima de 0.2 Por otro lado, el proceso de poda y sobre-ajuste en el árbol es una estimación directa del error de decisión, el cual se propaga de una manera notable conforme aumenta, en número de hojas, el tamaño del árbol (Figura 2).



***Figura 1: Precisión del clasificador en los conjuntos de datos de entrenamiento y testing***

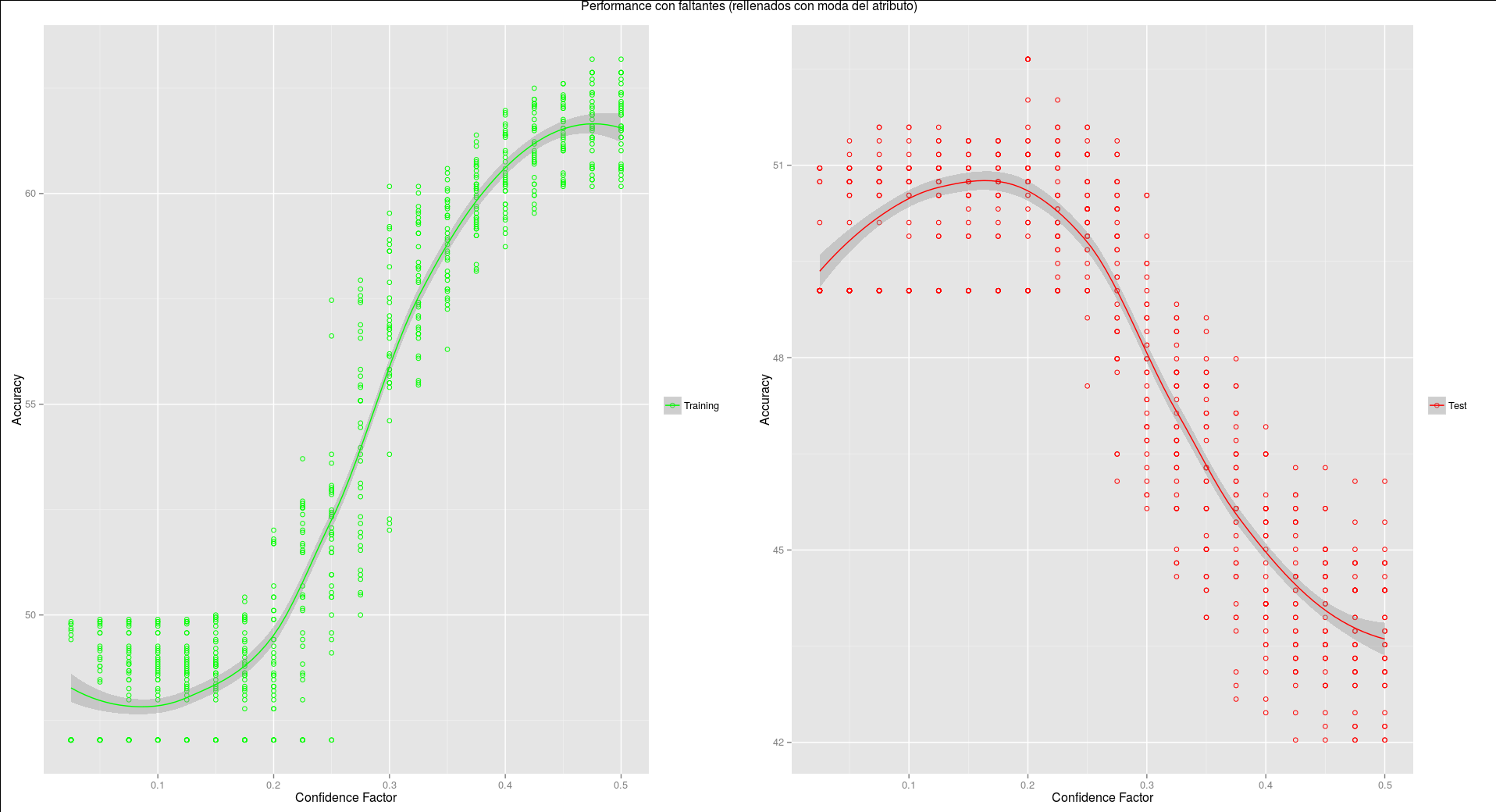


***Figura 2: Tamaño del árbol (cantidad de hojas y nodos) en función de la poda***

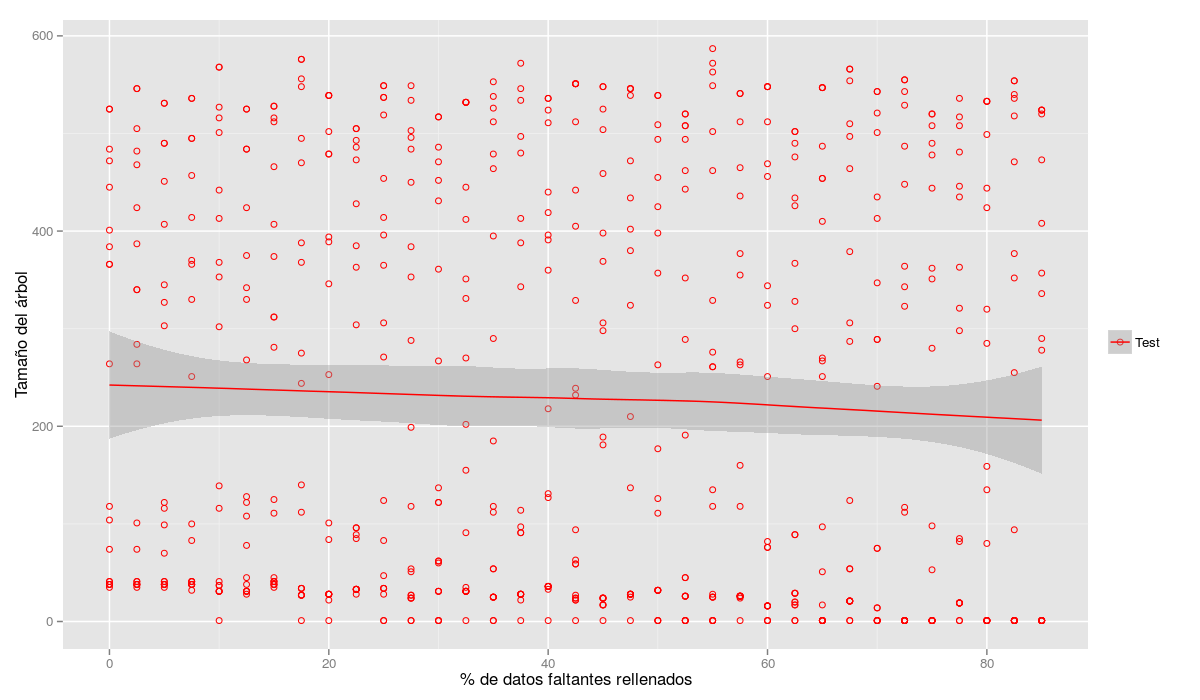
Datos Faltantes

La compleción de los datos faltantes fue realizada mediante las estrategias propuestas en la consigna, según la moda y moda-clase. Para lo cual en primer termino se procedió a obtener la “information gain” de cada atributo. La information gain es una propiedad estadística que básicamente mide cómo clasifica un cierto atributo a los ejemplos. Así, en nuestro conjunto de datos los atributos con mayor information gain fueron, ”member\_pol\_Party” y “age”. En la etapa siguiente, continuando con lo establecido en la consigna del trabajo, se procedió a establecer dos familias de data sets en función del criterio de compleción y rellenado de datos faltantes. En una familia los missing values fueron completados por la moda, en otra familia con la moda del atributo de la clase, y en otra según el método supervisado provisto por Weka. Los datos faltantes fueron colocados de modo creciente, de 0 a 85% en intervalos que fueron de 0 a 34. Las corridas del algoritmo fueron ejecutadas en el 80% del conjunto de datos, reservando un 20% para testing, (en ambas familias), incrementando el confidence factor en forma consecutiva en intervalos de 2.5.

Al igual que en el punto anterior, con el fin de analizar la performance del clasificador J48 mediante el análisis de la accuracy del modelo inducido en el set de test, y en base a los resultados obtenidos en este punto, podemos decir que una vez mas la robustez del algoritmo ha sido demostrada. En la figura 4, se observan ambos grupos de datos, de entrenamiento y de test en donde la curva generada denota poco cambio.



***Figura 3: Representación de del porcentaje de accuracy vs confidence factor en el set de training (izquierda) y set de test (derecha). La inclinación de las curvas es similar.***



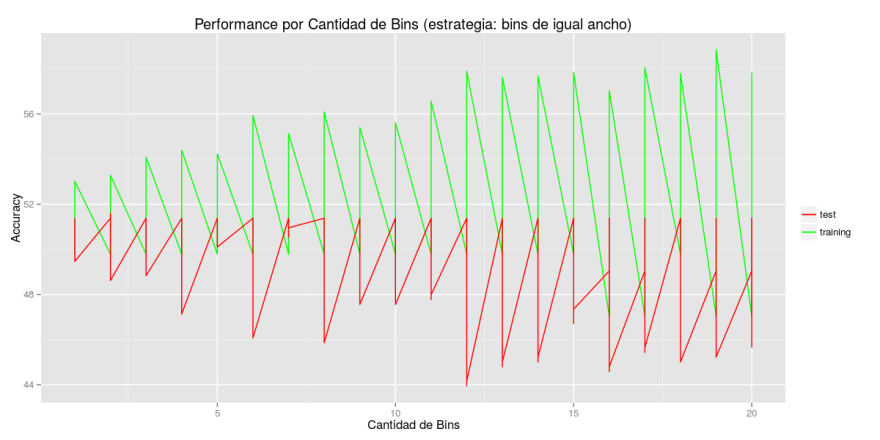
***Figura 4: Tamaño en función de los datos faltantes.***

Discretización

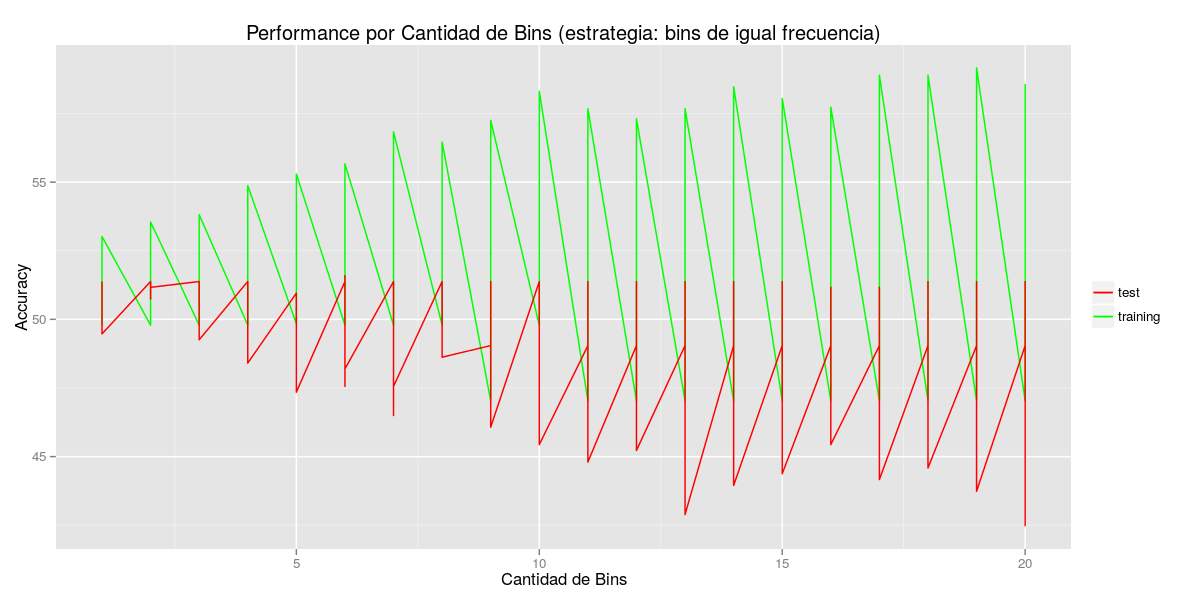
Algunos algoritmos de data mining, y en particular ciertos algoritmos de clasificación requieren que los datos sean de tipo categóricos. Por lo tanto es necesario transformar, en el caso de que existan, los atributos continuos en categóricos. A esta modificación se la denomina discretización. Esta transformación de un atributo continuo a uno categórico incluye dos sub-tareas. En primer lugar, decidir cuantas categorías se van a implementar. En segundo lugar, determinar cómo asignar los valores continuos dentro de cada una de las categorías generadas previamente. Para lo cual es importante establecer, en los valores que se van a discretizar, los ***n*** intervalos especificando los ***n-1*** puntos de Split así como también que todos los valores que caen dentro de los intervalos generados en la etapa anterior sean asignados al mismo valor categórico. Estos intervalos se denominan bins o buckets.

En el presente trabajo se usaron dos metodologías para la construcción de los bins. Esto es estrategias supervisadas y no supervisadas. Inicialmente, realizamos la discratización de atributos siguiendo la estrategia no supervisada. Esto quiere decir que al momento de transformar los atributos no se tuvo en cuenta la información de la clase. De este modo procedimos a dividir a los atributos, para generar los bins en función de la frecuencia (igual frecuencia de atributo en cada bin) y en función en función de la densidad (bins de igual ancho). Para la estrategia supervisada. Al igual que los puntos anteriores, se reservo el 20% del conjunto de datos para realizar el test.

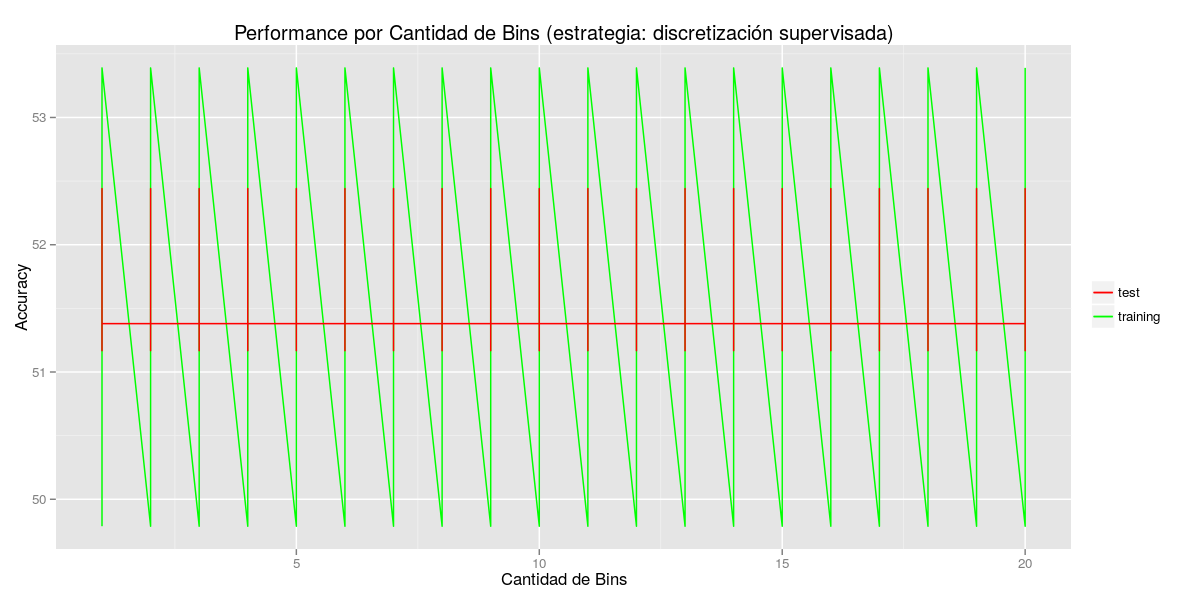
Las figuras siguientes, ***5a, 5b y5c***  muestran el rendimiento del clasificador en función de la estrategia de discretización. En dichos gráficos están representados los comportamientos de ambos conjuntos de datos , training (80%) y test (20%). Al analizar cada uno de lo gráficos separadamente y focalizando la accuracy del clasificador sobre el conjunto de datos de validación se demuestra que en la estrategia supervisada tiene un mejor rendimiento sobre los mismos. Dicho de otro modo, se observa una menor amplitud conforme aumenta la cantidad de sub-rangos, lo cual sugiere que es más precisa **(5c).**



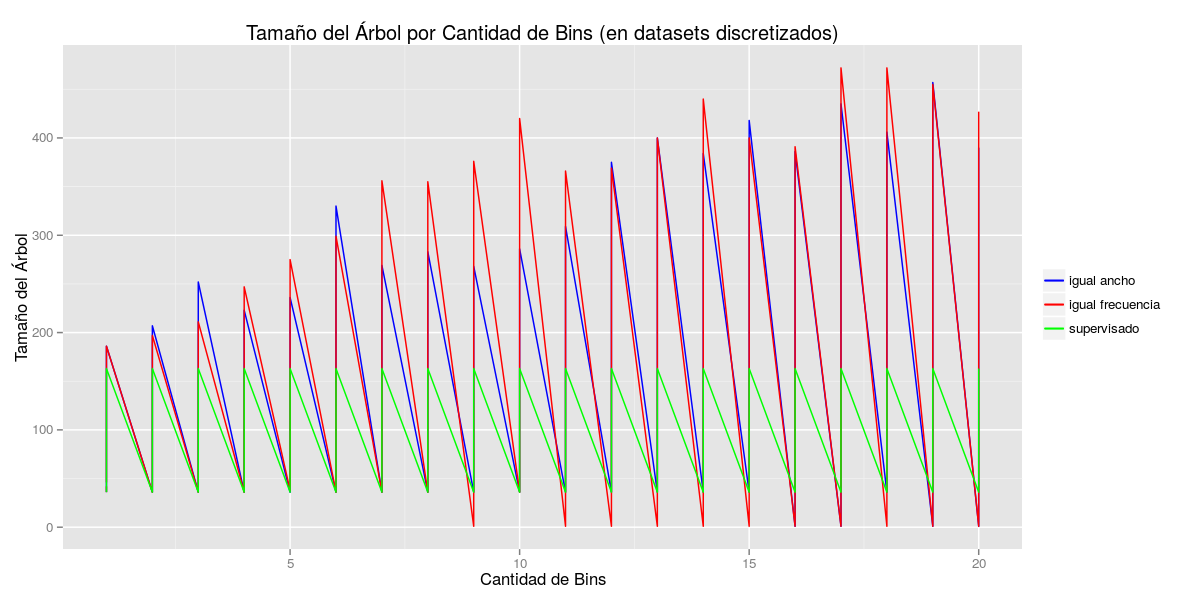
***Figura 5a: Discretización No supervisada: bins divididos en igual ancho (densidad) en función de la precisión del clasificador*.**



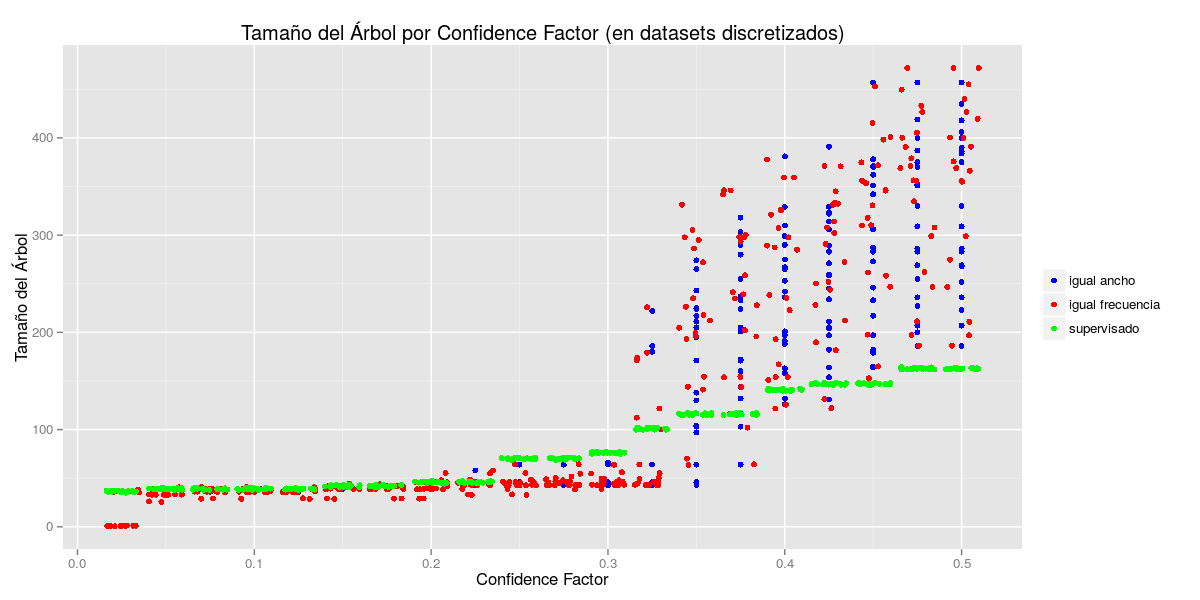
***Figura 5b: Discratización No supervisada: bins divididos en frecuencia vs la precisión del clasificador.***



***Figura 5c: Discratización Supervisada: bins divididos en igual frecuencia vs la precisión del clasificador.***



***Figura 6: Descripción del tamaño del árbol y la cantidad de bins generados en la discretización. Los diferentes colores representan las estrategias de discretización.***



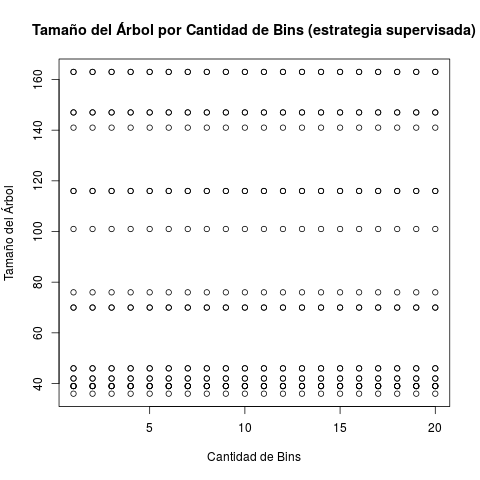
***Figura 7: comparación de estrategias de discretización: tamaño de los árboles generados en función del confidence factor***

Ahora bien, al observar la figura **6** donde se analiza la implementación las tres estrategias de discretización en función del tamaño del árbol, y para los dos conjuntos de datos, hay varios puntos importantes a tener en cuenta.

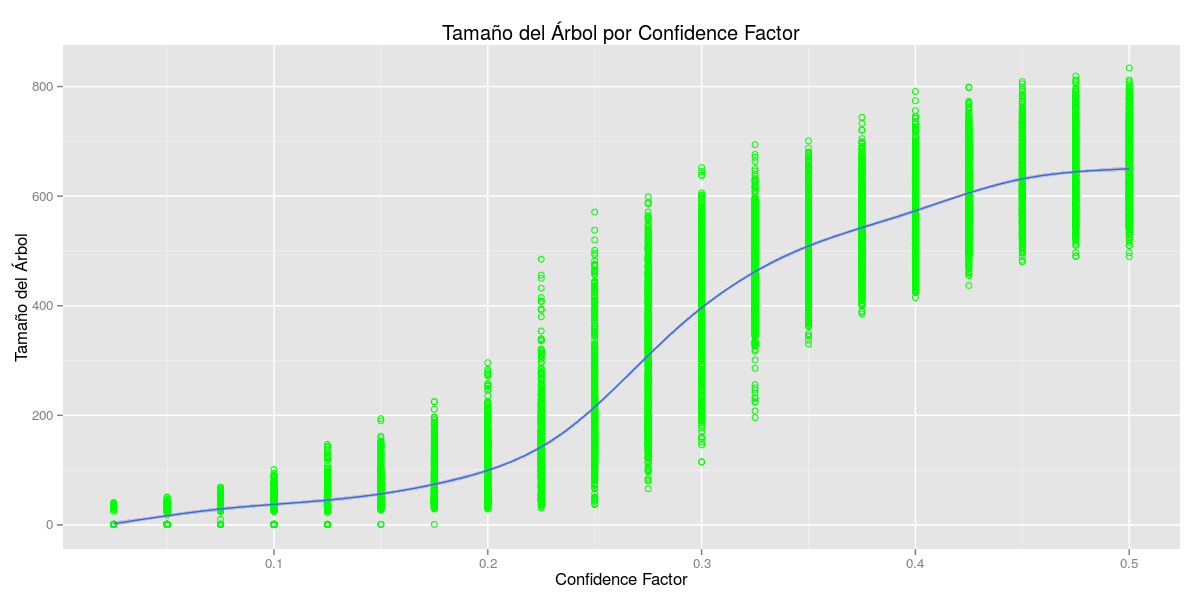
En primer lugar, es importante notar la diferencia entre la dispersión que existe entre los métodos no supervisados de discretización y supervisados. A partir del valor de 0.3 para el confidence factor esta diferencia es notoria. Si se hace foco en las estrategias no supervisadas usadas para discretizar notamos una dispersión aún mayor para el método de frecuencia.

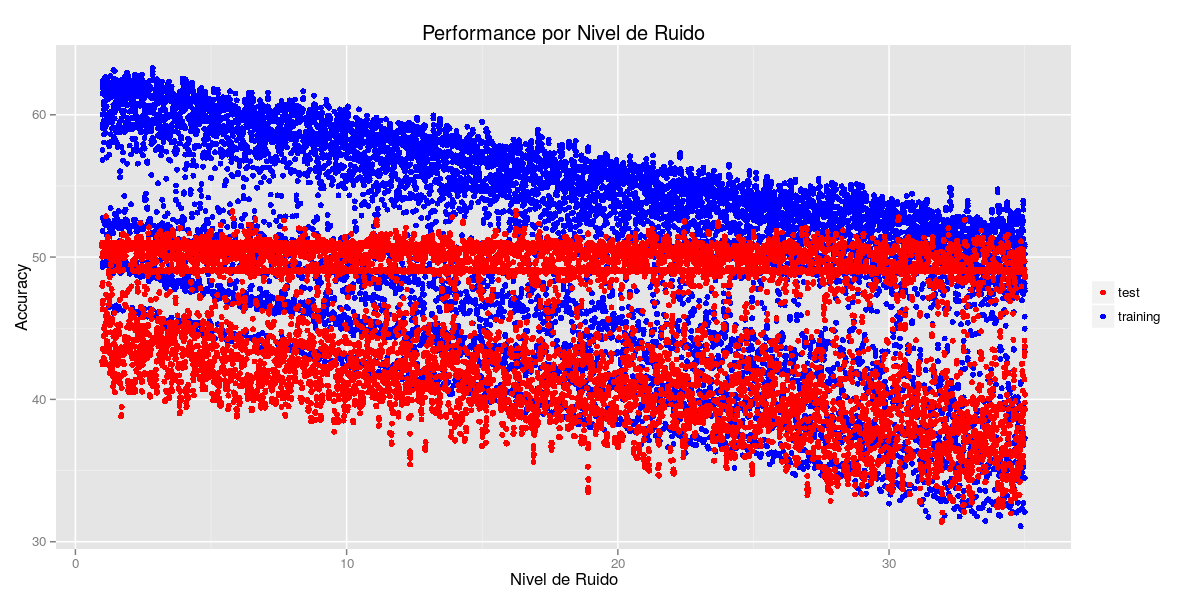
Otro punto importante a resaltar es la uniformidad del método supervisado conforme aumenta el tamaño del árbol ***figuras 7 y 8.*** Esto se debe a que la entropía de un intervalo es una medida que refleja la pureza de ese intervalo. Si un intervalo contiene solo valores de una clase, como el caso de la estrategia acá utilizada, se dice que es perfectamente puro. O sea que su entropía es cero y esto contribuye poco y nada al nivel de entropía general.

En base a los resultados obtenidos en este punto se puede concluir que las técnicas de discretización supervisadas otorgan mejores resultados. Lo cual es de esperar debido a que cuando se construye un intervalo sin tener el conocimiento de las clases, dicho intervalo posee una mezcla de las mismas. Tal situación se da en las técnicas de discretización no supervisadas.



***Figura 8: tamaño del árbol y cantidad de bins generados mediante la estrategia supervisada***

***Figura X: tamaño del árbol para distintos niveles de ruido***

***Figura X: performance para distintos niveles de ruido***