

TEST CON PREGUNTAS DE RESPUESTA BREVE
DURACIÓN: 1 HORA

1. (0.5 puntos) Poner un ejemplo de un individuo de Programación Genética con *bloat*

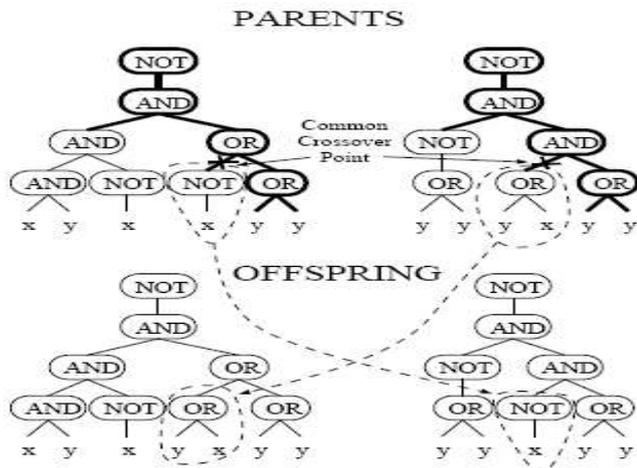
Existe *bloat* cuando aparecen individuos grandes con trozos de código que no tiene efecto (intrones). Por ejemplo, en el individuo siguiente, el trozo e código (* 0 (+ 4 8 7)) no tiene ningún efecto puesto que está multiplicado por cero.

(+ (* 0 (+ 4 8 7)) 1)

2. (0.5 puntos) ¿Porqué la búsqueda en Programación Genética se estanca cuando se produce *bloat*?

Cuando hay *bloat*, la mayor parte de los individuos está constituida por intrones (código sin efecto). Si una mutación o cruce se produce dentro de un intrón, no se van a producir cambios reales, por lo que los hijos harán exactamente lo mismo que los padres. Es decir, pasa a ser muy improbable que ocurra alguna mutación o cruce beneficiosos.

3. (0.5 puntos) Poner un ejemplo de dos individuos padres y dos hijos tras el 1-point crossover (cruce de un punto).

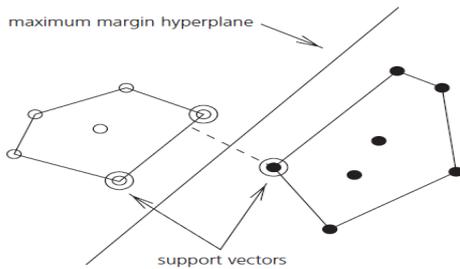


4. (0.5 puntos) Supongamos que tenemos un problema de clasificación con 3 clases A, B y C. En los datos de entrenamiento hay 50 datos de la clase A, 150 de la clase B y 50 de la clase C. Supongamos que el Naive Bayes obtiene un 60% de aciertos en validación cruzada. ¿Es un buen resultado? ¿Porqué?.

Para poder considerar que es un buen resultado tiene que ocurrir que sea mejor que el azar (1/3 en el caso de tres clases) y mejor que el porcentaje de la clase mayoritaria ($150/(150+50+50) = 60\%$ en este caso). $60\% = 60\%$, luego no es un buen resultado, porque NB obtiene lo mismo que el algoritmo que siempre devuelve la clase mayoritaria (ZeroR).

5. (1 punto) ¿Qué significado tienen (qué son) los vectores de soporte de las máquinas de vectores de soporte?

A groso modo, los vectores de soporte son los que se utilizan para construir la frontera entre las clases. Son justamente los datos que están en el borde entre las dos clases.



6. (1 punto) Explicar en uno o dos párrafos en qué consiste el algoritmo de Boosting

Se trata de una metatécnica, porque utiliza otro clasificador base (por ejemplo, J48). Construye varios modelos de manera secuencial, cada uno se centra en los datos que el anterior clasificaba mal. Al final se obtiene un conjunto (ensemble) de clasificadores. La decisión final del ensemble para un dato determinado se hace por votación ponderada: aquellos clasificadores que funcionaban bien mejor con los datos de entrenamiento tendrán más peso en la decisión final que aquellos que lo hacían peor.

7. (1 punto) Explicar por qué el “brood recombination” reduce el bloat, sin incrementar demasiado el coste computacional.

Una de las razones por las que ocurre el bloat es la defensa contra el cruce (*Accuracy theory*). El cruce es una operación destructiva (en la mayor parte de los casos, reduce la fitness de los hijos), por lo que una manera que tienen los individuos de protegerse es crear intrones (bloat), de tal manera que el cruce no tenga efectos destructivos. Si se consigue disminuir el efecto destructivo del cruce, se conseguirá disminuir el bloat. El brood recombination genera múltiples hijos, seleccionando distintos puntos de cruce. Sobrevive el mejor hijo de acuerdo a la fitness. Al generar varios hijos y quedarse con el mejor, se reduce el efecto destructivo del cruce, y por tanto disminuye la necesidad de los individuos para generar intrones (o sea, se reduce el bloat).

8. (1 punto) Describir brevemente como funciona el procedimiento de validación cruzada y por qué es preferible usarlo a dividir el conjunto de datos en una parte para entrenamiento y otra para test. Si utilizamos validación cruzada, ¿los clasificadores obtenidos serán mejores (mayor porcentaje de aciertos) que si utilizáramos el método de entrenamiento/test? ¿Por qué?

La validación cruzada consiste en dividir el conjunto de datos en k partes y repetir un ciclo en el que se entrena con $k-1$ partes y se hace el test con la parte restante. Al final, se calcula la media de los k tests. Es mejor que utilizar la técnica de separar un conjunto para entrenamiento y otro para test, porque en esta última técnica es más probable que aparezcan sesgos. En la validación cruzada, los sesgos que aparezcan en algunas iteraciones se cancelarán con los que aparezcan en otras iteraciones. Otra ventaja de la validación cruzada es que todos los datos figuran como entrenamiento o test en alguna de las iteraciones de validación cruzada, por lo que los datos están mejor utilizados que en la división en entrenamiento / test.

Los clasificadores obtenidos pueden ser mejores, porque una vez realizados los k ciclos de validación cruzada, se construye el clasificador final con todos los datos (en el caso de conjunto de entrenamiento/conjunto de test, sólo se utilizarían los de entrenamiento). Cuantos más datos se utilizan para construir el clasificador, más probable es que se obtengan mejores resultados.

9. (1 punto) ¿Qué método de selección de atributos utilizarías con los siguientes datos, de evaluación de atributos o de evaluación de subconjuntos de atributos, para detectar los atributos relevantes?. Justificarlo. Nota: los datos tienen cuatro atributos (X Y Z y W) y la Clase.

| <u>X Y Z W Clase</u> |
|----------------------|----------------------|----------------------|----------------------|
| 0 0 0 0 + | 0 1 0 0 - | 1 0 0 0 - | 1 1 0 0 + |
| 0 0 0 1 + | 0 1 0 1 - | 1 0 0 1 - | 1 1 0 1 + |
| 0 0 1 0 + | 0 1 1 0 - | 1 0 1 0 - | 1 1 1 0 + |
| 0 0 1 1 + | 0 1 1 1 - | 1 0 1 1 - | 1 1 1 1 + |

Se puede ver que ninguno de los cuatro atributos está correlacionado con la clase. Saber que cualquiera de ellos vale 0 o 1, no da ninguna información acerca de cual podría ser la clase. Sin embargo, la combinación de dos atributos (X e Y) si que da esa información. Si XY=00 o XY=10, la clase es +, en caso contrario es -. Para seleccionar los **subconjuntos** de atributos relevantes, es necesario un método basado en conjunto de clasificadores, que **evalúe subconjuntos**, y no atributos de manera independiente. Por tanto, el método a utilizar sería de evaluación de subconjuntos.

10. (1 punto) Supongamos que tenemos un problema de clasificación con 1000 atributos y dos clases. Tenemos 1000 datos para entrenar y 1000 datos para hacer el test. Si construimos con esos datos un clasificador lineal con los datos de entrenamiento, ¿cuál sería aproximadamente su porcentaje de aciertos en test?. ¿porqué?.

Un separador lineal con X atributos es capaz de separar X+1 datos cualesquiera, incluso datos generados aleatoriamente. Por ejemplo, un clasificador lineal con 2 atributos es capaz de separar 3 datos cualesquiera, independientemente de la clase a la que pertenezcan. Por tanto, el clasificador lineal construido con esos datos de entrenamiento, en test no lo hará mejor que el azar: 50% de aciertos, aproximadamente.

11. (1 punto) Supongamos que estamos experimentando con dos conjuntos de datos D_1 y D_2 y dos algoritmos de clasificación A y B. Queremos comparar los algoritmos A y B haciendo 10 validaciones cruzadas de 10 folds cada una y calculamos la media y la desviación típica de los porcentajes de aciertos. Supongamos que los resultados son estos (dado (a,b) , el valor a corresponde a la media y el valor b a la desviación típica):

- Datos D_1 : A = (90%, 8%), B=(94%, 7%)
- Datos D_2 : A = (90%, 0.001%), B=(91%, 0.002%)

La pregunta es, ¿en cual de los dos conjuntos de datos D_1 y D_2 es mas probable que la diferencia entre los algoritmos A y B sea estadísticamente significativa?. ¿Porqué?.

Es mas probable que la diferencia sea estadísticamente significativa en los datos D_2 . Aunque en D_2 la diferencia entre las medias es pequeña (91%-90%=1%), la dispersión es casi insignificante (alrededor del 0.001%), por lo que la diferencia entre A y B prácticamente seguro que no es debida al azar. El caso extremo es el que la desviación típica es 0. En ese caso, incluso una diferencia entre medias de 90.0001% y 90.0002% sería significativa, porque no hay ninguna variabilidad alrededor de la media (desviación típica cero).

En el caso de los datos D_1 , la diferencia entre medias es grande, pero la dispersión alrededor de la media también es muy grande (7%), por lo que es fácil que la diferencia sea debida al azar.

12. (1 punto) ¿Porqué generaliza mal el Naive Bayes si existen varios atributos redundantes?.

Porque el Naive Bayes calcula una probabilidad multiplicando varios términos, uno por cada atributo. Si dos atributos son redundantes, uno de los atributos tendrá el equivalente a dos términos, mientras que los demás tendrán sólo uno. Así, uno de los atributos tendrá más peso que los demás, no porque esto sea bueno o malo para la clasificación, sino por el hecho de que existe esa redundancia. Si el atributo redundante es además irrelevante, es prácticamente seguro que Naive Bayes generalizará mal.