Práctica 2: Selección de variables

Alumno: Pablo Alonso

2)

Lo primero que se observa es que los datos "A" y los datos "B" son generados de forma uniforme por lo que los valores de cada variable siguen una distribución uniforme y dependen sola de esta distribución, por lo tanto las variables son independientes entre si. Además para medir el error en la búsqueda greedy y la importancia en RFE se usan los modelos de SVM y RF.

Los datos "A" tienen las siguientes características:

- 50% de los datos tienen el signo de la 8va variable.
- 20% de los datos tienen el signo de la 6ta variable.
- 10% de los datos tienen el signo de la 4ta variable.
- 5% de los datos tienen el signo de la 2da variable.

En la siguiente tabla se muestran los resultados para los datos "A" , los rankings empiezan por la variable más importante y terminan por la menos importante:

Datos A		
Algoritmo	Ranking con SVM	Ranking con RF
Backward Greedy	8 9 7 3 4 10 5 2 1 6	8 2 5 9 6 7 4 1 3 10
RFE	8 6 4 10 2 1 3 5 7 9	8 6 4 10 5 1 2 3 9 7
Forward Greedy	8 2 7 1 5 9 3 10 4 6	8 1 2 9 3 5 7 10 4 6
Test No Paramétrico	8 6 4 2 7 1 9 10 5 3	

Los mejores resultados se ven con el test no paramétrico, pues reflejan mejor la correlación de las variables 8,6,4,2 con la clase. Esto es esperable por que este algoritmo no mezcla el ruido con la información relevante al hacer el test de forma individual para cada variables.

RFE captura la relevancia de las variables 8,6,4 pero no la de la variable 2. Ya que esta variable no aporta tanta información al problema se vuelva más descartable al haber ruido en los datos.

Las métodos greedy son los que dan peores resultados porque consideran variables conjuntas. La relevancia de las variables se calcula de acuerdo a la performance del modelo pero el ruido puede afectar esta performance haciendo que el algoritmo dé pasos equivocados incorporando variables que no son relevantes. Esto se observa claramente en el Forward, pues el primer paso es correcto

al considerar las variables de forma independiente y luego empieza a incorporar variables no tan relevantes al considerar en conjunto variables con ruido.

Los datos "B" tienen las siguientes características:

- La clase depende de las primeras 2 variables (xor entre ellas)
- Existe una correlación de las variables 3 y 4 con la clase en el 50% de los datos.

Datos B		
Algoritmo	Ranking con SVM	Ranking con RF
Backward Greedy	2 1 8 6 5 7 3 4	2 1 8 6 5 7 3 4
RFE	1 2 4 3 5 7 8 6	2 1 4 3 5 7 6 8
Forward Greedy	4 3 5 7 8 6 2 1	4 3 5 7 1 2 6 8
Test No Paramétrico	3 4 1 8 6 2 7 5	

En este Backward solo preserva aquellas variables que por separado no aportan nada pero que en conjunto aportan toda la información de la clase, pero relega las variables 3 y 4 las aunque estas tengan cierta correlación con la clase porque al descartar las variables que son ruido el modelo no mejora demasiado pues este las ignora.

Forward Greedy incorpora primero aquellas variables que aportan información de forma individual (variables 3 y 4) y cuando agrega la variable 1 o 2, en el siguiente paso incorporará la que le falta porque juntas son correlativas con la clase.

Como se espera, el test no paramétrico no funciona para este caso porque no captura la correlación de las variables en conjunto sino que lo hace de forma individual por lo que las variables 3 y 4 se ven favorecidas.

Para este problema el mejor test resulta ser RFE porque captura la ganancia de información de las variables y pone aquellas que son ruido al final del ranking.

3)

Los datos tienen las siguientes características:

- 100 datos del problema diagonal con 10 variables y $\sigma = 2$
- Se agrego 90 variables de ruido uniforme

La siguiente tabla muestra los resultados de haber aplicado los 4 métodos de selección:

Algoritmo	Frecuencia de aciertos
Backward Greedy	8 11 20 9 2 13 30 30 19 20
Forward Greedy	30 30 30 30 30 30 30 30 30 30
RFE	30 30 30 30 30 30 30 30 30 30
Test No Paramétrico	30 30 30 30 30 30 30 30 30 30

Se observa que los peores resultados son los que se obtienen con Backward Greedy. Hay que tener en cuenta que este algoritmo comienza considerando todas las variables y que las va descartando de acuerdo a la performance de los modelos que obtenga al descartar una por una. Como vimos en teoría, las variables que son ruido no afectan demasiado al ser descartadas sino que los grandes cambios de performance se dan al eliminar las variables gaussianas, con lo cual el algoritmo podría llegar a descartarla antes que a una variable que solo representa ruido. Además las variables con ruido tienen valores uniformes entre -1 y 1 (los centros de las distribuciones) por lo que también pueden confundir al modelo. Existen 2 variables (7 y 8) que claramente trabajan muy bien en conjunto y que logran mantenerse siempre en el ranking. Al repetir el experimento con otros datos, se confirma que siempre permanecen entre las primeras 10 posiciones. Esto nos indica que la dirección de sus ejes en conjunto se asemeja a la dirección del eje que separa las 2 distribuciones, es decir en la cual se encuentra la máxima varianza de los datos.

Forward greedy no incorpora las variables ruidosas en el top ten ya que el algoritmo no las agrega hasta que se ve obligado a hacerlo y esto sucede solo cuando todas las variables originales (aquellas que realmente aportan información) fueron seleccionadas.

RFE también contruye modelos multivariados y va descartando variables en cada paso, pero a diferencia de Backward Greedy, en vez de medir la performance del modelo mide la ganancia de información de cada variable lo que lo hace mucho más efectivo pues las variables gaussianas son las que realmente representan la distribución de los datos y su ganancia de información siempre va a ser mayor.

Incluso con oneway test el cual no asume ninguna distribución se logran buenos resultados pues al medir de forma independiente la varianza de las variables no es afectado por el ruido.

4)

El dataset se descargó de https://archive.ics.uci.edu/ml/datasets/DBWorld+e-mails.

Se tienen 64 emails con 4702 atributos, los cuales son palabras seleccionadas de los emails. Como vimos en teoría, al tener una dimensionalidad tan alta comparado con la cantidad de mails, el volumen de los datos disminuye.

El experimento se trata de seleccionar las 2000 palabras más influyentes para reducir la dimensionalidad y que el volumen de los datos sea mayor. Luego se compara la performance de los modelos entrenados con ambos dataset. Para esto se toman 14 ejemplos de los 64 como test.

Se hacen 5 iteraciones.Los resultados son los siguientes:

Dimensionalidad	Resultados	Media
4207	0.28 0.14 0 0.14 0	0.11
2000	0.28 0.14 0 0 0	0.08

En general se observa una mejora en la performance del modelo, además los tiempos de entrenamiento van a ser menores y se reduce el riesgo de sobreajuste.