Cuestionario 3 - AA

Óscar David López Arcos.

---

1. Tanto "bagging" como validación-cruzada cuando se aplican sobre una muestra de datos nos permiten dar una estimación del error de un modelo ajustado a partir de dicha muestra. Enuncie las diferencias y semejanzas entre ambas técnicas. Diga cual de ellas considera que nos proporcionará una mejor estimación del error en cada caso concreto y por qué.

- **Semejanzas:** Ambas técnicas reducen el sobreajuste del modelo, utilizando para ello distintas muestras de entrenamiento. De estas muestras obtendrán diferentes modelos de predicción que promediarán para calcular uno solo.

- **Diferencias:** Bagging obtiene las muestras de entrenamiento remuestreando de forma aleatoria con reemplazamiento. Por otro lado, validación-cruzada utiliza distintos métodos para escoger la muestra de entrenamiento y la de validación. Por tanto, en bagging no tenemos ningún control sobre los subconjuntos mientras que en validación-cruzada podemos exigir restricciones tales como que estos sean disjuntos.

2. Considere que dispone de un conjunto de datos linealmente separable. Recuerde que una vez establecido un orden sobre los datos, el algoritmo perceptron encuentra un hiperplano separador iterando sobre los datos y adaptando los pesos de acuerdo al algoritmo.

Algorithm 1 Perceptron

1: **Entradas**: $(\mathbf{x}_i, y_i), i = 1, \ldots, n$, $\mathbf{w} = 0$, $k = 0$
2: **repeat**
3: $k \leftarrow (k + 1) \bmod n$
4: **if** $\text{sign}(y_i) \neq \text{sign}(\mathbf{w}^T \mathbf{x}_i)$ **then**
5: $\mathbf{w} \leftarrow \mathbf{w} + y_i \mathbf{x}_i$
6: **end if**
7: **until** todos los puntos bien clasificados

Modificar este pseudo-código para adaptarlo a un algoritmo simple de SVM, considerando que en cada iteración adaptamos los pesos de acuerdo al caso peor clasificado de toda la muestra. Justificar adecuadamente/matemáticamente el resultado, mostrando que al final del entrenamiento solo estaremos adaptando los vectores soporte.

```
repeat
     PeorClasificado = y_i x_i  /  (y_i(w x_i − b)) = min{ (y_j(w x_j − b)) },  1 ≤ j ≤ |D|
     if PeorClasificado < 1
          w = w + PeorClasificado
until todos bien clasificados
```

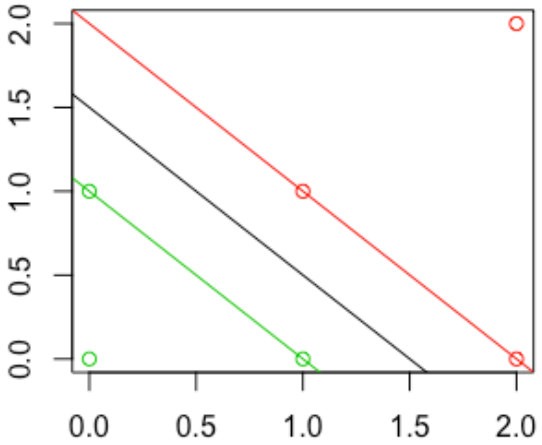
En cada iteración del algoritmo, el punto peor clasificado será aquel que se encuentre más lejos de ser clasificado correctamente. En la última iteración, como todos han sido clasificados, aquel con valor mínimo de la expresión $(y_i(wx_i - b))$ coincidirá con el vector soporte (el más lejano al hiperplano óptimo).

3. Considerar un modelo SVM y los siguientes datos de entrenamiento:

Clase-1: $\{(1,1),(2,2),(2,0)\}$, ***Clase-2:*** $\{(0,0),(1,0),(0,1)\}$

a) Dibujar los puntos y construir por inspección el vector de pesos para el hiperplano óptimo y el margen óptimo.

El corte con el hiperplano óptimo sería la recta $f(x) = -x + \frac{3}{2}$



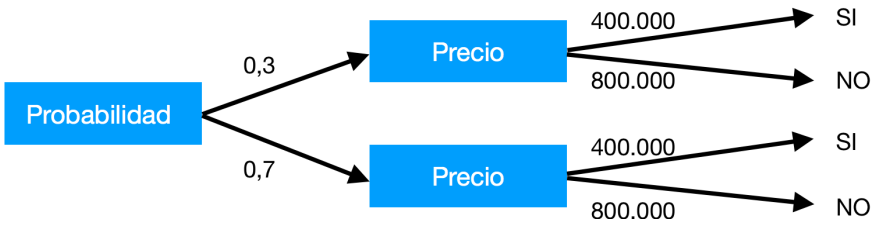
b) ¿Cuáles son los vectores soporte?

Los vectores formados por los puntos (0,1) y (1,0) y los puntos (0,2) y (2,0).

c) Construir la solución en el espacio dual. Comparar la solución con la del apartado (a)

4. Una empresas esta valorando cambiar su sistema de proceso de datos, para ello dispone de dos opciones, la primera es adquirir un nuevo sistema compuesto por dos sistemas idénticos al actual a 200.000 euros cada uno, y la segunda consiste en adquirir un nuevo sistema mucho mayor por 800.000 euros. Las ventas que la empresa estima que tendrá a lo largo de la vida útil de cualquiera de sus nuevos equipos es de 5.000.000 de euros en el caso de un mercado alcista, a lo que la empresa le asigna una probabilidad de que suceda del 30 %, en caso contrario, las ventas esperadas son de 3.500.000 euros. Construir el árbol de decisiones y decir que opción es la más ventajosa para la empresa.

Equipo	Prob. Alcista	Precio	Beneficio	Decisión
1	0,3	800.000	4.200.000	No
2	0,3	400.000	4.600.000	Si
1	0,7	400.000	2.700.000	No
2	0,7	800.000	3.100.000	Si



Finalmente puede observarse que, se cumpla o no la probabilidad, es más rentable adquirir dos sistemas de 200.000€ en ambos casos.

5. ¿Que algoritmos de aprendizaje no se afectan por la dimensionalidad del vector de características? Diga cuáles y por qué.

Support Vector Machine utiliza kernels para transformar los datos. Estas transformaciones calculan las distancias entre cada par de puntos y utilizan esta información como nuevos datos. Es por eso que finalmente el algoritmo no se ve afectado por el total de características de la muestra (dimensión), si no por el total de muestras usadas.

Los árboles tampoco se ven afectados por la dimensionalidad del vector de características. Si una única característica es capaz de separar los datos con una precisión total, es indiferente cuál fuese el tamaño de éste.

6. Considere la siguiente aproximación al aprendizaje. Mirando los datos, parece que los datos son linealmente separables, por tanto decidimos usar un simple perceptron y obtenemos un error de entrenamiento cero con los pesos óptimos encontrados. Ahora deseamos obtener algunas conclusiones sobre generalización, por tanto miramos el valor d_{vc} ***de nuestro modelo y vemos que es*** $d + 1$***. Usamos dicho valor de*** d_{vc} ***para obtener una cota del error de test.***

Argumente a favor o en contra de esta forma de proceder identificando los posible fallos si los hubiera y en su caso cual hubiera sido la forma correcta de actuación.

El primer fallo ha sido basarse en la disposición de los datos de entrenamiento para elegir el algoritmo de predicción. Esto provoca que ajustemos un modelo de acuerdo a la muestra, favoreciendo el sobreajuste y, en definitiva, yéndonos tras el training, algo que siempre tratamos de evitar. Por otro lado, la cota obtenida a partir de d_{vc} no es representativa del error fuera de la muestra, ya que es una cota laxa cuyo fin es más teórico que práctico.

La forma correcta de proceder hubiese sido experimentar con diferentes modelos de clasificación, utilizando técnicas de regularización para evitar el sobreajuste. Para calcular un error representativo de test, lo correcto sería utilizar procedimientos como validación en los que probar nuestro modelo con datos no usados para el entrenamiento.

7. Discuta pros y contras de los clasificadores SVM y Random Forest (RF). Considera que SVM por su construcción a través de un problema de optimización debería ser un mejor clasificador que RF. Justificar las respuestas.

Pros:

- RandomForest: Aborda de forma natural problemas multiclase. Funciona correctamente con mezcla de de características numéricas y categóricas, incluso si estas están en escalas distintas. Es más simple de entrenar y ajustar.

- SVM: Aporta minimización del riesgo estructural y evita el sobreajuste. Está definido como un problema de optimización, para los que existen métodos matemáticos muy eficientes. Su dimensión no se ve afectada por el número de características gracias al uso de kernels.

Contras:

-RandomForest: Generalmente necesita número de instancias elevado para producir resultados de calidad, aunque es cierto que funciona muy bien cuando esto se cumple. Sobreajusta en ciertos grupos de datos con tareas de clasificación ruidosas.

- SVM: Está diseñado intrínsecamente para problemas de dos clases y debe reducirse a múltiples problemas de clasificación binaria para trabajar con multiclase. Es necesario un preprocesado de los datos para poder trabajar con ellos. No es tan simple de entrenar y ajustar como RandomForest.

8. ¿Cuál es a su criterio lo que permite a clasificadores como Random Forest basados en un conjunto de clasificadores simples aprender de forma más eficiente? ¿Cuales son las mejoras que introduce frente a los clasificadores simples? ¿Es Random Forest óptimo en algún sentido? Justifique con precisión las contestaciones.

Los clasificadores simples son más fáciles de entrenar y ajustar que los clasificadores complejos, lo que permite que cada uno de ellos aprenda qué características son importantes para producir una buena clasificación.

Además, múltiples clasificadores "en secuencia" permiten que cada uno produzca una división de los datos de forma que el siguiente pueda utilizar el subconjunto generado para mejorar la clasificación.

RandomForest no es óptimo, ya que no utiliza ningún procedimiento que reduzca el riesgo estructural fuera de la muestra de entrenamiento. Solamente producirá resultados satisfactorios si la muestra es suficientemente representativa (gran tamaño) y los datos no presentan demasiado ruido.

9. En un experimento para determinar la distribución del tamaño de los peces en un lago, se decide echar una red para capturar una muestra representativa. Así se hace y se obtiene una muestra suficientemente grande de la que se pueden obtener conclusiones estadísticas sobre los peces del lago. Se obtiene la distribución de peces por tamaño y se entregan las conclusiones. Discuta si las conclusiones obtenidas servirán para el objetivo que se persigue e identifique si hay algo que lo impida.

Aunque el enunciado asegura que la muestra es representativa, no podemos afirmar que no estemos sesgando la población al realizar una única pesca, aunque sea una muestra de tamaño elevado. Para asegurarnos de obtener una distribución representativa del tamaño de los peces, sería más razonable tomar distintas muestras más pequeñas y así poder predecir con una varianza más reducida.

10. Identifique dos razones de peso por las que el ajuste de un modelo de red neuronal a un conjunto de datos puede fallar o equivalentemente obtener resultados muy pobres. Justifique la importancia de las razones expuestas.

Por un lado, el ajuste puede fallar si las neuronas comienzan a formar grupos para tomar decisiones de manera conjunta, no aportando información útil de forma individual. Esto crearía neuronas totalmente dependientes de otras y debe combatirse en el entrenamiento anulando las aportaciones de ciertas neuronas para que las de su entorno no cuenten siempre con ellas.

Por otro lado, el algoritmo de backpropagation según el tipo de funciones no-lineales que se utilicen, puede ocasionar que la red se sature. Por ejemplo, al derivar para encontrar el mínimo, la sigmoide devolverá resultados muy cercanos a cero para los valores donde es prácticamente continua. Esto provocaría una gran perdida de información. En los últimos años el uso de nuevas funciones no-lineales pero no acotadas ha permitido solventar este aspecto.