Cuestionario 3. Aprendizaje Automático

José Carlos Martínez Velázquez

9 de Junio de 2016

1. Considere los conjuntos de hipótesis H_1 y H_{100} que contienen funciones Booleanas sobre 10 variables Booleanas, es decir $X = \{-1, +1\}^{10}$. H_1 contiene todas las funciones Booleanas que toman valor +1 en un único punto de X y -1 en el resto. H_{100} contiene todas las funciones Booleanas que toman valor +1 en exactamente 100 puntos de X y -1 en el resto.

a) ¿Cuantas hipótesis contienen H_1 y H_{100}?

b) ¿Cuantos bits son necesarios para especificar una de las hipótesis en H_1?

c) ¿Cuantos bits son necesarios para especificar una de las hipótesis en H_{100}?

Argumente sobre la relación entre la complejidad de una clase de funciones y la complejidad de sus componentes.

Apartado a)

Parece claro que podemos realizar 2^{10} combinaciones de valores $+1/-1$, por lo que, nuestro espacio X se compone de 1024 puntos.

El conjunto de hipótesis H_1 proporciona salida positiva $(+1)$ en un único punto, luego el número de hipótesis que contiene H_1 (cardinalidad de H_1) es igual al numero de puntos. Siendo más precisos, diríamos que la cardinalidad de H_1 es igual al número de maneras diferentes de elegir un sólo punto de entre 1024, es decir 1024:

$$|H_1| = \binom{1024}{1} = \frac{1024!}{1!(1024-1)!} = 1024$$

Si seguimos el mismo razonamiento, la cardinalidad del conjunto de hipótesis de H_{100} se calcula del mismo modo. Ahora debemos saber cuántas maneras diferentes hay de elegir de forma diferente 100 puntos de entre 1024:

$$|H_{100}| = \binom{1024}{100} = \frac{1024!}{1!(1024-100)!} \approx 7.747 \cdot 10^{140}$$

Como vemos, H_{100} tiene muchas hipótesis, pero es un número finito de ellas.

Apartado b)

H_1 tiene 1024 hipótesis que deben ser especificadas o representadas de forma inequívoca. Dado que un bit puede tomar únicamente dos valores $(+1/-1)$, necesitamos un número x de bits tal que $2^x = 1024$, es decir, un número de bits que nos permita representar una función de forma diferente a cualquier otra perteneciente a H_1. Para despejar x, basta con tomar logarítmo en base 2:

$$x = log_2(1024) = 10$$

Es decir, necesitamos 10 bits para representar de forma única una función de H_1.

Apartado c)

Si seguimos el mismo razonamiento que antes, sabemos que H_{100} tiene $\binom{1024}{100}$ hipótesis que deben ser representadas de forma única. Entonces necesitamos un x tal que $2^x = \binom{1024}{100}$. Tomamos logaritmos igual que antes y nos queda que:

$$x = \left\lceil log_2 \left(\binom{1024}{100} \right) \right\rceil \approx \lceil log_2(7.747 \cdot 10^{140}) \rceil = 469$$

Es decir, necesitaremos 469 bits para identificar inequívocamente todas las funciones de H_{100}.

2. Suponga que durante 5 semanas seguidas, recibe un correo postal que predice el resultado del partido de futbol del domingo, donde hay apuestas substanciosas. Cada lunes revisa la predicción y observa que la predicción es correcta en todas las ocasiones. El día de después del quinto partido recibe una carta diciendole que si desea conocer la predicción de la semana que viene debe pagar 50.000e. ¿Pagaría?

a) ¿Cuantas son las posibles predicciones gana-pierde para los cinco partidos?

Es muy sencillo. Supongamos que tenemos 5 bits que pueden tomar el valor 0 (pierde) o el valor 1 (gana), entonces tenemos $2^5 = 32$ combinaciones de pronósticos posibles.

b) Si el remitente desea estar seguro de que al menos una persona recibe de él la predicción correcta sobre los 5 partidos, ¿Cual es el mínimo número de cartas que deberá de enviar?

Para estar seguro de que al menos una persona recibe la combinación de 5 pronósticos correctos debemos cubrir todas las combinaciones posibles, es decir las 32 que calculamos en el apartado anterior.

c) Después de la primera carta prediciendo el resultado del primer partido, ¿a cuantos de los seleccionados inicialmente deberá de enviarle la segunda carta?

Sólo a 16, es decir, a la mitad. De entre todas las combinaciones realizadas tenemos 16 para las que el primer partido toma el valor "gana" y otras 16 para las que el primer partido toma el valor "pierde", como no se puede ganar y perder a la vez, tras el primer partido, habremos fallado en la mitad y las personas que hayan recibido las combinaciones en las que hemos fallado en el primer partido no van a pagar ni de broma, luego sólo tiene sentido seguir enviando cartas a aquéllas personas que recibieron la primera predicción correcta, es decir, a 16.

d) ¿Cuantas cartas en total se habrán enviado depués de las primeras cinco semanas?

Antes de que se jugara el primer partido enviamos 32 cartas donde 16 decían "gana" y 16 decían "pierde". Lógicamente habremos fallado en la mitad de los casos. Antes del segundo partido, sólo tiene sentido enviar 16 cartas, de las que volveremos a fallar en la mitad de las ocasiones, antes del tercer partido, enviaremos 8 cartas, antes del cuarto mandaremos 4 cartas y antes del quinto 2 cartas, de las que una contendrá la combinación de 5 resultados correcta.

En total, habremos enviado $32 + 16 + 8 + 4 + 2 \ = 62$ cartas

e) Si el coste de imprimir y enviar las cartas es de 0.5e por carta, ¿Cuanto ingresa el remitente si el receptor de las 5 predicciones acertadas decide pagar los 50.000e?

El remitente ingresará $50000 - (0.5 \cdot 62) = 49969$ euros, ya que, para calcular el beneficio neto será necesario restar a los 50000 euros que ingresa los 31 que gastó en imprimir y enviar las cartas.

f) ¿Puede relacionar esta situación con la función de crecimiento y la credibilidad del ajuste de los datos?

Sí. Tenemos 5 puntos (partidos) que pueden tomar el valor "gana" (1) o "pierde" (0). Por otro lado, tenemos una clase de funciones H, tal que la salida es $+1$ para sólo un punto, este es la combinación correcta de partidos. Dado que tenemos 32 combinaciones, la cardinalidad de la clase H es cuántas formas diferentes de elegir un punto de entre 32 hay, entonces, la cardinalidad de H es 32.

Con la función de crecimiento pretendemos saber cuántas hipótesis útiles hay, es decir, cuántas dicotomías (formas de separar puntos) hay diferentes. En nuestro caso, todas las hipótesis de las 32 posibles son útiles, pues dadas dos diferentes, nos dicen que la combinación correcta es diferente en cada caso.

Dicho esto, sabemos que el número máximo de dicotomías posibles dados 5 puntos es $2^5 = 32$. Por otro lado sabemos que H tiene 32 funciones útiles. La función de crecimiento nos dice que el número de dicotomías que se pueden lograr es $mH(N) \leq 2^N$. El número de hipótesis útiles de la clase H es 2^k, entonces si $2^k = 2^N$ dicotomías, no habría punto de ruptura, es exactamente lo que pasa: $mH(5) = 2^5$. Con esto, sabemos que la dimensión de Vapnik & Chervonenkis del conjunto de hipótesis H es infinita. Podríamos ver que la Dim. VC era infinita de igual modo concluyendo que H es un conjunto convexo.

Podemos acotar el error fuera de la muestra con la siguiente expresión:

$$E_{out}(h) \leq E_{in}(h) + \sqrt{\frac{8}{N} log\left(\frac{4((2N)^{d_{vc}} + 1)}{\delta}\right)}$$

Si la Dim. VC fuera finita, podríamos garantizar la generalización y la credibilidad de los datos. Como la Dim. VC es infinita y aparece en un exponencial, no podemos acotar el error fuera de la muestra y, por ende, la credibilidad de los datos será nula.

3. 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 que lo impida.

En este caso estamos intentando generalizar a partir de una muestra obtenida, es decir, aprender de datos. Como en esta asignatura, debemos preguntarnos ¿se puede aprender? La respuesta es afirmativa sí y sólo sí definimos un margen de error, es decir, si permitimos equivocarnos en cierto grado (δ). En este caso no hemos dicho nada de eso, sino que decimos que pretendemos extraer conclusiones específicas, que sin este grado permitido de error no serán validas.

Por otro lado, suponiendo aceptado un cierto margen de error, deberíamos preguntarnos ¿Qué es una muestra suficientemente grande? ¿Qué cosas debemos medir? porque deberíamos ser capaces, antes de nada, de calcular la dimension VC y establecer el número de muestras necesarias para conseguir un margen de error conforme a la expresión:

$$E_{out}(h) \leq E_{in}(h) + \sqrt{\frac{8}{N} log\left(\frac{4((2N)^{d_{vc}}+1)}{\delta}\right)}$$

Sin definir esto es imposible aprender de los datos y conseguir el objetivo.

4. 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.

Desde el momento en que se dice "Miramos los datos", el resto de conclusiones están contaminadas y son erróneas.

Vayamos desde el final hasta el principio. Hemos obtenido una cota del error de test, y ciertas conclusiones a partir de un valor de Dim VC obtenido de mirar de nuevo los datos. Dicho valor de Dim VC ha sido obtenido porque hemos definido una clase de funciones de determinada complejidad y un algoritmo de aprendizaje sujeto a que los datos "parecen" linealmente separables.

La forma correcta de proceder es elegir la clase de funciones y el algoritmo de aprendizaje antes de ni siquiera pensar en los datos. Si hemos visto los datos, estamos limitando la clase de funciones y el algoritmo de aprendizaje a algo medido a ojo. ¿Y si lo que creemos es erróneo? Toda conclusión a la que se llega es también errónea.

Suponiendo corregido esto, otro fallo es elegir la hipótesis que minimiza el error dentro de la muestra, lo que terminará con total seguridad en sobreajuste (en la medida en que la complejidad del conjunto de hipótesis lo permite) y por ende el error fuera de la muestra será inaceptablemente elevado.

5. Suponga que separamos 100 ejemplos de un conjunto D que no serán usados para entrenamiento sino que serán usados para seleccionar una de las tres hipótesis finales g_1, g_2 y g_3 producidas por tres algoritmos de aprendizaje distintos entrenados sobre el resto de datos. Cada algoritmo trabaja con un conjunto H de tamaño 500. Nuestro deseo es caracterizar la precisión de la estimación $E_{out}(g)$ sobre la hipótesis final seleccionada cuando usamos los mismos 100 ejemplos para hacer la estimación.

a) ¿Que expresión usaría para calcular la precisión? Justifique la decisión

De nuevo usaría la expresión de la cota de generalización.

$$E_{out}(h) \leq E_{in}(h) + \sqrt{\frac{8}{N} log\left(\frac{4((2N)^{d_{vc}}+1)}{\delta}\right)}$$

Con una salvedad. Dado que separamos 100 ejemplos cada vez, si separamos siempre los mismos la cota obtenida no será demasiado significativa, utilizaría validación cruzada, es decir, calcularía esta cota separando cada vez un conjunto de 100 puntos diferente, por ejemplo, 10 veces y calcularía un promedio.

b) ¿Cual es el nivel de contaminación de estos 100 ejemplos comparandolo con el caso donde estas muestras fueran usadas en el entrenamiento en lugar de en la selección final?

Los 100 puntos separados no han sido contaminados, pues no se han utilizado en entrenamiento. Si se hubieran utilizado, la contaminación es total, pues estaríamos preguntando al modelo por puntos cuya salida conoce. Sería como entregar un folio con un examen y otro folio con las soluciones a un alumno. El hecho de que saque un 10 no es significativo, pues no hemos medido sus conocimientos. Del mismo modo, el error cometido no es significativo, pues no estamos midiendo para nada su capacidad de predicción, es decir, no estaríamos midiendo el aprendizaje, se parecería más bien a una búsqueda en una base de datos.

6. Considere la tarea de seleccionar una regla del vecino más cercano. ¿Qué hay de erróneo en la siguiente lógica que se aplica a la selección de k? (Los límites son cuando $N \to \infty$). "*Considere la posibilidad de establecer la clase de hipótesis H_{NN} con N reglas, las k-NN hipótesis, usando $k = 1, ..., N$. Use el error dentro de la muestra para elegir un valor de k que minimiza E_{in}. Utilizando el error de generalización para N hipótesis, obtenemos la conclusión de que $E_{in} \to E_{out}$ porque $log\left(\frac{N}{N}\right) \to 0$. Por lo tanto concluimos que asintóticamente, estaremos eligiendo el mejor valor de k, basadonos solo en E_in.*"

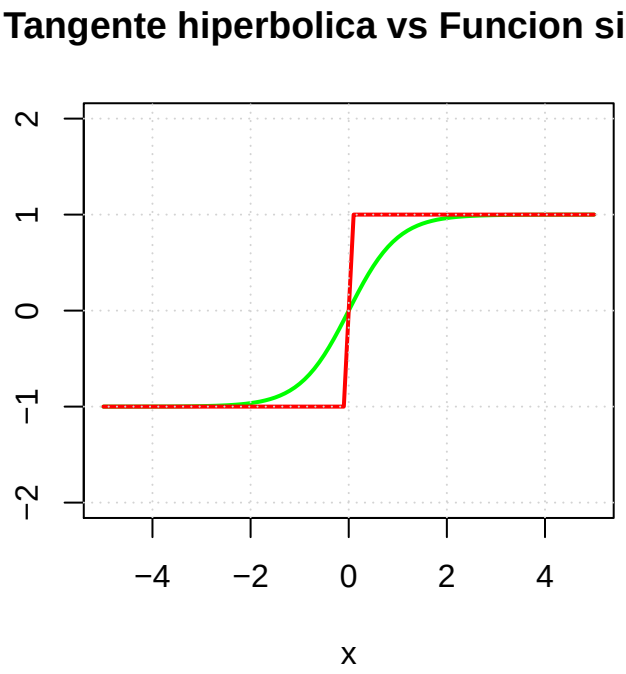
7.a) Considere un núcleo Gaussiano en un modelo de base radial. ¿Qué representa $g(x)$ (ecuación 6.2 del libro LfD) cuando $||x|| \to \infty$ para el modelo RBF no-paramétrico versus el modelo RBF paramétrico, asumiendo los w_n fijos?

7.b) Sea Z una matriz cuadrada de características definidas por $Z_{nj} = \phi_j(x_n)$ donde $\phi_j(x)$ representa una transformación no lineal. Suponer que Z es invertible. Mostrar que un modelo paramétrico de base radial, con $g(x) = w^T\phi(x)$ y $w = Z^{-1}y$, interpola los puntos de forma exacta. Es decir, que $g(x_n) = y_n$, con $E_{in}(g) = 0$.

7.c) ¿Se verifica siempre que $E_{in}(g) = 0$ en el modelo no-paramétrico?

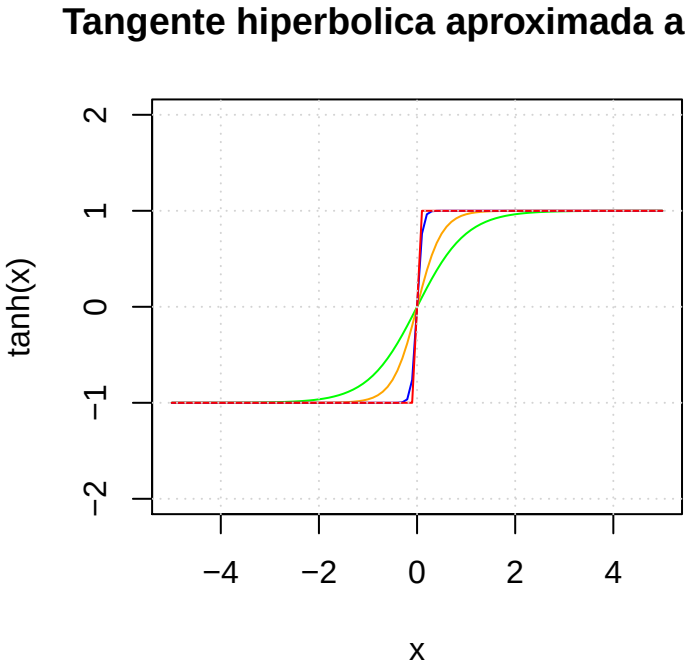
8. Verificar que la función sign puede ser aproximada por la función tanh. Dado w_1 y $\varepsilon > 0$ encontrar w_2 tal que $|sign(x_n^T w_1) - tanh(x_n^T w_2)| \leq \varepsilon$ para $x_n \in D$.

Veamos la diferencia entre $tanh(x)$, en verde y $sign(x)$, en rojo:



Es evidente que la función signo es muy parecida a la función tangente hiperbólica ¿Pero cómo "convertimos" la curva en escalón?. Sí dado un valor de x lo multiplicamos por un número α mayor que 1, αx será más grande o más pequeño (dependiendo del signo) que x. por lo que, al aplicar la función tangente hiperbólica, el valor $tanh(\alpha x)$ será más cercano a 1 ó -1 que $tanh(x)$.

Veámoslo con diversos valores de α. En verde, vemos la función tanh(x), en naranja, la función tanh$(2x)$ y en azul la función $tanh(10x)$. En rojo, la función $sign(x)$ original.



Para encontrar un w_2 tal que $|sign(x_n^T w_1) - tanh(x_n^T w_2)| \leq \varepsilon$ vamos a tratar la ecuación sin valor absoluto, entonces tenemos dos desigualdades:

$$sign(x_n^T w_1) - tanh(x_n^T w_2) \leq \varepsilon \Rightarrow -tanh(x_n^T w_2) \leq \varepsilon - sign(x_n^T w_1)$$
$$tanh(x_n^T w_2) - sign(x_n^T w_1) \leq \varepsilon \Rightarrow tanh(x_n^T w_2) \leq \varepsilon + sign(x_n^T w_1)$$

Para cambiar el signo a la expresión hay que voltear la desigualdad en el primer caso, en el segundo se queda como está:

$$tanh(x_n^T w_2) \geq sign(x_n^T w_1) - \varepsilon$$
$$tanh(x_n^T w_2) \leq sign(x_n^T w_1) + \varepsilon$$

Vamos a tomar la inversa de la tangente hiperbólica, es decir, arcotangente hiperbólica, para quedarnos sólo con los términos x_n^T y w_2.

$$arctanh(tanh(x_n^T w_2)) \geq arctanh(sign(x_n^T w_1) - \varepsilon) \Rightarrow x_n^T w_2 \geq arctanh(sign(x_n^T w_1) - \varepsilon)$$
$$arctanh(tanh(x_n^T w_2)) \leq arctanh(sign(x_n^T w_1) + \varepsilon) \Rightarrow x_n^T w_2 \leq arctanh(sign(x_n^T w_1) + \varepsilon)$$

Ahora sólo basta con despejar w_2 (Esta es la solución del ejercicio):

$$w_2 \geq x_n^{-T} arctanh(sign(x_n^T w_1) - \varepsilon)$$
$$w_2 \leq x_n^{-T} arctanh(sign(x_n^T w_1) + \varepsilon)$$

Los valores posibles de w_2 quedan acotados por estas dos expresiones.

Vamos ahora a probar con el mínimo valor en este intervalo, que sustituiremos en la expresión $|sign(x_n^T w_1) - tanh(x_n^T w_2)| \leq \varepsilon$, es decir $w_2 = x_n^{-T} arctanh(sign(x_n^T w_1) - \varepsilon$, entonces nos quedaría:

$$|sign(x_n^T w_1) - tanh(x_n^T x_n^{-T} arctanh(sign(x_n^T w_1) - \varepsilon))| \leq \varepsilon$$

$$|sign(x_n^T w_1) - sign(x_n^T w_1) - \varepsilon| \leq \varepsilon$$

$$|-\varepsilon| \leq \varepsilon$$

Como vemos, para el mínimo valor de la cota se cumplen las condiciones. Probemos para el máximo valor:

$$|sign(x_n^T w_1) - tanh(x_n^T x_n^{-T} arctanh(sign(x_n^T w_1) + \varepsilon))| \leq \varepsilon$$

$$|sign(x_n^T w_1) - sign(x_n^T w_1) + \varepsilon| \leq \varepsilon$$

$$|+\varepsilon| \leq \varepsilon$$

Si esto es así, vemos cómo cualquier valor de w_2 válido, es decir, acotado entre las dos expresiones obtenidas como solución, cumplirá las restricciones.

9. Sea V y Q el número de nodos y pesos en una red neuronal,

$$V = \sum_{l=0}^{L} d^{(l)}, \quad Q = \sum_{l=0}^{L} d^{(l)}(d^{(l+1)} + 1)$$

En términos de V y Q ¿cuántas operaciones se realizan en un pase hacia delante (sumas, multiplicaciones y evaluaciones de θ)?

10. Para el perceptrón sigmoidal $h(x) = tanh(x^T w)$, sea el error de ajuste $E_{in}(w) = \frac{1}{N}\sum_{n=1}^{N}(tanh(x_n^T w) - y_n)^2$. Mostrar que

$$\nabla E_{in}(w) = \frac{2}{N}\sum_{n=1}^{N}(tanh(x_n^T w) - y_n)(1 - tanh(x_n^T w)^2)x_n$$

Partamos de la expresión de $E_{in}(w)$:

$$E_{in}(w) = \frac{1}{N}\sum_{n=1}^{N}(tanh(x_n^T w) - y_n)^2$$

que reescribiremos como

$$E_{in}(w) = \frac{1}{N} * T(w)$$

Vayamos desde lo más superficial hasta lo más profundo. Lo primero que vemos es que la expresión de $E_{in}(w)$ es un producto, luego hay que hallar la derivada del producto, que es:

$$\nabla E_{in}(w) = 0 * T(w) + \frac{1}{N} * T'(w) = \frac{1}{N} * T'(w)$$

Vamos a calcular la $T'(w)$, con la regla de la cadena:

$$T(w) = \sum_{n=1}^{N}(tanh(x_n^T w) - y_n)^2 \Rightarrow$$

$$\Rightarrow T'(w) = \sum_{n=1}^{N} 2(tanh(x_n^T w) - y_n)(1 - tanh(x_n^T w)^2)x_n^T$$

Retomemos la expresión anterior, donde $\nabla E_{in}(w) = \frac{1}{N} * T'(w)$:

$$\nabla E_{in}(w) = \frac{1}{N} * T'(w) \Rightarrow \nabla E_{in}(w) = \frac{1}{N}\sum_{n=1}^{N} 2(tanh(x_n^T w) - y_n)(1 - tanh(x_n^T w)^2)x_n^T$$

Dado que 2 es constante, lo sacamos fuera de la sumatoria, y nos queda:

$$\nabla E_{in}(w) = \frac{2}{N}\sum_{n=1}^{N}(tanh(x_n^T w) - y_n)(1 - tanh(x_n^T w)^2)x_n^T$$

Si $w \to \infty$ ¿qué le sucede al gradiente? ¿Cómo se relaciona esto con la dificultad de optimizar el perceptrón multicapa?

Si $w \to \infty$, tenemos que el término $(1 - tanh(x_n^T w)^2)$ tiende a 0 y por tanto, toda la gradiente es 0. Cuando esto ocurre en una determinada capa, en backpropagation se entiende que ya se ha minimizado el error y que ya no se puede mejorar más, por lo que quedaríamos atrapados en un falso mínimo. Esta es la principal dificultad al optimizar el perceptrón multicapa, cuando se tienen pesos muy grandes, el error decrece y el perceptrón entiende que ya no puede mejorar, cuando esto es falso.