Aprendizaje Automático
Cuestionario de teoría 2

Alfonso García Martínez
alfonsogmw@correo.ugr.es

Abril y mayo de 2019

1. Identificar de forma precisa dos condiciones imprescindibles para que un problema de predicción puede ser aproximado por inducción desde una muestra de datos. Justificar la respuesta usando los resultados teóricos estudiados

   Solución: Las dos condiciones que debe cumplir la muestra de datos para poder inducir resultados a partir de ella es que sea **independiente e idénticamente distribuida** (i.i.d.).

   Cuando se dan estas dos condiciones, podemos hacer uso de la desigualdad de Hoeffding:

   $$\mathbb{P}\left[|\nu - \mu| > \epsilon\right] \leq 2e^{-2\epsilon^2 N} \qquad \forall \epsilon > 0$$

   Donde:

   - ν es la fracción o probabilidad de casos positivos en la muestra de una variable aleatoria binaria. En el modelo del recipiente de bolas rojas y verdes que planteamos en clase, equivale a la proporción de bolas rojas en la muestra que hemos extraído del recipiente.
   - μ es la fracción o probabilidad de casos positivos en toda la población de una variable aleatoria binaria. En el modelo del recipiente, equivale a la proporción de bolas rojas en todo el recipiente.
   - ϵ es un valor positivo de *tolerancia* (normalmente pequeño) que nosotros fijamos.
   - N es el tamaño de la muestra.

   Esta desigualdad es de gran relevancia, porque gracias a ella sabemos que ν *nos dice algo de* μ, pero ya profundizaremos en ello más adelante en este ejercicio.

   La desigualdad de Hoeffding puede aplicarse al problema del aprendizaje si damos cierto significado a ν y μ. En tal caso, se expresa como:

   $$\mathbb{P}\left[|E_{\text{in}}(h) - E_{\text{out}}(h)| > \epsilon\right] \leq 2e^{-2\epsilon^2 N} \qquad \forall \epsilon > 0$$

   Donde:

- $E_{\text{in}}(h)$ es el error dentro de la muestra dado un modelo h:

$$E_{\text{in}}(h) = \frac{1}{N}\sum_{n=1}^{N}[\![h(\mathbf{x}_n) \neq f(\mathbf{x}_n)]\!]$$

- $E_{\text{out}}(h)$ es el error fuera de la muestra dado un modelo h:

$$E_{\text{out}}(h) = \mathbb{P}\left[h(\mathbf{x}_n) \neq f(\mathbf{x}_n)\right]$$

Obsérvese que el lado derecho de la desigualdad no depende de h, sino únicamente de ϵ (que como ya hemos dicho lo escogemos nosotros) y de N. Así pues, la cota que esta desigualdad establece disminuye exponencialmente en $\epsilon^2 N$.

Dicho esto, la idea intuitiva es que, cuanto mayor sea nuestra muestra (cuanto mayor sea N), menos probable será que la diferencia entre el error dentro de la muestra y el error fuera de la muestra sea mayor que ϵ. Dicho en otras palabras: dada una hipótesis h, a mayor tamaño muestral, más se parecerá $E_{\text{out}}(h)$ a $E_{\text{in}}(h)$.

De alguna forma, con la desigualdad de Hoeffding estamos sacando conclusiones de la población a partir de la muestra obtenida, porque gracias a ella sabemos que $E_{\text{in}}(h) \approx E_{\text{out}}(h)$ **si disponemos de una muestra aleatoria (i.i.d.) lo suficientemente grande**.

En conclusión, la teoría de la probabilidad nos dice que existen, bajo ciertas condiciones, dependencias probabilísticas entre una población y una muestra suya que posibilitan el aprendizaje inductivo. Y esas condiciones no son otras que el hecho de que la muestra sea i.i.d.

2. El jefe de investigación de una empresa con mucha experiencia en problemas de predicción de datos tras analizar los resultados de los muchos algoritmos de aprendizaje usados sobre todos los problemas en los que la empresa ha trabajado a lo largo de su muy dilatada existencia, decide que para facilitar el mantenimiento del código de la empresa van a seleccionar un único algoritmo y una única clase de funciones con la que aproximar todas las soluciones a sus problemas presentes y futuros. ¿Considera que dicha decisión es correcta y beneficiará a la empresa? Argumentar la respuesta usando los resultados teóricos estudiados.

   Solución: Esta decisión no va a beneficiar a la empresa.

   Según el teorema NFL (*No Free Lunch*), para cualquier algoritmo, existe un problema (o distribución de probabilidad) $\mathcal{P}$ para el que falla, y que sin embargo existe otro algoritmo que sí obtiene buenos resultados con $\mathcal{P}$. Si un algoritmo funciona bien para una cierta clase de problemas, su desempeño no será tan bueno para el resto de problemas. Esto hace que el promedio de rendimiento de cualquier algoritmo en todos los problemas posibles sea siempre el mismo

   Cuando nos enfrentamos a un problema, debemos elegir el algoritmo y la clase de funciones adecuados para ese problema, porque el que todos los sistemas de aprendizaje tengan el mismo rendimiento promedio en todos los problemas posibles no significa que siempre tengan el mismo rendimiento para todos los problemas. Más bien al contrario: el teorema NFL nos indica que **existen problemas con los que un sistema de aprendizaje falla mientras que con otros sistemas se obtiene un buen desempeño**.

   Por ese motivo, elegir siempre el mismo algoritmo y la misma clase de funciones es un craso error, porque si el sistema que se haya elegido falla para un cierto problema, **nos estaremos cerrando las puertas a encontrar otro algoritmo u otra clase de funciones que sí funcionen para ese mismo problema**.

3. ¿Que se entiende por una solución PAC a un problema de aprendizaje? Identificar el porqué de la incertidumbre e imprecisión.

   Solución: PAC son las siglas de *Probably Approximately Correct*, es decir, una solución PAC es probable y aproximadamente correcta, de forma que la hipótesis que hemos escogido tendrá una **alta probabilidad** de tener un error bajo tanto dentro como fuera de la muestra, de forma que el error fuera de la muestra **se aproxime lo más posible** al error dentro de la muestra.

   Partiendo de la desigualdad original de Hoeffding:

   $$\mathbb{P}\left[|E_{\text{in}}(h) - E_{\text{out}}(h)| > \epsilon\right] \leq 2e^{-2\epsilon^2 N}$$

   $|E_{\text{in}}(h) - E_{\text{out}}(h)| > \epsilon$ mide cómo aproximamos nuestro resultado, y esa aproximación, según la desigualdad, tiene cierta una cierta probabilidad en función de N.

   Si definimos el valor $\delta = 2e^{-2N\epsilon^2}$, entonces $\epsilon = \sqrt{\frac{1}{2N}\ln\frac{2}{\delta}}$, con lo que podemos acotar el error fuera de la muestra:

   $$E_{\text{out}}(h) \leq E_{\text{in}}(h) + \sqrt{\frac{1}{2N}\ln\frac{2}{\delta}}$$

   ϵ siempre nos indicará que existe una imprecisión, puesto que siempre habrá cierta diferencia entre $E_{\text{out}}(h)$ y $E_{\text{in}}(h)$ que hará que no sea más que una aproximación.

   δ por su parte es una tolerancia que nos dice con qué probabilidad $(1-\delta)$ la cota es cierta. Como estamos hablando en términos probabilísticos, siempre habrá cierta incertidumbre sobre hasta qué punto se cumple la desigualdad.

4. Suponga un conjunto de datos $\mathcal{D}$ de 25 ejemplos extraidos de una funcion desconocida $f : \mathcal{X} \to \mathcal{Y}$, donde $\mathcal{X} = \mathbb{R}$ e $\mathcal{Y} = \{-1, +1\}$. Para aprender f usamos un conjunto simple de hipótesis $\mathcal{H} = \{h_1, h_2\}$ donde h_1 es la función constante igual a +1 y h_2 la función constante igual a -1. Consideramos dos algoritmos de aprendizaje, S(smart) y C(crazy). S elige la hipótesis que mejor ajusta los datos y C elige deliberadamente la otra hipótesis.

   a) ¿Puede S producir una hipótesis que garantice mejor comportamiento que la aleatoria sobre cualquier punto fuera de la muestra? Justificar la respuesta

   Solución: No, **no hay garantías de que S se comporte mejor que la aleatoria**. Sí es cierto que eso es algo muy poco probable, tal y como se ve en la desigualdad de Hoeffding: si tenemos poco error en $\mathcal{D}$, es **altamente probable** que también tengamos poco error en los puntos generados fuera de $\mathcal{D}$.

   Sin embargo, estas conclusiones las sacamos desde un enfoque probabilístico. Desde un punto de vista puramente determinista, no hay ninguna ley que nos asegure que si lo hacemos bien dentro de la muestra, lo hagamos igual de bien fuera de ella.

5. Con el mismo enunciado de la pregunta 4:

   a) Asumir desde ahora que todos los ejemplos en $\mathcal{D}$ tienen $y_n = +1$. ¿Es posible que la hipotesis que produce C sea mejor que la hipótesis que produce S? Justificar la respuesta.

   Solución: **Sí es posible un escenario en el que C sea mejor que S.** Veamos por qué:

   En este escenario, S elegirá la hipótesis h_1, mientras que C tendrá a h_2.

   En $\mathcal{D}$, está claro lo que ocurrirá: S hará una clasificación perfecta, sin errores, mientras que C se equivocará del todo.

   La cuestión está en qué pasará con los puntos fuera de $\mathcal{D}$. Debemos tener en cuenta la distribución de probabilidad que determine la proporción o probabilidad p de casos en los que $f(\mathbf{x}) = +1$:

   - Si $p \approx 1 \rightarrow$ la mayoría de puntos fuera de la muestra tendrían un $f(\mathbf{x}) = +1$, luego S acertaría en la mayoría de ellos y C fallaría en la mayoría.
   - Si $p \approx 0 \rightarrow$ la mayoría de puntos fuera de la muestra tendrían un $f(\mathbf{x}) = -1$, luego S se equivocaría en la mayoría de ellos y C acertaría.

   Desde un punto de vista probabilístico, es muy probable que $p \approx 1$ si todos los puntos en $\mathcal{D}$ tienen $y_n = +1$ (si suponemos que $\mathcal{D}$ es una muestra i.i.d. y que 25 es un tamaño muestral suficiente), en cuyo caso $\mathcal{D}$ representaría bien la distribución de probabilidad subyacente y S sería claramente superior a C.

   Pero de nuevo, desde un enfoque determinista, nada nos garantiza que la proporción de casos positivos en $\mathcal{D}$ se tenga que acercar a p. Es perfectamente posible (que no *probable*) que $\mathcal{D}$ no sea una muestra representativa aún siendo i.i.d., en cuyo caso C podría dar mejores resultados que S.

6. Considere la cota para la probabilidad de la hipótesis solución g de un problema de aprendizaje, a partir de la desigualdad generalizada de Hoeffding para una clase finita de hipótesis,

$$\mathbb{P}\left[|E_{\text{out}}(g) - E_{\text{in}}(g)| > \epsilon\right] \leq \delta$$

a) ¿Cuál es el algoritmo de aprendizaje que se usa para elegir g?

Solución: Se usa el algoritmo de aprendizaje que genere una g que **minimice el error dentro de la muestra**.

Con la desigualdad de Hoeffding sabemos únicamente que, muy probablemente, $E_{\text{in}}(g) \approx E_{\text{out}}(g)$.

Si con nuestro algoritmo logramos que $E_{\text{in}}(g) \approx 0$, también estaremos consiguiendo que $E_{\text{out}}(g) \approx 0$ con una alta probabilidad, y ese es justamente el objetivo fundamental del problema del aprendizaje.

b) Si elegimos g de forma aleatoria ¿seguiría verificando la desigualdad?

Solución: Sí. Como esta desigualdad ya tiene en cuenta a todas las funciones de la clase, la cota es válida para cualquiera de ellas, por muy malos resultados que se obtengan con g. Recordemos que la desigualdad de Hoeffding nos habla de lo probable que es que $E_{\text{in}}(g) \approx E_{\text{out}}(g)$ sea cual sea g. Que $E_{\text{in}}(g) \approx 0$ es otra cosa distinta que sí varía en función de la g escogida.

c) ¿Depende g del algoritmo usado?

Solución: Naturalmente. La g que se use será distinta según cómo la ajuste el algoritmo.

Aunque lo ideal es que el algoritmo devuelva la g que minimice $E_{\text{in}}(g)$ tal y como hemos comentado en el apartado *a*), puede haber algoritmos que funcionen de otra forma y en base a otros criterios. De nuevo, que $E_{\text{in}}(g) \approx E_{\text{out}}(g)$ y que $E_{\text{in}}(g) \approx 0$ son cosas diferentes, y esto último es algo que, efectivamente, dependerá

del algoritmo.

d) ¿Es una cota ajustada o una cota laxa?

Solución: Es una cota más bien laxa.

La desigualdad de Hoeffding para clases de funciones finitas no es incorrecta ni mucho menos, ya que tiene en cuenta a todas las posibles funciones de una clase dada.

El problema es que, precisamente, el tener en cuenta *todas* las funciones provoca solapamientos o intersecciones (*overlapping*) debidas a la existencia de funciones que generan resultados (dicotomías en el caso de la clasificación binaria) idénticos o muy similares, por lo que al acotar, con la suma de probabilidades, la probabilidad de la unión de los eventos "$|E_{\text{in}}(h_i) - E_{\text{out}}(h_i)| > \epsilon$"para cada función $h_i \in \mathcal{H}$, no se tienen en cuenta las intersecciones que habría que restar, de forma que la cota se vuelve laxa, poco exacta.

La combinatoria da una solución a este inconveniente con el principio de inclusión-exclusión, pero el problema está en calcular las intersecciones. Otra manera de solventarlo consiste en calcular de alguna forma el número de funciones o hipótesis efectivas por medio de la función de crecimiento y la dimensión de Vapnik-Chervonenkis (d_{VC}) de una clase de funciones, que es justo lo que se hace cuando la clase está compuesta por infinitas funciones.

7. ¿Por qué la desigualdad de Hoeffding definida para clases $\mathcal{H}$ de una única función no es aplicable de forma directa cuando el número de hipótesis de $\mathcal{H}$ es mayor de 1? Justificar la respuesta.

   Solución: La desigualdad de Hoeffding definida para clases $\mathcal{H}$ de una única función es la misma que se aplica cuando hemos prefijado la hipótesis h antes de obtener la muestra y sólo la tenemos a ella en cuenta:

   $$\mathbb{P}\left[|E_{\text{in}}(h) - E_{\text{out}}(h)| > \epsilon\right] \leq 2e^{-2\epsilon^2 N}$$

   Pero, cuando debemos tener en cuenta más de una función entre las que podamos elegir, debemos calcular la probabilidad de la unión de los eventos "$|E_{\text{in}}(h_i) - E_{\text{out}}(h_i)| > \epsilon$"para cada función $h_i \in \mathcal{H}$, que puede ser acotada sumando las probabilidades de cada uno de acuerdo con la desigualdad de Boole:

   $$\mathbb{P}\left[\mathcal{B}_1 \cup \mathcal{B}_2 \cup \cdots \cup \mathcal{B}_M\right] \leq \mathbb{P}\left[\mathcal{B}_1\right] + \mathbb{P}\left[\mathcal{B}_2\right] + \cdots + \mathbb{P}\left[\mathcal{B}_M\right]$$

   Con lo cual:

   $$\begin{gathered}
   \mathbb{P}\left[|E_{\text{in}}(h) - E_{\text{out}}(h)| \leq \epsilon\right] \leq \\
   \mathbb{P}[|E_{\text{in}}(h_1) - E_{\text{out}}(h_1)| > \epsilon \\
   \vee |E_{\text{in}}(h_2) - E_{\text{out}}(h_2)| > \epsilon \\
   \vee \ldots \\
   \vee |E_{\text{in}}(h_{|\mathcal{H}|}) - E_{\text{out}}(h_{|\mathcal{H}|})| > \epsilon] \\
   \leq \sum_{i=1}^{|\mathcal{H}|} \mathbb{P}\left[|E_{\text{in}}(h_i) - E_{\text{out}}(h_i)|\right]
   \end{gathered}$$

   Y aplicando la desigualdad de Hoeffding en todos los términos de la sumatoria a la vez, nos queda:

   $$\mathbb{P}\left[|E_{\text{in}}(h) - E_{\text{out}}(h)| > \epsilon\right] \leq 2|\mathcal{H}|e^{-2\epsilon^2 N}$$

   La diferencia entre esta expresión y la desigualdad original está únicamente en la inclusión de $|\mathcal{H}|$. Por ello, si solo tenemos una función en la clase, entonces $|\mathcal{H}| = 1$ y esta expresión equivale a la original. Pero cuando tenemos más de una función en $\mathcal{H}$, entonces se ha de usar forzosamente la desigualdad que tiene incluye a $|\mathcal{H}|$.

8. Si queremos mostrar que k^* es un punto de ruptura para una clase de funciones $\mathcal{H}$ cuales de las siguientes afirmaciones nos servirían para ello:

   a) Mostrar que existe un conjunto de k^* puntos $x_1, \ldots, x_{k^*}$ que $\mathcal{H}$ puede separar ("shatter").

   b) Mostrar que $\mathcal{H}$ puede separar cualquier conjunto de k^* puntos.

   c) Mostrar un conjunto de k^* puntos $x_1, \ldots, x_{k^*}$ que $\mathcal{H}$ no puede separar.

   d) Mostrar que $\mathcal{H}$ no puede separar ningún conjunto de k^* puntos

   e) Mostrar que $m_{\mathcal{H}}(k) = 2^{k^*}$

Solución: Antes de dar cualquier respuesta, recordemos algunos conceptos relevantes:

La **función de crecimiento** de una clase de funciones $\mathcal{H}$ nos indica el **máximo número de dicotomías** que $\mathcal{H}$ puede generar en N puntos. Es decir, si considerásemos todas las posibles muestras de de N puntos de X, cogeríamos aquella que permitiese un mayor número de dicotomías. Formalmente, la función de crecimiento se describe de la siguiente manera:

$$m_{\mathcal{H}}(k) = \max_{\mathbf{x}_1, \ldots, \mathbf{x}_N \in \mathcal{X}} |\mathcal{H}(\mathbf{x}_1, \ldots, \mathbf{x}_N)|$$

Donde $\mathcal{H}(\mathbf{x}_1, \ldots, \mathbf{x}_N)$ es el conjunto de todas las dicotomías generadas por la clase $\mathcal{H}$ en una muestra de puntos $\mathbf{x}_1, \ldots, \mathbf{x}_N$:

$$\mathcal{H}(\mathbf{x}_1, \ldots, \mathbf{x}_N) = \{(h(\mathbf{x}_1), \ldots, h(\mathbf{x}_N)) \mid h \in \mathcal{H}\}$$

Por una cuestión de combinatoria, como las dicotomías consisten en etiquetados binarios, como mucho podremos hacer 2^N dicotomías sea cual sea la clase $\mathcal{H}$ que tengamos. Por tanto, la función de crecimiento siempre estará acotada superiormente para cualquier $\mathcal{H}$:

$$m_{\mathcal{H}}(N) \leq 2^N$$

Si una clase $\mathcal{H}$ es capaz de hacer todas las posibles 2^N dicotomías en un conjunto de puntos $\mathbf{x}_1, \ldots, \mathbf{x}_N$, entonces decimos que $\mathcal{H}$ puede **separar**

(*shatter*) esos puntos.

k es un **punto de ruptura** cuando $\mathcal{H}$ no es capaz separar ningún conjunto de k puntos $x_1, \ldots, x_k$. De alguna forma, el punto de ruptura pone de manifiesto las limitaciones de $\mathcal{H}$. Como es de ver, esto no tiene nada que ver con lo dicho en a) y b)

Por su parte, la definición d) es correcta para definir un punto de ruptura. Con que exista un solo conjunto de k^* puntos para el cual $\mathcal{H}$ pueda hacer las 2^{k^*} dicotomías posibles (o sea, que $\mathcal{H}$ pueda separarlo), entonces k^* no sería un punto de ruptura. No es necesario que $\mathcal{H}$ separe todos los conjuntos de k^* puntos posibles. Esto es contrario a la definición de c): no basta con encontrar un conjunto de k^* puntos que $\mathcal{H}$ no pueda separar, sino que hay que demostrar que no puede hacerlo con ningún conjunto de k^* puntos.

Además, si $\mathcal{H}$ puede hacer las 2^{k^*} separaciones aunque sea para un conjunto de k^* puntos, eso significa que su función de crecimiento cumple que $m_{\mathcal{H}} = 2^{k^*}$, ya que 2^{k^*} es el máximo de dicotomías para k^* puntos. Si k^* fuese efectivamente un punto de ruptura, $\mathcal{H}$ no podría llegar a hacer el máximo de dicotomías, por lo que $m_{\mathcal{H}}(k^*) < 2^{k^*}$. Luego la definición e) tampoco sirve.

En resumen: la única afirmación válida para definir un punto de ruptura es la d).

9. Para un conjunto $\mathcal{H}$ con $d_{VC} = 10$, ¿qué tamaño muestral se necesita (según la cota de generalización) para tener un 95 % de confianza (δ) de que el error de generalización (ϵ) sea como mucho 0.05?

   Solución: La cota de generalización nos dice que:

$$E_{\text{out}}(g) \leq E_{\text{in}}(g) + \sqrt{\frac{8}{N} \ln \frac{4m_{\mathcal{H}}(2N)}{\delta}}$$

$$E_{\text{out}}(g) - E_{\text{in}}(g) \leq \sqrt{\frac{8}{N} \ln \frac{4m_{\mathcal{H}}(2N)}{\delta}}$$

   Y, como ya sabemos de antes:

$$|E_{\text{out}}(g) - E_{\text{in}}(g)| > \epsilon$$

   Entonces

$$\sqrt{\frac{8}{N} \ln \frac{4m_{\mathcal{H}}(2N)}{\delta}} \leq \epsilon$$

   De donde podemos *aislar* N en la parte izquierda:

$$N \geq \frac{8}{\epsilon^2} \ln \frac{4m_{\mathcal{H}}(2N)}{\delta}$$

   Y si desde el principio sustituimos $m_{\mathcal{H}}(2N)$ por su cota polinómica:

$$m_{\mathcal{H}}(2N) \leq N^{d_{VC}} + 1$$

   Obtenemos

$$N \geq \frac{8}{\epsilon^2} \ln \frac{4(N^{d_{VC}} + 1)}{\delta}$$

   Obsérvese que N está a ambos lados de la desigualdad, luego se deberá calcular numéricamente con un método iterativo con la siguiente regla de actualización:

$$N(t+1) \leftarrow \frac{8}{\epsilon^2} \ln \frac{4(N(t)^{d_{VC}} + 1)}{\delta}$$

Implementando el método iterativo en un sencillo *script* en Python, fijamos un tamaño inicial $N(0) = 1000$, un $d_{VC} = 10$, un $\delta = 0{,}05$, un $\epsilon = 0{,}05$ y una precisión de 10^{-14} para la diferencia entre los resultados de dos iteraciones consecutivas, obtenemos que $N(t_{final}) \approx 452956{,}865$ tras 17 iteraciones, de forma que, redondeando:

$$N \geq 452957$$

10. Considere que le dan una nuestra de tamaño N de datos etiquetados $\{-1, +1\}$ y le piden que encuentre la función que mejor ajuste dichos datos. Dado que desconoce la verdadera función f , discuta los pros y contras de aplicar los principios de inducción ERM y SRM para lograr el objetivo. Valore las consecuencias de aplicar cada uno de ellos.

Solución: La cota que se usa para la regla ERM es

$$E_{\text{out}}(g) \leq E_{\text{in}}(g) + \sqrt{\frac{8}{N} \ln \frac{4(N^{d_{VC}} + 1)}{\delta}} =$$
$$E_{\text{out}}(g) \leq E_{\text{in}}(g) + O(\sqrt{\frac{d_{VC} \ln N - \ln \delta}{N}}) =$$
$$E_{\text{out}}(g) \leq E_{\text{in}}(g) + \Omega(N, \mathcal{H}, \delta)$$

Donde $\Omega(N, \mathcal{H}, \delta)$ se interpreta como la penalización por la complejidad del modelo. en ERM, si utilizamos un modelo muy complejo, este tenderá a hacer *overfitting*, haciendo que $E_{\text{out}}(g)$ diste de $E_{\text{in}}(g)$, es decir, habrá mucha penalización a $E_{\text{out}}(g)$.

Cuando d_{VC} es finita, ERM suele ser una buena garantía de que $E_{\text{out}}(g) \approx E_{\text{in}}(g)$. Sin embargo, cuando $\frac{N}{d_{VC}} < 20$, entonces nuestro modelo es demasiado complejo para lo pequeña que es nuestra muestra, y la penalización en ERM será excesivamente grande.

En esos casos, es mejor usar el criterio SRM, en los que se utilizan subconjuntos de la clase $\mathcal{H}$ con menores d_{VC}, y se busca aquel subconjunto que produzca un menor error. De esta forma, estamos reduciendo la complejidad del modelo para aquellos casos en los que la complejidad es demasiado alta en comparación con el tamaño de la muestra.

Material citado y/o consultado

- *Learning from Data*, Y.S. Abu-Mustafa, M. Magdom-Ismail, H. Lin, AMLbook.com, 2012.
- *No Free Lunch Theorems for Optimization*, David H. Wolpert and William G. Macready, 1997.