Aprendizaje Automático (2018-2019)
Grado en Ingeniería Informática
Universidad de Granada

Cuestionario 2

Montserrat Rodríguez Zamorano

4 de mayo de 2019

Índice

1. Preguntas obligatorias

1.1. Ejercicio 1

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

Para que un problema de predicción pueda ser aproximado por inducción desde una muestra de datos tienen que darse dos condiciones:

- $E_{in}(g) \approx E_{out}(g)$
- $E_{in}(g) \approx 0$

Juntos, tenemos $E_{out}(g) \approx 0$, que es lo que necesitamos para considerar que el aprendizaje ha sido exitoso. La primera condición se verifica gracias a la desigualdad de Hoeffding.

$$P(\mathcal{D} : |E_{in}(g) - E_{out}(g)| > \epsilon) < 2e^{-\epsilon^2 N}$$

que nos dice que la probabilidad de que la diferencia entre E_{out} y E_{in} sea mayor que un ϵ dado disminuirá exponencialmente con el tamaño de la muestra N.

La segunda se consigue a través del algoritmo de aprendizaje. Dependerá de la complejidad de la función. Si la función es simple, se necesitará un N más pequeño para conseguir $E_{in}(g) \approx 0$. Si la función es compleja, se necesitará un tamaño de muestra más grande.

1.2. Ejercicio 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.

No considero que esta decisión sea buena para la empresa. Para argumentarlo, se puede hacer uso del teorema de *No-Free-Lunch*. El teorema de *No-Free-Lunch* establece que para cada algoritmo existe un problema en el que falla, aunque sea un problema que otro algoritmo pueda resolver con éxito. Además, todos los algoritmos tienen una eficacia o rendimiento equivalente en media cuando se consideran todas las posibles f.

El jefe ha podido verse confundido por esta última parte del algoritmo. Sin embargo, aunque el rendimiento en media sea equivalente, seleccionar un único algoritmo y una única clase de funciones restringimos mucho el espacio de soluciones y no tenemos en cuenta la naturaleza y conocimiento específico del problema, no pudiendo así garantizar el éxito en su resolución.

1.3. Ejercicio 3

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

Una solución PAC a un problema de aprendizaje es aquella que es, con una alta probabilidad, una buena aproximación.

La incertidumbre o imprecisión viene de la muestra que se tenga. Hay posibilidades de que el *training sample* no sea representativo de la realidad. Por ejemplo, si consideramos un jarrón con 50 bolas verdes y 50 bolas rojas y cogemos 10, podemos tener la "mala suerte"de que nuestra muestra tenga solo bolas verdes.

Por este tipo de casos en los que la muestra no es representativa se habla de "solución **probablemente** correcta".

1.4. Ejercicio 4

Suponga un conjunto de datos $\mathcal{D}$ de 25 ejemplos extraídos de una función 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.

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

Si el algoritmo S escoge la hipótesis h_1 y C escoge h_2, S sólo puede garantizar mejor comportamiento que C sobre cualquier punto si todos los puntos del conjunto de datos fuera de la muestra tienen la etiqueta $+1$. En otro caso, si un punto tiene la etiqueta -1, el algoritmo C tendría un mejor comportamiento.

Análogamente, si S escoge h_2, se podría garantizar un mejor comportamiento en cualquier punto fuera de la muestra que la aleatoria sólo si cualquiera de estos puntos tiene etiqueta -1.

1.5. Ejercicio 5

Con el mismo enunciado de la pregunta 4: asumir desde ahora que todos los ejemplos en $\mathcal{D}$ tienen $y_n = +1$. ¿Es posible que la hipótesis que produce C sea mejor que la hipótesis que produce S? Justificar la respuesta.

Nos encontramos en un caso similar al que describimos en el ejercicio 3. Si de un jarrón con 100 bolas, 50 verdes y 50 rojas cogemos 10, podemos tener la "mala suerte"de que todas sean verdes y por tanto no sea una muestra representativa de lo que hay en el jarrón.

En este caso, puede ser que todos los datos tengan $y = +1$ dentro de la muestra, por lo que S escogería la hipótesis h_1, pero que sin embargo fuera de ella todos los puntos o la mayoría de ellos tengan la etiqueta -1. De esta forma, C escogería la hipótesis con un mejor comportamiento.

1.6. Ejercicio 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,

$$(\mathbb{P}[|E_{out}(g) - E_{in}(g)|] > \epsilon) < \delta$$

- *¿Cuál es el algoritmo de aprendizaje que se usa para elegir g?*

  Buscaremos un algoritmo de aprendizaje que garantice $E_{in} \approx 0$, es decir, que minimice el error empírico (ERM), de forma que junto con la desigualdad de Hoeffding, con una muestra lo suficientemente grande, se cumplirá $E_{out} \approx 0$.

- *Si elegimos g de forma aleatoria, ¿seguiría verificando la desigualdad?*

  Sí, lo que nos dice la desigualdad es que la probabilidad de que la diferencia entre E_{out} y E_{in} sea mayor que un ϵ dado disminuirá exponencialmente con el tamaño de la muestra, no se habla de la función.

- *¿Depende g del algoritmo usado?*

  Hay muchos tipos de algoritmos de aprendizaje, por lo que g obviamente dependerá del algoritmo que se escoja.

- *¿Es una cota ajustada o una cota laxa?*

  Es una cota ajustada, ya que sabemos que la probabilidad no puede ser menor que 0, por lo que ya tenemos una cota inferior, y la desigualdad nos proporciona la cota superior.

1.7. Ejercicio 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.

Si consideramos $\mathcal{H} = \{h\}$, con h fijada antes de conocer la muestra de datos, podemos calcular $E_{out}(h)$ y $E_{in}(h)$ de forma directa y aplicar la desigualdad.

Sin embargo, si consideramos el caso de $\mathcal{H}$ finita (en concreto $|\mathcal{H}| > 1$), necesitamos adaptar la desigualdad, por lo que no es aplicable de forma directa. Una solución es considerar una hipótesis genérica g compatible con todas las hipótesis en $\mathcal{H}$.

$$\{\mathcal{D} : |E_{in}(g) - E_{out}(g)| > \epsilon\} = \bigcup_{h_i \in \mathcal{H}} (\{\mathcal{D} : |E_{in}(h_i) - E_{out}(h_i)| > \epsilon\})$$

Y una vez hecho este primer paso, ya podremos aplicar la desigualdad:

$$P(\{\mathcal{D} : |E_{in}(h_i) - E_{out}(h_i)| > \epsilon\}) < 2|\mathcal{H}|e^{-2\epsilon^2 N}$$

1.8. Ejercicio 8

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

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

  En el libro *Learning from data: Training vs Test*, tenemos una definición que, traducida directamente del inglés nos dice: "Si ningún conjunto de k puntos puede ser separado por $\mathcal{H}$, entonces se dice que k es un punto de ruptura para $\mathcal{H}$". Por tanto, esta afirmación no nos sirve para ello.

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

  En este caso, que la afirmación no nos sirve se deduce del apartado anterior: si $\mathcal{H}$ puede separar cualquier conjunto de k^* puntos, en concreto existe uno que $\mathcal{H}$ pueda separar. Por tanto, nos encontramos en el caso anterior y la afirmación no nos sirve.

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

  Que un conjunto no se pueda separar no implica que ninguno de ellos pueda ser separado por $\mathcal{H}$. Por tanto, esta afirmación tampoco es válida.

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

  En este caso, nos encontramos ante la definición que se ha dado en el primer apartado, por lo que esta afirmación sí nos sirve para mostrar que k^* es punto de ruptura de $\mathcal{H}$.

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

  Tenemos que si k^* es un punto de ruptura, entonces se cumple $m_{\mathcal{H}}(k) < 2^{k^*}$. Por tanto, esta afirmación no nos sirve para mostrar que k^* es punto de ruptura.

1.9. Ejercicio 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?

Se utiliza la siguiente fórmula para estimar el tamaño muestral que se necesita.

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

Utilizamos un proceso iterativo. Teniendo en cuenta que $d_{VC} = 10$, $\delta = 1 - 0{,}95 = 0{,}05$ y $\epsilon = 0{,}05$ comenzamos con un tamaño muestral $N = 1000$ y vemos como converge.

$$N \geq \frac{8}{0{,}05^2} ln\left(\frac{4((2 * 1000)^{10} + 1)}{0{,}05}\right) \approx 257251$$

$$N \geq \frac{8}{0{,}05^2} ln\left(\frac{4((2*257251)^{10}+1)}{0{,}05}\right) \approx 434853$$

$$N \geq \frac{8}{0{,}05^2} ln\left(\frac{4((2*434853)^{10}+1)}{0{,}05}\right) \approx 451652$$

$$N \geq \frac{8}{0{,}05^2} ln\left(\frac{4((2*451652)^{10}+1)}{0{,}05}\right) \approx 452865$$

$$N \geq \frac{8}{0{,}05^2} ln\left(\frac{4((2*452865)^{10}+1)}{0{,}05}\right) \approx 452950$$

$$N \geq \frac{8}{0{,}05^2} ln\left(\frac{4((2*452950)^{10}+1)}{0{,}05}\right) \approx 452956$$

$$N \geq \frac{8}{0{,}05^2} ln\left(\frac{4((2*452956)^{10}+1)}{0{,}05}\right) \approx 452957$$

$$N \geq \frac{8}{0{,}05^2} ln\left(\frac{4((2*452957)^{10}+1)}{0{,}05}\right) \approx 452957$$

Tenemos convergencia y por tanto se necesita un tamaño muestral $N = 452957$.

1.10. Ejercicio 10

Considere que le dan una muestra 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.*

Consideramos en primer lugar el principio de inducción ERM. Recordamos que el objetivo de este principio es escoger una hipótesis que minimice E_{in}. Es una hipótesis que suele dar buenos resultados pero tiene desventajas como que la existencia de ruido puede generar sobreajuste o que cuando tenemos $\frac{N}{d_{VC}} < 20$ no garantiza el aprendizaje (esto es, $E_{out} \approx 0$).

En este caso, podemos utilizar el principio SRM (*Structural Risk Minimation*). SRM fija la clase de funciones (complejidad) y trata de minimizar el error E_{out}. Esto puede solucionar el problema del sobreajuste al fijar la complejidad y evitar que la hipótesis se vea afectada por el ruido.