TRABAJO 2: Cuestiones de Teoría

Sergio Aguilera Ramirez

2 de 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.**

   Estas dos condiciones son:

   - Los datos deben ser muestras independientes e idénticamente distribuidas (iid) en una distribución con probabilidad P, ya que si esto no ocurriese el error en los datos test en las regiones donde no se ha aprendido pueden devolver errores mayores y dicho problema pasaría a un problema de aprendizaje sobre una región determinada de f.

   - Por otro lado, el término M perteneciente a la desigualdad de Hoeffding, mediante los resultados obtenidos al aplicarlo sobre el SRM define que la dimensión de Vapnik-Chervonenkis de la clase de hipótesis seleccionada debe ser finita, en caso contrario, solo podemos afirmar que la probabilidad de que la diferencia dentro y fuera de la muestra sea menor que δ sería menor o igual a ∞.

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.**

   Esta decisión no beneficiará a la empresa, ya que mediante el teorema de Non-Free-Lunch sabemos que para cada (A,H) existe una probabilidad P que falla, aunque dicha probabilidad P pueda ser aprendida mediante otro modelo. Además, todos los modelos homólogos sobre todas las funciones objetivo f.

   Debido a esto, si en un futuro la empresa tuviera que enfrentarse a un problema totalmente diferente a los que desarrolla en la actualidad y dicho algoritmo seleccionado no pudiera manejar los datos relacionados al problema, no tendrán la posibilidad de resolver dicho problema. Cada modelo debería ser aplicado en su distribución de clases de probabilidad P.

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

Entendemos como solución PAC, aquellas soluciones que no son realmente ciertas. Por ejemplo, tenemos un conjunto de 50 individuos, algunos son de Granada y otros de Cádiz, cogemos una muestra de 20 individuos y obtenemos el porcentaje de cada localidad, obtenemos que el 700 % son de Granada, pero cabe la posibilidad de que en esa muestra se coja todos los residentes de Granada y los 30 individuos restantes sean todos de Cádiz, no podríamos garantizar que con una probabilidad del 0.7 cogeremos un individuo del conjunto total que sea de Granada ya que no será verdad, ha estas soluciones se les denomina PAC, ya que existe cierta incertidumbre e impresión.

En estas soluciones obtenemos un $\boldsymbol{h}$ tal que $P\left[E_{in}(f) - E_{out}(f)\right] > \epsilon$

4. **Suponga un conjunto de Datos D de 25 ejemplos extraidos de una función desconocida $f : X \to Y$, donde $X = \mathbb{R}$ e $y = \{-1, +1\}$. Para aprender f usamos un conjunto simple de hipótesis $\mathcal{H} = \{\boldsymbol{h}_1, \boldsymbol{h}_2\}$ donde $\boldsymbol{h}_1$ es la función constante igual a +1 y $\boldsymbol{h}_2$ la función constante igual a -1. Consideremos 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**

Si, para demostrar esto realizaremos la siguiente deducción: Para comprobar que el algoritmo S produce una hipótesis mejor que C debemos demostrar que $P\left[E_{out}(S(D)) < E_{out}(C(D))\right]$, es decir, que el error obtenido por el algoritmo S sea menor que el error obtenido por C.

$$\begin{aligned} P\left[E_{out}(\boldsymbol{S}(\boldsymbol{D})) < \boldsymbol{E}_{\text{out}}(\boldsymbol{C}(\boldsymbol{D}))\right] &= \boldsymbol{P}\left[\boldsymbol{E}_{\text{out}}\left(\boldsymbol{h}_1\right) < \boldsymbol{E}_{\text{out}}\left(\boldsymbol{h}_2\right)\right] \\ &= \boldsymbol{P}\left[\boldsymbol{P}\left[\boldsymbol{f}(\boldsymbol{x}) \neq \boldsymbol{h}_1\right] < \boldsymbol{P}\left[\boldsymbol{f}(\boldsymbol{x}) \neq \boldsymbol{h}_2\right]\right] \\ &= \boldsymbol{P}[\boldsymbol{P}[\boldsymbol{f}(\boldsymbol{x}) = -\boldsymbol{1}] < \boldsymbol{P}[\boldsymbol{f}(\boldsymbol{x}) = +\boldsymbol{1}]] \\ &= \boldsymbol{P}[\boldsymbol{1} - \boldsymbol{p} < \boldsymbol{p}] \end{aligned}$$

Con la demostración anterior podemos concretar que para una p >0.5, la probabilidad de que el comportamiento de la hipótesis de S sea mejor que el comportamiento de la hipótesis de C es elevada, también puede influir el número de datos fuera de la muestra, ya que para un mayor número de datos fuera de la muestra la probabilidad de que S sea mejor aumenta.

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 hipótesis que produce C sea mejor que la hipótesis que produce S ?. Justificar la respuesta.**

   Sabemos que $\mathcal{D}$ solo contiene datos con etiquetas $y_n = +1$. Debido a que S realiza el aprendizaje sobre este conjunto de datos, podemos decir que si fuera de la muestra aparecen datos con etiquetas -1, la hipótesis generada por S, que no ha aprendido sobre etiquetas -1, obtendrá un error mayor, en cambio la hipótesis generada por C, puede que al ajustar de forma menos efectiva que S sobre los datos de muestra, fuera de la muestra obtenga un mejor resultado sobre datos con etiquetas +1 y -1.

   De esto concluimos que hay probabilidad de que la hipótesis producida por C obtenga un mejor ajuste que la hipótesis generada por S fuera de la muestra.

6. **Considere la cota para la probabilidad de la hipótesis solución $\boldsymbol{g}$ de un problema de aprendizaje, a partir de la desigualdad generalizada de Hoeffding para una clase finita de hipótesis, $\mathbb{P}\left[|\boldsymbol{E_{\text{out}}}(\boldsymbol{g}) - \boldsymbol{E_{\text{in}}}(\boldsymbol{g})| > \epsilon\right) < \boldsymbol{\delta}$**

   a) **¿Cuál es el algoritmo de aprendizaje que se usa para elegir $\boldsymbol{g}$?**

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

   c) **¿Depende $\boldsymbol{g}$ del algoritmo usado?**

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

   a) El algoritmo que consiga minimizar de forma más óptima el error en las clases de hipótesis $\mathcal{H}$.

   b) Si, pero al ser escogido de forma aleatoria, no podemos verificar que sea el que obtiene el mínimo error posible.

   c) Como viene especificado en las transparencias de teoría $\boldsymbol{g}$ depende de D por lo que el algoritmo usado no afecta a $\boldsymbol{g}$.

   d) Es una cota laxa.

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 tenemos varias hipótesis, el algoritmo de selección cogerá la hipótesis final basada en D, donde $P\left[|E_{in}(h) - E_{out}(h)| > \epsilon\right]$ es pequeño. Si $\mathcal{H}$ es mayor que 1 entonces la desigualdad de Hoeffding se aplicará para cada h perteneciente a $\mathcal{H}$, es decir se aplica para M términos. Esto podemos deducirlo de la siguiente forma:

$$\begin{aligned}\mathbb{P}\left[|E_{in}(g) - E_{out}(g)| > \epsilon\right] \leq \mathbb{P}\left[|E_{in}\left(h_1\right) - E_{out}\left(h_1\right)| > \epsilon\right. \\ \text{ó}\left.|E_{in}\left(h_2\right) - E_{out}\left(h_2\right)| > \epsilon\right. \\ \ldots \\ \text{ó}\left.|E_{in}\left(h_M\right) - E_{out}\left(h_M\right)| > \in\right] \\ \leq \sum_{n=1}^{N} \mathbb{P}\left[|E_{in}\left(h_n\right) - E_{out}\left(h_n\right)| > \in\right]\end{aligned}$$

Esta expresión quedaría: $\mathbb{P}\left[|E_{in}(g) - E_{out}(g)| > \in\right] \leq 2Me^{-2\epsilon^2 N}$

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 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 $\boldsymbol{m_{\mathcal{H}}(k) = 2^{k^*}}$**

   a) No sirve para probar que k^* es un punto de ruptura, pero puede usarse para confirmar que no es un punto de ruptura, ya que es necesario, para que sea un punto de ruptura, que $\mathcal{H}$ no pueda separar ningún conjunto de k^* puntos. Por lo que si $\mathcal{H}$ consigue separar algún conjunto entonces k^* no es un punto de ruptura.

   b) b) No sirve ya que como hemos dicho anteriormente esto solo sirve para verificar que no es un punto de ruptura.

   c) Esto no es útil ya que solo comprobamos que es punto de ruptura para ese conjunto, pero no para todos los conjuntos de k^* puntos.

d) Esto si nos sirve ya que como indica la propia definición, un punto de ruptura es aquel en el que $\mathcal{H}$ no es capaz de separar ningún conjunto de k^* puntos.

e) Esto no serviría ya que lo único que podríamos afirmar es que, si k=k^* confirmamos que k^* no es un punto de ruptura, y si $k>k^{**}$ no podríamos afirmas si es un punto de ruptura o no lo es.

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

Para el cálculo del tamaño muestral (N) utilizaremos la fórmula descrita en teoría: $N \geq 8 \ln \frac{4 * \frac{\left[(2*N)^{d_{vc}+1}\right]}{\delta}}{\epsilon^2}$

$$0{,}05 = \sqrt{\frac{8}{N} \ln \frac{4((2N)^{10}+1)}{0{,}05}} =>$$

$$0,05^2 = \sqrt{\frac{8}{N} \ln \frac{4((2N)^{10}+1)}{0{,}05}} =>$$

$$0,0025 = \frac{8}{N} \ln \frac{4\left((2N)^{10}+1\right)}{0{,}05} =>$$

$$0,0025N = 8 \ln \frac{2^{12}N^{10}+4}{0{,}05} =>$$

$$0,0025 = 8 \ln \left(2^{12}N^{10} + 4\right) - \ln 0,05 =>$$

$$N = \frac{8 \ln\left(2^{12}N^{10}+4\right) - \ln 0{,}05}{0{,}0025}$$

El siguiente programa en Python calcula el tamaño mínimo muestral, probando de forma iterativa diferentes tamaños de N.

```
# -*- coding: utf-8 -*-
"""
Created on Mon Apr 29 21:30:31 2019

@author: SERGIO
"""
import numpy as np

def calcula_N(x):
    res = 8/(0.05**2) * np.log(4*((pow(2*x,10)+1)/0.05))
    return res

N = 999999
n = calcula_N(N)

while np.fabs(n-N) > np.e**-6:
    N = n
    n = calcula_N(N)

print("Tamanio: ", round(N))
```

El resultado obtenido tras ejecutar este programa es un tamaño muestras de 452957.0

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.**

Si aplicamos el principio de inducción ERM, se obtiene un conjunto pesos a partir de la minimización del error, en función de la clasificación de cada dato con respecto a las etiquetas. Para dicha clasificación debemos comprobar si la etiqueta es correcta o no, etiquetamos los datos en función a una recta, asignamos las etiquetas -1 y +1 dependiendo si el punto se encuentra por encima o por debajo de la recta respectivamente. En cada clasificación de un dato modificamos el conjunto de pesos. Cuando tenemos el conjunto de pesos final, podemos obtener el hiperplano del conjunto de datos.

El principio de inducción ERM es ventajoso sobre grandes conjuntos de datos, en cambio, si el conjunto de entrenamiento es pequeño respecto a la dimensión, no podemos asegurar que mediante ERM se pueda aprender sobre dicho conjunto de datos.

SRM nos permite aproximar el error a un conjunto H y disminuir la dimensión VC. SRM permite movernos por diferentes clases de funciones delimitadas por su dimensión de Vapnik-chervonenkis, por medio de estas podemos encontrar la mejor función que obtenga el menor error.