Pontificia Universidad Católica de Chile
Escuela de Ingeniería
Departamento de Ciencia de la Computación

IIC2613 – Inteligencia Artificial (I/2020)

Interrogación/Tarea 3

Aspectos preliminares

El objetivo de esta evaluación es que se familiaricen con los aspectos prácticos y teóricos de los tópicos de aprendizaje de máquina vistos hasta ahora en el curso. En esta evaluación hay preguntas teóricas y de programación, que estarán marcadas con **(T)** o una **(P)** al lado del número que las identifica, para diferenciarlas claramente. Para facilitar la el desarrollo de ambos tipos de preguntas, todas las respuestas deberán ser realizadas en Jupyter Notebooks. De esta manera, todas las respuestas pueden incluir texto y código si así lo necesitan. Cada una de las preguntas de esta evaluación debe ser desarrollada en un archivo separado, con nombre `PX.ipynb`, donde `X` indica el número de la pregunta. Dentro de cada archivo, cada ítem debe estar claramente demarcado. La evaluación debe ser entregada a través de sus repositorios privados en GitHub. Para esto, deben crear la carpeta `T3`, y dentro de ella subir los archivos de cada pregunta. **La fecha de entrega es el jueves 25/06 hasta las 23:59**.

La evaluación esta diseñada para que deban buscar material y recursos que no están en el sitio del curso para contestar las preguntas. Tomando esto en consideración, como esta tarea tiene un fin pedagógico, la regla de integridad académica que deben cumplir es la siguiente:

> *Toda respuesta a cualquier pregunta debe ser escrita individualmente. Esto significa que al momento de editar la respuesta, no se debe estar mirando o usando un texto escrito por otra persona, un video hecho por otra persona, o cualquier material que haya sido desarrollado por otra persona. En otras palabras, lo que sea que se escriba debe provenir directamente de ustedes. Sin embargo, antes de comenzar a escribir cada respuesta pueden estudiar y comentar la pregunta con otros alumnos, leer libros, mirar videos, o realizar cualquier otra acción que los ayude a responder la pregunta. Esta regla aplica tanto a pregunta teóricas como prácticas.*

Finalmente, la entrega deberá incluir un archivo llamado INTEGRIDAD.txt, en donde afirman que han leído, entendido y respetado la regla de arriba. De no cumplirse esto, la tarea no será corregida y obtendrán la nota mínima.

1. Sesgo Inductivo

Cada algoritmo de aprendizaje posee un sesgo inductivo, que corresponde a los supuestos que realiza este sobre las soluciones al problema. Esto permite que un algoritmo de aprendizaje priorice una solución (o interpretación) sobre otra, independientemente de los datos observados. Así mismo, los sesgos inductivos pueden expresar suposiciones sobre el proceso de generación de datos o el espacio de soluciones.

a) (T) ¿Qué relación se podría establecer entre el sesgo inductivo de un algoritmo y el sesgo en la predicción de una variable? **(2.0 ptos.)**

b) (T) Identifique y explique los sesgos inductivos del k-NN y de los árboles de decisión basados en ganancia de información. ¿Qué relación existe entre el sesgo del k-NN y la clasificación realizada en un nodo hoja de un árbol? **(2.0 ptos.)**

c) (P) Implemente un árbol de decisión, donde el algoritmo de *split* de los nodos siempre elija el atributo con la ganancia de información más baja. Escoja un set de datos, y en base a este compare y comente sobre los árboles

y el rendimiento generados por este esquema y por el de un árbol de decisión tradicional. ¿Qué puede decir acerca del sesgo inductivo de este nuevo esquema? ¿Es mejor que el sesgo inductivo de un árbol de decisión tradicional?

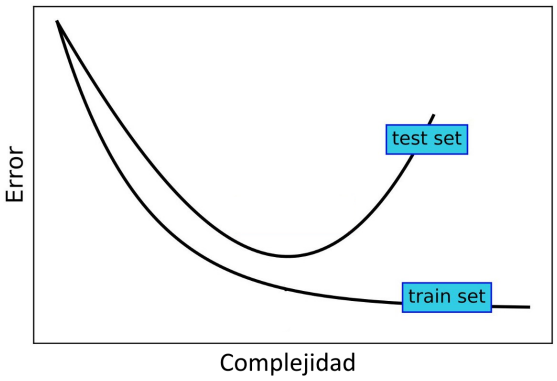
Para esta pregunta puede basarse en la implementación disponible en el sitio del curso, o puede utilizar otra. **(2.0 ptos.)**

2. Complejidad y Sobreentrenamiento

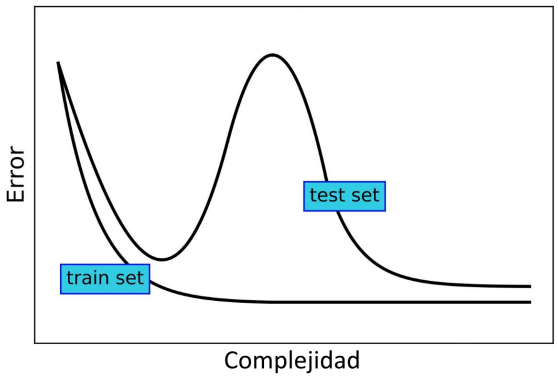
La complejidad de un modelo de aprendizaje tiene directa relación con el rendimiento que este puede obtener en el set de entrenamiento. Sin embargo, una complejidad excesiva en el modelo hace que generalmente su poder de generalización disminuya, debido al sobreentrenamiento. Este fenómeno se conoce como el *bias-variance tradeoff*.

a) (P) Implemente un ejemplo que capture el sobreentrenamiento en una regresión logística, a medida que se aumenta su complejidad mediante coeficiente polinomiales de mayor grado. Genere sets de datos no triviales para el análisis,*i.e.*, que no sean linealmente separables. Visualice el espacio de características, incluyendo las categorías de los ejemplos y comente sobre la evolución de la superficie de decisión al aumentar la complejidad. Grafique los rendimientos en los sets de entrenamiento y test a medida que aumenta la complejidad. **(3.0 ptos.)**

b) (T) Al graficar la complejidad de un modelo vs su rendimiento en un set de test, generalmente se obtiene un curva con forma parecida a la de una U, como el de la figura (a). Sin embargo, existen modelos para los cuales se ha observado empíricamente una curva con la forma descrita en la figura (b):



(a)



(b)

Esboce una explicación de lo que puede estar causando la segunda curva, asumiendo que no hay errores en los datos ni en el modelo. **(1.0 ptos.)**

Hint: Esta situación se sigue investigando ampliamente sin que haya consenso sobre su causa, por lo que acá se esperan sólo hipótesis (razonables) y no demostraciones.

c) (T) Como hemos visto, en una regresión (lineal o logística), una manera de cuantificar la complejidad del modelo es contando la cantidad de variables de entrada que son utilizadas para hacer la estimación. Basado en esto, una manera de controlar la complejidad de una regresión es mediante la incorporación en la función objetivo, de un término que penalice la cantidad de variables de entrada que son utilizadas por la solución actual para realizar la estimación. Proponga un mecanismo que permita realizar esto, manteniendo la convexidad de la función objetivo, y justifique su decisión mediante una (de)mostración. **(2.0 ptos.)**

Hint: Investigue sobre este tema antes de contestar la pregunta, ya que existe abundante literatura científica relacionada. Para la (de)mostración, puede referenciar los resultados de algún trabajo que haya encontrado en su revisión de la literatura.

d) (T) **Bonus**: ¿qué sesgo inductivo tiene el modelo del ítem anterior? ¿cómo se relaciona con los sesgos inductivos de la pregunta 1.b)? **(1.0 ptos. extra para esta pregunta, nota máxima 7.0)**

3. Ensambles de modelos

Los *random forests* corresponde a una clase de modelos conocidos como ensambles, que combinan múltiples modelos para mejorar las predicciones y disminuir el sobreentrenamiento. Este tipo de técnica ha tenido gran éxito en múltiples dominios.

a) (T) Proponga un esquema de ensamble para regresiones logísticas, distinto al usado para construir *random forests*, *i.e.*, aleatoriedad en los sets de datos y en las características de entrada. **(2.0 ptos.)** *Hint*: Si no se le ocurre nada, investigue sobre el concepto de *boosting*.

b) (P) Implemente ensambles de regresiones logísticas utilizando dos metodologías distintas. La primera basada en el algoritmo para construir *random forests* y la segunda basada en la estrategia propuesta en el ítem anterior de esta pregunta. Compare el rendimiento de estos ensambles y el de una regresión logística con características polinomiales, haciendo un análisis de sensibilidad basado en la complejidad de los modelos.

   Para entrenar los modelos, utilice el set de datos *"Financial Indicators of US stocks"*, disponible en el sitio del curso. Este conjunto de datos consiste en más de 200 indicadores financieros para distintas acciones. La tarea consiste en predecir si esas acciones son convenientes de comprar (en base a la variación de su precio), categoría que está dada por la columna `Class`. De los cinco archivos que conforman el set de datos, deben utilizar los correspondientes a los años entre 2014 y 2017 para entrenar, y el archivo correspondiente a 2018 para evaluar el rendimiento. Se recomienda fuertemente revisar más información sobre el set de datos en: `https://www.kaggle.com/cnic92/200-financial-indicators-of-us-stocks-20142018`

   (4.0 ptos.)

c) (T) **Bonus**: Repita ambos ítems anteriores, pero esta vez utilizando regresiones lineales. Para esta tarea, utilice el mismo set de datos, pero como objetivo use la columna `PRICE VAR [%]`. **(0.5 ptos. extra para la prueba, nota máxima 7.0)**