CIMAT Centro de Investigación en Matemáticas, A.C.

ANÁLISIS EXPERIMENTAL DE LA COMPLEJIDAD ESTADÍSTICA DE LA METODOLOGÍA DE INFORMATION BOTTLENECK

T E S I S

Que para obtener el grado de

Maestro en Ciencias

con Orientación en

Probabilidad y Estadística

Presenta

Lic. Judith Tavarez Rodríguez

Director de Tesis:

Dr. Mario Alberto Diaz Torres

Autorización de la versión final

Guanajuato, Gto., Agosto de 2019.

A mis personas favoritas

Agradecimientos

Agradezco a CONACYT el financiamiento obtenido mediante la asignación de una beca para mis estudios de maestría. A CIMAT por todo el apoyo académico y económico brindado desde el primer momento en que llegué a la institución. A mi asesor de tesis, Dr. Mario Diaz, por el tiempo dedicado a este trabajo y por ser un excelente guía. A mis sinodales Dr. Enrique Villa, Dr. Rogelio Ramos y Dr. Emilien Joly por sus comentarios y observaciones. A mi tutor académico Dr. Miguel Nakamura por sus valiosos consejos durante la maestría. A mi familia Fidel Tavarez, Ofelia Rodríguez y Monserrat Tavarez, por todo el apoyo moral que me brindaron. Y al M. C. Edgar Castañeda por su constante ayuda tanto en el plano académico como en el personal.

Resumen

En esta tesis presentamos un análisis empírico de las dificultades que conlleva aplicar la metodología de *Information Bottleneck*. Esta metodología permite analizar, desde una perspectiva de teoría de la información, la evolución de la información que contienen las variables que se manipulan en una técnica estadística o de aprendizaje máquina. Las variables en cuestión forman parte de un problema de predicción o clasificación, lo cual permite tener un marco común para distintas técnicas. En particular, se ha mostrado en artículos recientes que la metodología se puede aplicar a redes neuronales. Uno de los principales objetivos de los autores es generar conocimiento sobre cómo es que una red neuronal manipula la información en cada capa. Dichos trabajos han generado controversia debido a las dificultades en cuanto al uso de teoría de la información. La contribución que se tiene en este trabajo es ilustrar, con ejemplos relativamente sencillos, algunos de los límites fundamentales estadísticos subyacentes, los cuales no han sido analizados satisfactoriamente en la literatura y que constituyen una fuente en la controversia en torno al *Information Bottleneck*.

Palabras Clave

Information Bottleneck, Redes Neuronales, Análisis Empírico, Evolución de Información

Índice general

Agradecimientos v

Resumen vii

Introducción 1

1. Conceptos de Teoría de la Información 5
1.1. Entropía e Información Mutua de variables aleatorias discretas 5
1.2. Entropía e Información Mutua de variables aleatorias continuas 18
1.2.1. Información mutua y algoritmos deterministas 22

2. Machine Learning y Teoría de la Información 25
2.1. Modelo de aprendizaje supervisado 25
2.2. Desempeño y Generalización de Algoritmos 27
2.3. Generalización e Información Mutua 34
2.4. Redes Neuronales 36
2.4.1. *Stochastic Gradient Descent* 39

3. Metodología de Information Bottleneck 41
3.1. Curva de Information Bottleneck 41
3.2. Information Bottleneck y Redes Neuronales 42
3.3. Information Bottleneck y Estimación de Información Mutua 45
3.3.1. Estimación de información mutua mediante discretizaciones de las variables 45
3.3.2. Otros métodos de estimación de información mutua 53

Conclusiones 55

Índice de figuras

1. Plano de información en el cual se muestra la región correspondiente a posibles representaciones de X. En rojo se muestra la curva de Information Bottleneck. 2

1.1. Gráfica de la información mutua entre X^n y $T_{n,\gamma}$, $I(X^n; T_{n,\gamma})$ con respecto a $\gamma > 0$, donde $X_1, \ldots, X_N$ v.a.i.i.d. como $\mathcal{N}(0,1)$ y $T_{n,\gamma} = \frac{1}{n}\sum_{i=1}^{n} X_i + \sqrt{\gamma}Z$, $Z \sim \mathcal{N}(0,1)$ y es independiente de X_1, X_2, $X_3, \ldots$. 24

2.1. Gráfica de la pérdida esperada del algoritmo $\mathcal{A}_\gamma$ con respecto a γ y una cota inferior. 33

2.2. Gráfica del error de generalización del algoritmo $\mathcal{A}_\gamma$ con respecto a γ. . 35

2.3. Modelo perceptrón. Los datos de entrada son $x_1, \ldots, x_n$ y $x_{n+1} = 1$; los pesos de cada vértice son $w_1, \ldots, w_n$ y $w_{n+1} = b$; $\sigma : \mathbb{R} \to \mathbb{R}$ es función de activación. En la neurona se efectúa una suma ponderada y como salida se tiene la función de activación aplicada a dicha suma. 37

2.4. Red neuronal (*fully connected feedforward*), con una capa oculta. Su arquitectura consta de 3 capas con $3 - 4 - 1$ neuronas respectivamente. Los datos de entrada son $x = (x_1, x_2)$; las matrices de pesos $w_1 \in \mathbb{R}^{3\times 2}$ y $w_2 \in \mathbb{R}^{1\times 3}$; los bias $b_1 \in \mathbb{R}^3$ y $b_2 \in \mathbb{R}$. 38

3.1. Plano de información en el cual se muestra la región correspondiente a posibles representaciones de X. En rojo se muestra la curva de Information Bottleneck. 42

3.2. Resultados mostrados en [1]. Dinámica de aprendizaje de redes neuronales con distintas arquitecturas en el plano de información. Cada una de las trayectorias observadas corresponde a la dinámica promedio de cada capa de la red para 50 entrenamientos con inicializaciones y muestras distintas. 44

3.3. Gráfica de la información mutua entre X^Δ y Y^Δ (discretizaciones de las variables unidimensionales X y Y respectivamente) con respecto a Δ, así como también $\hat{I}(X^\Delta; Y^\Delta)$ que es una estimación de $I(X^\Delta; Y^\Delta)$ con 30 muestras de (X, Y), cada una de tamaño $N = 10$. 48

3.4. Gráfica de la información mutua entre X^Δ y Y^Δ con respecto a Δ, así como también $\hat{I}(X^\Delta; Y^\Delta)$ que es una estimación de $I(X^\Delta; Y^\Delta)$ con 30 muestras de las variables unidimensionales (X, Y), cada una de tamaño $N = 50$. 48
3.5. Gráfica de la información mutua entre X^Δ y Y^Δ con respecto a Δ, así como también $\hat{I}(X^\Delta; Y^\Delta)$ que es una estimación de $I(X^\Delta; Y^\Delta)$ con 30 muestras de las variables unidimensionales (X, Y), cada una de tamaño $N = 100$. 49
3.6. Gráfica de la información mutua entre X^Δ y Y^Δ con respecto a Δ, así como también $\hat{I}(X^\Delta; Y^\Delta)$ que es una estimación de $I(X^\Delta; Y^\Delta)$ con 30 muestras de las variables unidimensionales (X, Y), cada una de tamaño $N = 200$. 49
3.7. Gráfica de la información mutua entre X^Δ y Y^Δ (discretizaciones de las variables bidimensionales X y Y respectivamente) con respecto a Δ, así como también $\hat{I}(X^\Delta; Y^\Delta)$ que es una estimación de $I(X^\Delta; Y^\Delta)$ con 30 muestras de (X, Y), cada una de tamaño $N = 10$. 51
3.8. Gráfica de la información mutua entre X^Δ y Y^Δ con respecto a Δ, así como también $\hat{I}(X^\Delta; Y^\Delta)$ que es una estimación de $I(X^\Delta; Y^\Delta)$ con 30 muestras de las variables bidimensionales (X, Y), cada una de tamaño $N = 50$. 51
3.9. Gráfica de la información mutua entre X^Δ y Y^Δ con respecto a Δ, así como también $\hat{I}(X^\Delta; Y^\Delta)$ que es una estimación de $I(X^\Delta; Y^\Delta)$ con 30 muestras de las variables bidimensionales (X, Y), cada una de tamaño $N = 100$. 52
3.10. Gráfica de la información mutua entre X^Δ y Y^Δ con respecto a Δ, así como también $\hat{I}(X^\Delta; Y^\Delta)$ que es una estimación de $I(X^\Delta; Y^\Delta)$ con 30 muestras de las variables bidimensionales (X, Y), cada una de tamaño $N = 200$. 52

Introducción

Dentro del contexto de un problema de clasificación, supongamos que se tienen dos variables aleatorias X y Y correlacionadas, cuya relación está dada por la naturaleza. Por ejemplo, X puede ser una fotografía y Y puede ser una variable que toma el valor de 1 si en la fotografía aparece un gato o -1 si en la fotografía aparece un perro. Un problema común en estadística es el de querer estudiar la predicción que una variable puede dar sobre la otra, por lo que es de interés estimar o predecir a Y por medio de X. Es decir, se quiere obtener una variable $\hat{Y}$ por medio de la información que se posee sobre X y tratando de que sea lo más similar posible a Y.

Retomando el ejemplo de la fotografía, pudiese ser que cuando se le apliquen técnicas de procesamiento a la imagen, esta resulte tener miles de pixeles, lo cual generaría costos altos, así como dificultades en el manejo de los pixeles. Idealmente, se quisiera tener una representación de la fotografía original, que tenga el menor número de pixeles posibles (es decir, que tenga la máxima *compresión*). A su vez, se quisiera que la representación aún pueda dar información suficiente sobre si lo que contiene la fotografía original es un gato o un perro (es decir, que maximice el poder de *predicción*). En otros términos, dada la cadena de Markov $Y - X - T - \hat{Y}$, es de interés diseñar una representación T de X tal que, para $\mathcal{T}$ y $\mathcal{X}$, alfabetos de T y X,

$$|\mathcal{T}| \ll |\mathcal{X}| \quad \text{y} \quad \Pr\left(Y = \hat{Y}\right) \gg 0$$

(compresión y predicción respectivamente).

Por otro lado, la teoría de la información es el área encargada de cuantificar la información de variables aleatorias. Dos conceptos de gran importancia en teoría de la información son la entropía y la información mutua. La entropía cuantifica la incertidumbre en una variable aleatoria mientras que la información mutua, como su nombre lo dice, cuantifica la cantidad de información que tienen en común dos variables aleatorias.

En la intersección entre teoría de la información y estadística, podemos encontrar un resultado que relaciona la probabilidad de equivocarse en la estimación de la variable de interés Y, con la información mutua entre la variable predictora X y su representación T. Dicho resultado lleva por nombre *desigualdad de Fano*, y de esta desigualdad surge de manera natural escribir el problema de clasificación mencionado anteriormente en

términos de información mutua. Es decir, lo anterior se traduce en que la representación de X debe satisfacer

$$I(X;T) \ll 1 \quad \text{y} \quad I(Y;T) \gg 0$$

($|\mathcal{T}| \ll |\mathcal{X}|$ y $\Pr(Y = \hat{Y}) \gg 0$ respectivamente).

Existe una variedad amplia de técnicas estadísticas que dan solución al problema de comprimir la información manteniendo una buena predicción de la variable de interés. La metodología de *Information Bottleneck* estudia la dinámica de las técnicas estadísticas que se utilizan para resolver problemas de esta índole, en términos de información mutua.

Este concepto fue introducido por primera vez en 1999 por Naftali Tishby, Fernando C. Pereira y William Bialek. En su artículo [5] definen a la curva de *Information Bottleneck*, la cual se ilustra en la Figura 1 (curva roja). También se observa un plano, llamado el *plano de información*. A cada posible representación T de X le corresponde un único punto en el plano y al considerar a todos los puntos que corresponden a las representaciones factibles, queda una región como la delimitada por las dos curvas que se aprecian en la figura.

Al fijar un nivel de compresión a lo más de ϵ (es decir, si se quiere que $I(X;T) \leq \epsilon$) el óptimo en cuanto a precisión en la estimación (el valor más alto de $I(T;Y)$ que cumple con dicha compresión) es el $IB(\epsilon)$.

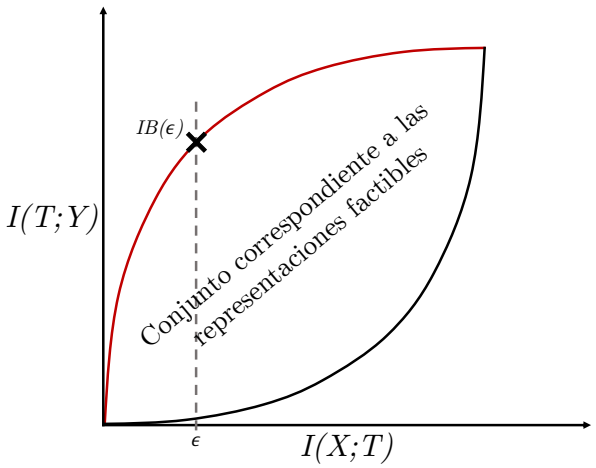

Figura 1: Plano de información en el cual se muestra la región correspondiente a posibles representaciones de X. En rojo se muestra la curva de Information Bottleneck.

En 2015, Tishby y Zaslavsky, mostraron que la dinámica de aprendizaje de una *red neuronal* se puede representar en un plano de información. La manera de hacerlo es asociando a cada una de las *capas* una representación, por lo que en este caso se tiene una cadena de representaciones.

Luego, en 2017, Tishby y Schwartz, publicaron resultados acerca de algunos experimentos realizados, en un intento por entender la dinámica de aprendizaje de una red neuronal. Aplicaron la metodología de IB a las capas de un modelo de red para representar la evolución de la información que contienen con respecto a la variable de entrada y a la variable de interés.

Conjeturaron varios resultados a partir de las simulaciones realizadas para distintas *arquitecturas* de redes neuronales. Por ejemplo, observaron que la dinámica de aprendizaje parece converger cerca de la curva de Information Bottleneck. Otra de sus conjeturas es que la red neuronal tiene una fase de ajuste en la precisión del aprendizaje, seguida de una fase de compresión en la información.

Sus resultados han recibido críticas por parte de algunos autores [3] así como también ha habido autores que apoyan parcialmente los resultados [2] y otros más han hecho análisis teóricos concernientes a la metodología [8]. Sin embargo, las opiniones al respecto continúan divididas.

La metodología de Information Bottleneck es relativamente nueva y aún no es clara la trascendencia que tendrá, por lo que es de interés analizar el procedimiento que han seguido los autores mencionados. Posteriormente se podría discernir sobre su uso en el análisis tanto de redes neuronales como de otras técnicas estadísticas.

Es por ello que en esta tesis se busca comprender e ilustrar mediante simulaciones con ejemplos sencillos, las dificultades que conlleva la aplicación de esta metodología.

En el Capítulo 1 se enunciarán algunas definiciones y resultados de teoría de la información que servirán como preámbulo. Además se da un primer acercamiento a situaciones desfavorables que surgen al hacer uso de dicha teoría.

El Capítulo 2 tiene un enfoque principal en aprendizaje máquina lo cual da mayor contexto al trabajo realizado en esta tesis. Cabe enfatizar que se utilizan resultados contemporáneos [6] en donde se unen la teoría de la información y el *machine learning*, teniendo consecuencias en la *generalización* de algoritmos.

Por último, en el Capítulo 3 se habla de la metodología de IB aplicada a redes neuronales. Se presentan los resultados de las simulaciones realizadas en esta tesis, las cuales permiten obtener un panorama amplio sobre la dificultad que conlleva la aplicación de la metodología de IB.

De este trabajo se pudo concluir que en algunos de los artículos existentes sobre IB no es evidente que se contemplen los límites fundamentales en la estimación de la información mutua. Esto afecta directamente a los resultados reflejados en el plano de información, lo cual da pauta a considerar que aún se requiere de trabajo en la metodología, previo a la aplicación a cualquier técnica estadística, no solo redes neuronales.

Capítulo 1

Conceptos de Teoría de la Información

En este capítulo se formaliza el hecho de utilizar teoría de la información para analizar la compresión y predicción de variables aleatorias que se manipulan en alguna técnica estadística, a través de resultados clásicos de teoría de la información.

El análisis que se realiza mediante la metodología de Information Bottleneck se puede fundamentar en la relación que existe entre información mutua y probabilidad de estimar correctamente. La desigualdad de Fano formaliza esta conexión y se demostrará más adelante en este capítulo. Para ello, se enunciarán las definiciones necesarias de teoría de la información y se demostrarán los resultados que sustentan dicha desigualdad, que es una base de la metodología de IB.

Además, se dará un ejemplo en el cual se ilustra uno de los problemas que surgen al analizar algoritmos deterministas con teoría de la información y una solución alternativa.

La mayor parte de este capítulo está basada en el libro de Cover & Thomas [9].

1.1. Entropía e Información Mutua de variables aleatorias discretas

Los conceptos de lo que se conoce como teoría de la información nacieron de la necesidad de formalizar matemáticamente la transmisión de mensajes en canales de comunicación. En la actualidad son utilizados en diversas áreas como estadística, computación, economía y física, entre otras. En estadística, el concepto de información mutua, es una medida de la dependencia entre dos variables aleatorias, es decir, la información en común entre ambas. Por lo anterior, es razonable considerar el hacer uso

de este concepto para cuantificar la información que se mantiene o se pierde durante la evolución de un algoritmo iterativo de interés en el que se realicen estimaciones de variables.

Para hacer uso de estas herramientas, primero se enunciarán definiciones para variables aleatorias discretas ya que, a pesar de que las que se utilizan en la práctica son continuas, es en base a estas que se podrá demostrar la desigualdad de Fano. Además, se considerarán discretizaciones de variables aleatorias continuas como parte de la metodología. Las definiciones y resultados para variables aleatorias continuas se verán en la siguiente sección.

Durante este capítulo, por conveniencia, se denotará a la función de masa de probabilidad $p_X(x)$ como $p(x)$. Por lo tanto, $p(x)$ y $p(y)$ harán alusión a dos variables aleatorias distintas con funciones de densidad $p_X(x)$ y $p_Y(y)$ respectivamente.

Se comenzará enunciando la definición de entropía de una variable aleatoria.

Definición 1.1. *La entropía de una variable aleatoria discreta X con función de masa de probabilidad $p(x)$ está definida por*

$$H(X) = -\sum_{x\in\mathcal{X}} p(x)\log p(x),$$

donde $\mathcal{X}$ es el alfabeto de X. Se usará la convención de que $0\log 0 = 0$, $a\log\frac{a}{0} = \infty$ si $a > 0$ y $0\log\frac{0}{0} = 0$, por continuidad.

A lo largo del capítulo, el logaritmo se tomará en base 2 para que de esta manera la entropía esté dada en bits, a menos que se especifique otra base.

La entropía de una variable aleatoria cuantifica la incertidumbre promedio que hay en la variable. Además es el número promedio de bits que se requieren para describir a una variable aleatoria ([9], Cap. 5). Es sencillo verificar que la entropía es una cantidad no negativa y una observación importante de hacer es que solo depende de la distribución de la variable aleatoria.

Ejemplo 1.2 (Entropía de variables binarias)**.** *Sea*

$$X = \begin{cases} 1 & \text{con probabilidad } p, \\ 0 & \text{con probabilidad } 1-p. \end{cases}$$

Entonces

$$H(X) = -p\log p - (1-p)\log(1-p).$$

A la cantidad anterior también se le suele denotar por $H(p)$ o $H((p,(1-p))$.

Ahora se presenta un resultado importante que se utiliza para justificar varias de las propiedades que satisface la entropía de una variable aleatoria.

Teorema 1.3 (Log-sum inequality)**.** *Sean a_i, b_i números no negativos para $i = 1, \ldots, n$. Entonces*

$$\sum_{i=1}^{n} a_i \log \frac{a_i}{b_i} \geq \left(\sum_{i=1}^{n} a_i \right) \log \frac{\sum_{i=1}^{n} a_i}{\sum_{i=1}^{n} b_i},$$

con igualdad si y solo si $a_i = cb_i$ para todo i, con $c \in \mathbb{R}$.

Demostración. Tomando en cuenta la convención dada en la Definición 1.1, sin pérdida de generalidad, se puede suponer que $a_i > 0$ y $b_i > 0$.

Recordar que la desigualdad de Jensen enuncia lo siguiente:

Si f es función real convexa, $t_1, t_2, \ldots, t_n \in \mathbb{R}$ y $\alpha_i > 0$, $i = 1, \ldots, n$ tales que $\sum_{i=1}^{n} \alpha_i = 1$, entonces

$$\sum_{i=1}^{n} \alpha_i f(t_i) \geq f\left(\sum_{i=1}^{n} \alpha_i t_i \right),$$

con igualdad si y solo si $t_1 = t_2 = \cdots = t_n$ o f es lineal. Si f es cóncava la desigualdad se cumple en sentido contrario.

Observar que la función $f(t) = t \log t$ es estrictamente convexa en $(0, \infty)$ ya que $f''(t) = \frac{1}{t} \log e > 0$. Entonces para $\alpha_i = \frac{b_i}{\sum_{j=1}^{n} b_j}$, $i = 1, \ldots, n$ y $t_i = \frac{a_i}{b_i}$, por la desigualdad de Jensen se tiene que,

$$\sum_{i=1}^{n} \frac{b_i}{\sum_{j=1}^{n} b_j} f\left(\frac{a_i}{b_i} \right) \geq f\left(\sum_{i=1}^{n} \frac{b_i}{\sum_{j=1}^{n} b_j} \frac{a_i}{b_i} \right)$$

si y solo si

$$\sum_{i=1}^{n} \frac{a_i}{\sum_{j=1}^{n} b_j} \log \frac{a_i}{b_i} \geq \sum_{i=1}^{n} \frac{a_i}{\sum_{j=1}^{n} b_j} \log \sum_{i=1}^{n} \frac{a_i}{\sum_{j=1}^{n} b_j},$$

si y solo si

$$\sum_{i=1}^{n} a_i \log \frac{a_i}{b_i} \geq \left(\sum_{i=1}^{n} a_i \right) \log \frac{\sum_{i=1}^{n} a_i}{\sum_{i=1}^{n} b_i}.$$

Y la igualdad se cumple si y solo si $t_1 = t_2 = \cdots = t_n$, es decir, $\frac{a_i}{b_i} = c$ para todo $i = 1, \ldots, n$.

□

Como consecuencia de la definición de entropía y del teorema anterior se deriva la siguiente propiedad.

Teorema 1.4. *Sea X variable aleatoria discreta y $\mathcal{X}$ su alfabeto. Entonces se cumple que*

$$0 \leq H(X) \leq \log|\mathcal{X}|,$$

con igualdades si y solo si X es constante y si y solo si X tiene distribución uniforme sobre $\mathcal{X}$, respectivamente.

Demostración. La primer desigualdad se sigue directamente de la definición ya que $0 \leq p(x) \leq 1$ para todo $x \in \mathcal{X}$, entonces $-\log p(x) > 0$, por lo que se concluye que $H(X) \geq 0$.

Observar que si $X \equiv x_0$ con x_0 constante,

$$H(X) = -\sum_{x\in\mathcal{X}} p(x)\log(x) = -p(x_0)\log p(x_0) = \log 1 = 0.$$

Además, si $H(X) = 0$, X es constante, ya que, como $p(x)$ es función de masa de probabilidad, existe $x_0 \in \mathcal{X}$ tal que $p(x_0) > 0$. Pero $-p(x)\log p(x) \geq 0$ para todo $x \in \mathcal{X}$, entonces, dado que $\sum_{x\in\mathcal{X}} p(x)\log(x) = 0$, se tiene que $p(x)\log p(x) = 0$ para todo x. Lo anterior implica que $\log p(x_0) = 0$, así $p(x_0) = 1$.

Por otro lado, de la definición de entropía y de la log-sum inequality (Teorema 1.3),

$$\begin{aligned}
\log|\mathcal{X}| - H(X) &= \log|\mathcal{X}| + \sum_{x\in\mathcal{X}} p(x)\log p(x) \\
&= \sum_{x\in\mathcal{X}} p(x)[\log|\mathcal{X}| + \log p(x)] \\
&= \sum_{x\in\mathcal{X}} p(x)\log\frac{p(x)}{(1/|\mathcal{X}|)} \\
&\geq \left(\sum_{x\in\mathcal{X}} p(x)\right)\log\frac{\sum_{x\in\mathcal{X}} p(x)}{\sum_{x\in\mathcal{X}}(1/|\mathcal{X}|)} \qquad (1.1) \\
&= \log 1 \\
&= 0.
\end{aligned}$$

Observar que la igualdad en (1.1) se cumple si y solo si $p(x) = c\frac{1}{\mathcal{X}}$, con $c \in \mathbb{R}$, para todo $x \in \mathcal{X}$ (Teorema 1.3). Es decir, si $p(x)$ es función de masa de la distribución uniforme sobre $\mathcal{X}$.

□

La definición de entropía se puede extender a más de una variable. Más aún, se puede definir la entropía condicional de una variable aleatoria dada otra.

Definición 1.5. *Sean X y Y variables aleatorias discretas con distribución conjunta $p(x, y)$ y sean $\mathcal{X}$ y $\mathcal{Y}$ los alfabetos de X y Y respectivamente.*

La entropía conjunta $H(X, Y)$ se define como

$$H(X, Y) = -\sum_{x\in\mathcal{X}}\sum_{y\in\mathcal{Y}} p(x, y) \log p(x, y).$$

La entropía condicional $H(Y \mid X)$ se define como

$$\begin{aligned} H(Y \mid X) &= \sum_{x\in\mathcal{X}} p(x) H(Y \mid X = x) \\ &= -\sum_{x\in\mathcal{X}}\sum_{y\in\mathcal{Y}} p(x, y) \log p(y \mid x). \end{aligned}$$

Las definiciones de entropía conjunta y entropía condicional para varias variables aleatorias son extensiones de la definición anterior. Dichas cantidades se ven relacionadas en el siguiente resultado, que dice que la entropía de una colección de variables aleatorias es la suma de entropías condicionales.

Teorema 1.6 (Regla de la cadena para entropía)**.** *Sean $X_1, X_2, \ldots, X_n$ variables aleatorias discretas con distribución conjunta $p(x_1, x_2, \ldots, x_n)$. Entonces*

$$H(X_1, X_2, \ldots, X_n) = \sum_{i=1}^{n} H(X_i \mid X_{i-1}, \ldots, X_1).$$

Demostración. Por definición de entropía condicional, probabilidad condicional y propiedades del logaritmo,

$$\begin{aligned} H(X_1, X_2, \ldots, X_n) &= -\sum_{x_1,x_2,\ldots,x_n} p(x_1, x_2, \ldots, x_n) \log p(x_1, x_2, \ldots, x_n) \\ &= -\sum_{x_1,x_2,\ldots,x_n} p(x_1, x_2, \ldots, x_n) \log \prod_{i=1}^{n} p(x_i \mid x_{i-1}, \ldots, x_1) \\ &= -\sum_{i=1}^{n} \sum_{x_1,x_2,\ldots,x_n} p(x_1, x_2, \ldots, x_n) \log p(x_i \mid x_{i-1}, \ldots, x_1) \\ &= -\sum_{i=1}^{n} \sum_{x_1,x_2,\ldots,x_i} p(x_1, x_2, \ldots, x_i) \log p(x_i \mid x_{i-1}, \ldots, x_1) \\ &= \sum_{i=1}^{n} H(X_i \mid X_{i-1}, \ldots, X_1) \end{aligned}$$

□

Del teorema anterior se tiene el siguiente corolario, el cual se utiliza fuertemente para demostrar la desigualdad de Fano.

Corolario 1.7. *Sean X, Y y Z variables aleatorias discretas, entonces*

$$H(X, Y \mid Z) = H(X \mid Z) + H(Y \mid X, Z).$$

Ya que la entropía es una medida de la incertidumbre de una variable aleatoria, es intuitivo que si se tiene cierto conocimiento sobre la variable en cuestión, su incertidumbre se vea reducida. Esto se formaliza en el siguiente teorema.

Teorema 1.8 (Condicionar reduce la entropía)**.** *Sean X y Y variables aleatorias discretas, entonces*

$$H(X \mid Y) \leq H(X)$$

con igualdad si y solo si X y Y son independientes.

Demostración. Por definición,

$$\begin{aligned} H(X) - H(X \mid Y) &= -\sum_{x \in \mathcal{X}} p(x) \log p(x) + \sum_{x \in \mathcal{X}, y \in \mathcal{Y}} p(x, y) \log p(x \mid y) \\ &= -\sum_{x \in \mathcal{X}, y \in \mathcal{Y}} p(x, y) \log p(x) + \sum_{x \in \mathcal{X}, y \in \mathcal{Y}} p(x, y) \log p(x \mid y) \\ &= \sum_{x \in \mathcal{X}, y \in \mathcal{Y}} p(x, y) \log \frac{p(x \mid y)}{p(x)} \\ &= \sum_{x \in \mathcal{X}, y \in \mathcal{Y}} p(x, y) \log \frac{p(x, y)}{p(x)p(y)}. \end{aligned} \tag{1.2}$$

Aplicando la log-sum inequality en (1.2),

$$\begin{aligned} H(X) - H(X \mid Y) &\geq \left(\sum_{x \in \mathcal{X}, y \in \mathcal{Y}} p(x, y) \right) \log \frac{\sum_{x \in \mathcal{X}, y \in \mathcal{Y}} p(x, y)}{\sum_{x \in \mathcal{X}, y \in \mathcal{Y}} p(x)p(y)} \\ &= \log 1 \\ &= 0. \end{aligned}$$

Y por la igualdad en Teorema 1.3, $H(X) - H(X \mid Y) = 0$ si y solo si $p(x, y) = p(x)p(y)$, es decir, si y solo si X y Y son independientes.

□

De acuerdo con el Teorema 1.8 y el comentario que le precede, al poseer conocimiento sobre otra variable relacionada con X, se puede reducir su incertidumbre, por lo que, si se conociera a la variable en sí, la incertidumbre sería nula, lo cual se enuncia en el siguiente teorema.

Teorema 1.9. *Sean X y Y variables aleatorias discretas.*

$$H(Y \mid X) = 0 \text{ si y solo si } Y = f(X) \text{ para alguna función } f.$$

Demostración. Supongamos que $H(Y \mid X) = 0$ y que para $x_0 \in \mathcal{X}$ con $p(x_0) > 0$ existen $y_1, y_2 \in \mathcal{Y}$ distintos tales que $p(x_0, y_1) > 0$ y $p(x_0, y_2) > 0$. Entonces

$$p(x_0) \geq p(x_0, y_1) + p(x_0, y_2) > 0,$$

por lo que

$$1 \geq p(y_1 \mid x_0) + p(y_2 \mid x_0) > 0.$$

De esta manera se tiene que

$$0 < p(y_1 \mid x_0),\, p(y_2 \mid x_0) < 1. \tag{1.3}$$

Por otro lado,

$$\begin{aligned} 0 = H(Y \mid X) &= -\sum_{x \in \mathcal{X}} \sum_{y \in \mathcal{Y}} p(x, y) \log p(y \mid x) \\ &= -\sum_{x \in \mathcal{X}} \sum_{y \in \mathcal{Y}} p(x) p(y \mid x) \log p(y \mid x). \end{aligned}$$

Como todos los términos de la suma son no negativos,

$$\begin{aligned} H(Y \mid X) &= -\sum_{x \in \mathcal{X}} \sum_{y \in \mathcal{Y}} p(x) p(y \mid x) \log p(y \mid x) \\ &\geq p(x_0) \left(-p(y_1 \mid x_0) \log p(y_1 \mid x_0) - p(y_2 \mid x_0) \log p(y_2 \mid x_0)\right) \\ &> 0 \quad \text{por (1.3)} \end{aligned}$$

lo cual es una contradicción. Por lo tanto, existe un único $y \in \mathcal{Y}$ tal que si $p(x) > 0$, $p(x, y) > 0$.

De esta manera se tiene que existe una función f tal que $Y = f(X)$.

Recíprocamente, si suponemos que $Y = f(X)$ para alguna función f, existe un único $y \in \mathcal{Y}$ tal que

$$p(x, y) = p(x, f(x)) = p(x),$$

para $x \in \mathcal{X}$ que cumple que $p(x) > 0$. Por lo que

$$p(y \mid x) = 1$$

y se sigue que

$$H(Y \mid X) = -\sum_{x \in \mathcal{X}} \sum_{y \in \mathcal{Y}} p(x) p(y \mid x) \log p(y \mid x) = 0.$$

□

Resumiendo lo anterior, al tener conocimiento sobre una variable aleatoria, se reduce su entropía. Pero esto es verdadero solamente EN PROMEDIO, ya que $H(X \mid Y = y)$ podría ser mayor o igual o menor que $H(X)$.
Sin embargo,

$$H(X \mid Y) \leq H(X),$$

donde $H(X \mid Y) = \sum_{y \in \mathcal{Y}} p(y) H(X \mid Y = y)$. Por ejemplo, en [9] se menciona que para un caso en una corte, presentar cierta evidencia específica puede aumentar la incertidumbre en el caso, sin embargo, toda la evidencia en promedio reduce la incertidumbre y hace posible resolverlo. Esta observación es muy importante, sobre todo en los capítulos donde se considera la metodología de IB. En ellos se verá que el error de tomar en cuenta solo una muestra de las variables para dar conclusiones en general, afecta a las mismas. Así pues, es fundamental recordar que la entropía y conceptos relacionados (como el que se verá a continuación), toman consideraciones en promedio. Lo anterior ha sido de relevancia para las conclusiones a las que se llega en esta tesis.

Ahora introducimos una medida que cuantifica la reducción en la incertidumbre de una variable aleatoria al tener conocimiento sobre otra variable, es decir, cuantifica la reducción en la entropía al tener información previa. A esta medida se le llama información mutua y mide la cantidad de información que poseen en común dos variables aleatorias. Además es una medida de la dependencia entre ellas.

Definición 1.10. *Sean X y Y variables aleatorias con función de masa de probabilidad conjunta $p(x, y)$ y funciones de masa de probabilidad marginales $p(x)$ y $p(y)$. La información mutua $I(X;Y)$ se define como*

$$I(X;Y) = H(X) - H(X \mid Y).$$

Además, se define a la información mutua condicional de las variables aleatorias X y Y dada Z como

$$I(X;Y \mid Z) = H(X \mid Z) - H(X \mid Y, Z).$$

Observemos que en la igualdad (1.2) se demuestra que la definición de información mutua se puede reescribir de tal manera que se den las siguientes relaciones con la entropía.

$$\begin{aligned} I(X;Y) &= \sum_{x \in \mathcal{X}} \sum_{y \in \mathcal{Y}} p(x,y) \log \frac{p(x,y)}{p(x)p(y)} \\ &= H(Y) - H(Y|X) \\ &= H(X) + H(Y) - H(X,Y). \end{aligned}$$

Además, la información mutua es una cantidad no negativa, lo cual se enuncia en el siguiente corolario.

Corolario 1.11. *Sean X y Y variables aleatorias discretas, entonces*

$$I(X;Y) \geq 0$$

con igualdad si y solo si X y Y son independientes.

Demostración. Del Teorema 1.8 y la definición de información mutua,

$$0 \leq H(X) - H(X \mid Y) = I(X;Y).$$

□

Un resultado similar se satisface para la información mutua condicional y se enuncia a continuación.

Teorema 1.12. *Sean X, Y y Z variables aleatorias discretas, entonces*

$$I(X;Y \mid Z) \geq 0$$

con igualdad si y solo si X y Y son condicionalmente independientes dado Z.

Demostración. Por definición,

$$\begin{aligned} I(X;Y \mid Z) &= H(X \mid Z) - H(X \mid Y,Z) \\ &= -\sum_{x\in\mathcal{X},z\in\mathcal{Z}} p(x,z)\log p(x \mid z) + \sum_{x\in\mathcal{X},y\in\mathcal{Y},z\in\mathcal{Z}} p(x,y,z)\log p(x \mid y,z) \\ &= -\sum_{x\in\mathcal{X},y\in\mathcal{Y},z\in\mathcal{Z}} p(x,y,z)\log p(x \mid z) + \sum_{x\in\mathcal{X},y\in\mathcal{Y},z\in\mathcal{Z}} p(x,y,z)\log p(x \mid y,z) \\ &= \sum_{x\in\mathcal{X},y\in\mathcal{Y},z\in\mathcal{Z}} p(x,y,z)\log \frac{p(x \mid y,z)}{p(x \mid z)} \\ &= \sum_{x\in\mathcal{X},y\in\mathcal{Y},z\in\mathcal{Z}} p(x,y,z)\log \frac{p(x,y,z)}{p(x \mid z)p(y,z)}. \end{aligned} \tag{1.4}$$

Aplicando la log-sum inequality,

$$\begin{aligned} I(X;Y \mid Z) &\geq \left(\sum_{x\in\mathcal{X},y\in\mathcal{Y},z\in\mathcal{Z}} p(x,y,z)\right)\log \frac{\sum_{x\in\mathcal{X},y\in\mathcal{Y},z\in\mathcal{Z}} p(x,y,z)}{\sum_{x\in\mathcal{X},y\in\mathcal{Y},z\in\mathcal{Z}} p(x \mid z)p(y,z)} \\ &= (1)\log \frac{1}{\sum_{y\in\mathcal{Y},z\in\mathcal{Z}} p(y,z)} \\ &= \log 1 \\ &= 0. \end{aligned}$$

Además de (1.4) se tiene que

$$I(X;Y \mid Z) = \sum_{x\in\mathcal{X},y\in\mathcal{Y},z\in\mathcal{Z}} p(x,y,z)\log \frac{p(x,y \mid z)}{p(x \mid z)p(y \mid z)}$$

por lo que, de la condición de igualdad en el Teorema 1.3, $I(X;Y \mid Z) = 0$ si y solo si $p(x, y \mid z) = p(x \mid z)p(y \mid z)$. Es decir, si y solo si X y Y son condicionalmente independientes dado Z.

□

Al igual que con la entropía, la definición de información mutua se puede extender para más variables y también cumple resultados como la regla de la cadena.

Teorema 1.13 (Regla de la cadena para información mutua)**.** *Sean $X_1, X_2, \ldots, X_n, Y$ variables aleatorias discretas, entonces*

$$I(X_1, X_2, \ldots, X_n; Y) = \sum_{i=1}^{n} I(X_i; Y \mid X_{i-1}, \ldots, X_1).$$

Demostración. Por la definición de información mutua, la regla de la cadena para entropía y la definición de información mutua condicional se tiene que

$$\begin{aligned} I(X_1, X_2, \ldots, X_n; Y) &= H(X_1, X_2, \ldots, X_n) - H(X_1, X_2, \ldots, X_n \mid Y) \\ &= \sum_{i=1}^{n} H(X_i \mid X_{i-1}, \ldots, X_1) - \sum_{i=1}^{n} H(X_i \mid X_{i-1}, \ldots, X_1, Y) \\ &= \sum_{i=1}^{n} I(X_i; Y \mid X_1, \ldots, X_{i-1}). \end{aligned}$$

□

Para comenzar a relacionar los conceptos enunciados hasta el momento con el problema de procesamiento de información, se presenta el siguiente teorema. Este dice que, sin importar las transformaciones que se realicen a los datos, por ingeniosas que sean, la información que se puede obtener de ellas no puede ser mayor a la información que dan los datos por sí mismos. El resultado se conoce como *data processing inequality* y hace uso de un tipo de procesos con variables aleatorias, de los cuales se enuncia primeramente su definición.

Definición 1.14. *Se dice que las variables aleatorias X, Y y Z forman una cadena de Markov en ese orden, lo cual se denota por $X - Y - Z$ si X y Z son condicionalmente independientes dado Y. Específicamente, para todo x, y, z, la distribución conjunta se puede escribir como*

$$p(x, y, z) = p(x)p(y \mid x)p(z \mid y).$$

Teorema 1.15 (Data-Processing Inequality)**.** *Si $X - Y - Z$ forman una cadena de Markov, entonces*

$$I(X;Y) \geq I(X;Z) \quad y \quad I(Y;Z) \geq I(X;Z).$$

Demostración. Utilizando la regla de la cadena para información mutua, se cumplen las siguientes igualdades:

$$I(X;Y,Z) = I(X;Z) + I(X;Y \mid Z),$$
$$I(X;Y,Z) = I(X;Y) + I(X;Z \mid Y).$$

Como X, Y y Z forman una cadena de Markov, X y Z son condicionalmente independientes dado Y. Entonces por el Teorema 1.12 se tiene que $I(X;Z \mid Y) = 0$. Además $I(X;Y \mid Z) \geq 0$, por lo que se concluye que

$$I(X;Y) \geq I(X;Z).$$

La otra desigualdad se demuestra de manera similar.

□

Como se mencionó al inicio de este capítulo, para entablar teóricamente la conexión entre información mutua y probabilidad de estimación correcta, se presenta la desigualdad de Fano. Con esto queda justificado el hecho de utilizar información mutua para analizar técnicas estadísticas que dan solución a problemas de estimación.

Teorema 1.16 (Desigualdad de Fano)**.** *Sea $Y - X - \hat{Y}$ una cadena de Markov en donde $\hat{Y}$ es un estimador de Y y sea $P_e = Pr(Y \neq \hat{Y})$. Entonces*

$$-\log(1-P_e) \leq H(Y) - I(X;Y) \leq -P_e \log P_e - (1-P_e)\log(1-P_e) + P_e \log|\mathcal{Y}|.$$

Demostración. Se procede a demostrar que

$$H(Y \mid \hat{Y}) \leq -P_e \log P_e - (1-P_e)\log(1-P_e) + P_e \log|\mathcal{Y}|$$

para después utilizar *data-processing inequality* y así obtener el resultado deseado.

Sea E la variable aleatoria dada por

$$E = \begin{cases} 1 & \text{si } \hat{Y} \neq Y, \\ 0 & \text{si } \hat{Y} = Y. \end{cases}$$

Utilizando la regla de la cadena para entropía (Corolario 1.7), se tiene lo siguiente

$$H(E,Y \mid \hat{Y}) = H(Y \mid \hat{Y}) + H(E \mid Y,\hat{Y}), \tag{1.5}$$
$$H(E,Y \mid \hat{Y}) = H(E \mid \hat{Y}) + H(Y \mid E,\hat{Y}). \tag{1.6}$$

En (1.5), se puede observar que, dado que E depende de Y y $\hat{Y}$, $H(E \mid Y,\hat{Y}) = 0$ (Teorema 1.9), por lo que

$$H(E,Y \mid \hat{Y}) = H(Y \mid \hat{Y}).$$

Y por (1.6) se sigue que

$$H(Y \mid \hat{Y}) = H(E \mid \hat{Y}) + H(Y \mid E, \hat{Y}). \tag{1.7}$$

Por el hecho de que condicionar reduce la entropía (Teorema 1.8) y por definición de entropía se tiene que

$$H(E \mid \hat{Y}) \leq H(E) = -P_e \log P_e - (1 - P_e) \log(1 - P_e). \tag{1.8}$$

Por otro lado

$$H(Y \mid E, \hat{Y}) = \Pr(E = 0) H(Y \mid \hat{Y},\, E = 0) + \Pr(E = 1) H(Y \mid \hat{Y},\, E = 1).$$

Como $E = 0$ implica que $Y = \hat{Y}$,

$$H(Y \mid \hat{Y},\, E = 0) = 0.$$

Más aún, si $E = 1$ entonces $Y \neq \hat{Y}$ y así, se puede acotar la entropía condicional por el logaritmo del número de valores posibles que puede tomar la variable (Teorema 1.4), por lo que

$$H(Y \mid \hat{Y},\, E = 1) \leq \log|\mathcal{Y}|.$$

De esto, de (1.7) y de (1.8) se sigue que

$$H(Y \mid \hat{Y}) \leq -P_e \log P_e - (1 - P_e) \log(1 - P_e) + P_e \log|\mathcal{Y}|.$$

Por *data-processing inequality*, $I(Y; \hat{Y}) \leq I(X; Y)$, entonces por definición se tiene que $H(Y \mid X) \leq H(Y \mid \hat{Y})$. Por lo tanto se concluye que

$$H(Y) - I(X; Y) \leq -P_e \log P_e - (1 - P_e) \log(1 - P_e) + P_e \log|\mathcal{Y}|.$$

Para la desigualdad[1] $-\log(1 - P_e) \leq H(Y) - I(X; Y)$, observemos que, por definición de entropía condicional,

$$\begin{aligned} 2^{-H(X|Y)} &= 2^{\sum_{x,y} p(x,y) \log p(y|x)} \\ &\leq 2^{\sum_{x,y} p(x,y) \log(\max_y p(y|x))}. \end{aligned}$$

Aplicando la desigualdad de Jensen para funciones cóncavas se tiene que

$$\begin{aligned} 2^{\sum_{x,y} p(x,y) \log(\max_y p(y|x))} &\leq 2^{\log\left(\sum_{x,y} p(x,y) \max_y p(y|x)\right)} \\ &= \sum_x \max_y p(y \mid x) \sum_y p(x, y) \end{aligned}$$

[1] Una versión extendida de esta desigualdad se puede encontrar en [7].

$$\begin{aligned}
&= \sum_x \max_y p(y \mid x)p(x) \\
&= \sum_x \max_y p(x,y) \\
&= 1 - P_e.
\end{aligned}$$

Así,

$$2^{-H(X|Y)} \leq 1 - P_e$$

y por lo tanto

$$-\log(1 - P_e) \leq H(Y) - I(X;Y).$$

□

A pesar de las nuevas herramientas que relacionan teoría de la información con otras áreas como *machine learning*, la desigualdad de Fano es un resultado que data de hace varios años siendo uno de los primeros en relacionar información mutua con probabilidad de estimación correcta. Provee de cotas para el error de estimación de una variable aleatoria en términos de información mutua con lo cual se podría buscar un balance entre cantidad de información y calidad en la estimación. Además, observemos que para la cadena de Markov $Y - X - \hat{Y}$, se satisface la siguiente versión de la desigualdad de Fano:

$$P_e \geq \frac{H(Y \mid X) - 1}{\log|\mathcal{Y}|} \tag{1.9}$$

debido a que si $E = \begin{cases} 1 & \text{si } \hat{Y} \neq Y \\ 0 & \text{si } \hat{Y} = Y \end{cases}$, por Teoremas 1.16 y 1.4,

$$\begin{aligned}
H(Y \mid Z) &\leq -\left(P_e \log P_e + (1 - P_e)\log(1 - P_e)\right) + P_e \log \mid \mathcal{Y} \mid \\
&= H(E) + P_e \log \mid \mathcal{Y} \mid \\
&\leq \log 2 + P_e \log \mid \mathcal{Y} \mid \\
&= 1 + P_e \log \mid \mathcal{Y} \mid .
\end{aligned}$$

Y al considerar a la cadena de Markov $Y - X - Z - \hat{Y}$, por 1.9 y *data processing inequality*,

$$P_e \geq \frac{H(Y \mid Z) - 1}{\log|\mathcal{Y}|} \geq \frac{H(Y \mid X) - 1}{\log|\mathcal{Y}|}.$$

Esto es, debido a transformaciones en los datos y a la pérdida de información que se obtiene de ello, se puede encontrar una cota inferior más grande para el error en la estimación.

1.2. Entropía e Información Mutua de variables aleatorias continuas

En lo siguiente, se darán las definiciones de entropía e información mutua para variables aleatorias continuas, ya que son las variables que se emplean en esta investigación.

Definición 1.17. *Se define la entropía diferencial $h(X)$ de una variable aleatoria continua X con densidad $f(x)$ como*

$$h(X) = -\int_S f(x) \log f(x)\, dx,$$

donde S es el soporte de X.

La entropía diferencial puede ser negativa [9], depende solo de la función de densidad de probabilidad de la variable y cabe mencionar que la integral puede no existir. Al igual que la entropía para variables aleatorias discretas, el resultado está dado en bits.

Ejemplo 1.18 (Distribución Normal)**.** Sea $X \sim N(0, \sigma^2)$, su densidad está dada por

$$\phi(x) = \frac{1}{\sqrt{2\pi\sigma^2}}\, e^{\frac{-x^2}{2\sigma^2}}.$$

Al calcular la entropía diferencial obtenemos que

$$\begin{aligned}
h(X) &= -\int \phi(x) \log \phi(x)\, dx \\
&= -\frac{1}{\ln 2} \int \phi(x) \ln \phi(x)\, dx \\
&= -\frac{1}{\ln 2} \int \phi(x) \left[-\frac{x^2}{2\sigma^2} - \ln \sqrt{2\pi\sigma^2} \right] dx \\
&= \frac{1}{\ln 2} \left(\frac{\mathbb{E}X^2}{2\sigma^2} + \frac{1}{2} \ln(2\pi\sigma^2) \right) \\
&= \frac{1}{\ln 2} \left(\frac{1}{2} + \frac{1}{2} \ln(2\pi\sigma^2) \right) \\
&= \frac{1}{\ln 2} \left(\frac{1}{2} \ln(2\pi e\sigma^2) \right) \\
&= \frac{1}{2} \log(2\pi e\sigma^2) \quad \text{bits.}
\end{aligned}$$

Por lo tanto,

$$h(X) = \frac{1}{2} \log 2\pi e\sigma^2 \quad \text{bits.}$$

También se pueden calcular la entropía diferencial de un conjunto finito de variables aleatorias que tengan densidad conjunta y la entropía condicional de una variable dada otra.

Definición 1.19. *La entropía diferencial de un conjunto de variables aleatorias* $X_1, X_2, \ldots, X_n$ *con densidad* $f(x_1, x_2, \ldots, x_n)$ *se define como*

$$h(X_1, X_2, \ldots, X_n) = -\int f(x^n) \log f(x^n)\, dx^n$$

donde $x^n = (x_1, x_2, \ldots, x_n)$.

Definición 1.20. *Si* X *y* Y *tienen densidad conjunta* $f(x, y)$*, se puede definir la entropía condicional* $h(X|Y)$ *como*

$$h(X|Y) = -\int f(x, y) \log f(x|y)\, dx\, dy.$$

Como en general $f(x|y) = f(x, y)/f(y)$, se puede escribir

$$h(X|Y) = h(X, Y) - h(Y).$$

El siguiente resultado será empleado más adelante en este capítulo.

Proposición 1.21 (Distribución Normal Multivariada)**.** *Sean* $X_1, X_2, \ldots, X_n$ *variables aleatorias con distribución Normal multivariada con media* $\mu \in \mathbb{R}^n$ *y matriz de covarianza* $\Sigma \in \mathbb{R}^{n\times n}$*. Entonces*

$$h(X_1, X_2, \ldots, X_n) = \frac{1}{2} \log\left((2\pi e)^n |\Sigma|\right).$$

Demostración. La densidad de $X_1, X_2, \ldots, X_n$ está dada por

$$f(x) = \frac{1}{\left(\sqrt{2\pi}\right)^n |\Sigma|^{1/2}}\, e^{-\frac{1}{2}(x-\mu)^t |\Sigma|^{-1}(x-\mu)},$$

por lo que de la Definición 1.19

$$\begin{aligned}
h(f) &= -\frac{1}{\ln 2} \int f(x) \left[-\frac{1}{2}(x-\mu)^t \Sigma^{-1}(x-\mu) - \ln\left((\sqrt{2\pi})^n |\Sigma|^{1/2}\right)\right] dx \\
&= \frac{1}{2\ln 2}\, \mathbb{E}\left[\sum_{i,j} (x_i - \mu_i)(\Sigma^{-1})_{ij}(x_j - \mu_j)\right] + \frac{1}{2\ln 2} \ln\left((2\pi)^n |\Sigma|\right) \\
&= \frac{1}{2\ln 2} \sum_{i,j} \mathbb{E}\left[(x_j - \mu_j)(x_i - \mu_i)\right](\Sigma^{-1})_{ij} + \frac{1}{2\ln 2} \ln\left((2\pi)^n |\Sigma|\right) \\
&= \frac{1}{2\ln 2} \sum_{j} \sum_{i} \Sigma_{ji}(\Sigma^{-1})_{ij} + \frac{1}{2\ln 2} \ln\left((2\pi)^n |\Sigma|\right) \\
&= \frac{1}{2\ln 2} \sum_{j} (\Sigma\Sigma^{-1})_{jj} + \frac{1}{2\ln 2} \ln\left((2\pi)^n |\Sigma|\right)
\end{aligned}$$

$$
\begin{aligned}
&= \frac{1}{2\ln 2}\sum_j I_{jj} + \frac{1}{2\ln 2}\ln\left((2\pi)^n|\Sigma|\right) \\
&= \frac{n}{2\ln 2} + \frac{1}{2\ln 2}\ln\left((2\pi)^n|\Sigma|\right) \\
&= \frac{1}{2\ln 2}\ln\left((2\pi e)^n|\Sigma|\right) \\
&= \frac{1}{2}\log\left((2\pi e)^n|\Sigma|\right) \quad \text{bits.}
\end{aligned}
$$

□

Una vez enunciadas las definiciones anteriores, se puede definir la información mutua entre dos o más variables aleatorias continuas.

Definición 1.22 (Información mutua)**.** *Sean X y Y variables aleatorias con densidad que además tienen densidad conjunta $f(x,y)$. Su información mutua se define como*

$$I(X;Y) = \int f(x,y)\log\frac{f(x,y)}{f(x)f(y)}\,dx\,dy.$$

De la definición de información mutua, se observa que también se puede escribir de las siguientes formas

$$
\begin{aligned}
I(X;Y) &= h(X) - h(X|Y) \\
&= h(Y) - h(Y|X) \\
&= h(X) + h(Y) - h(X,Y).
\end{aligned} \tag{1.10}
$$

Además, existe una relación entre entropía diferencial de una variable aleatoria y entropía de una versión discretizada de la misma. Con esta relación se puede aproximar a la información mutua de variables aleatorias continuas mediante la información mutua entre variables aleatorias discretas.

Sea X una variable aleatoria continua con densidad $f(x)$. Se denota por X^Δ a la variable aleatoria discreta definida por

$$X^\Delta = i \quad \text{si} \quad i\Delta \le X < (i+1)\Delta.$$

Teorema 1.23. *Si la densidad $f(x)$ de una variable aleatoria continua X es Riemann integrable, entonces*

$$\lim_{\Delta\to 0^+}(H(X^\Delta) + \log\Delta) = h(X).$$

Demostración. Como $f(x)$ es continua, existe $x_i \in [i\Delta,\,(i+1)\Delta)$ tal que

$$f(x_i)\Delta = \int_{i\Delta}^{(i+1)\Delta} f(x)dx,$$

por Teorema del valor medio. De aquí que

$$\sum_{-\infty}^{\infty} f(x_i)\Delta = \int_{-\infty}^{\infty} f(x)dx = 1.$$

Debido a que $X^\Delta = i$ si $i\Delta \leq X < (i+1)\Delta$, la probabilidad de que $X^\Delta = i$ es

$$p_i = \int_{i\Delta}^{(i+1)\Delta} f(x)dx = f(x_i)\Delta.$$

Luego,

$$\begin{aligned} H(X^\Delta) &= -\sum_{-\infty}^{\infty} p_i \log p_i \\ &= -\sum_{-\infty}^{\infty} f(x_i)\Delta \log\left(f(x_i)\Delta\right) \\ &= -\sum_{-\infty}^{\infty} f(x_i)\Delta \log f(x_i) - \sum_{-\infty}^{\infty} f(x_i)\Delta \log \Delta \\ &= -\sum_{-\infty}^{\infty} f(x_i)\Delta \log f(x_i) - \log \Delta \end{aligned}$$

Por lo que

$$H(X^\Delta) + \log \Delta = -\sum_{-\infty}^{\infty} f(x_i)\Delta \log f(x_i).$$

Al tomar el límite cuando $\Delta \to 0^+$, suponiendo que $f(x)\log f(x)$ es Riemann integrable, se tiene que

$$\lim_{\Delta \to 0^+} (H(X^\Delta) + \log \Delta) = -\int f(x)\log f(x) = h(X).$$

□

Proposición 1.24. *Sean X y Y variables aleatorias con densidad conjunta $f(x,y)$ Riemann integrable. Entonces la información mutua entre las dos variables aleatorias es el límite de la información mutua entre las variables discretizadas correspondientes. Es decir,*

$$\lim_{\Delta \to 0} I(X^\Delta; Y^\Delta) = I(X;Y).$$

Demostración. Observar que

$$\begin{aligned} I(X^\Delta; Y^\Delta) &= H(X^\Delta) - H(X^\Delta \mid Y^\Delta) \\ &= H(X^\Delta) + \log\Delta - (H(X^\Delta \mid Y^\Delta) + \log\Delta). \end{aligned}$$

Obteniendo límite cuando $\Delta \to 0^+$, se obtiene que

$$\lim_{\Delta \to 0^+} I(X^\Delta; Y^\Delta) = h(X) - h(X \mid Y) = I(X;Y).$$

□

Los resultados anteriores se retomarán en el Capítulo 3 como parte de una técnica que se utiliza para estimar la información mutua y así poder aplicar la metodología de IB.

1.2.1. Información mutua y algoritmos deterministas

Para esta sección, se ha optado por considerar un modelo sencillo, en el cual se aproxima la media empírica de un conjunto de variables aleatorias. Esto con el fin de aplicar los conceptos vistos en este capítulo y de ilustrar el uso de información mutua para monitorear la evolución de algoritmos. En este caso, se observará que la adición de ruido en el algoritmo es necesaria para que el uso de información mutua sea de provecho, aunque esto lleva consigo un cierto costo.

Supongamos que se tienen variables aleatorias X_1, X_2, $X_3, \ldots$ i.i.d. como $\mathcal{N}(0,1)$.

Se quiere conocer la cantidad de información que un conjunto de n variables preserva en cada iteración sobre su media empírica, considerando además un ruido gaussiano agregado.

Proposición 1.25. *Consideremos X_1, X_2, $X_3, \ldots$ variables aleatorias i.i.d. como $\mathcal{N}(0,1)$, $X^n = (X_1, \ldots, X_n)$ y para $\gamma > 0$, $T_{n,\gamma} = \frac{1}{n}\sum_{i=1}^n X_i + \sqrt{\gamma} Z$ donde $Z \sim \mathcal{N}(0,1)$ y es independiente de $X_1, \ldots, X_n$.*
Entonces se cumple que

$$I(X^n; T_{n,\gamma}) = \frac{1}{2} \log\left(1 + \frac{1}{n\gamma}\right).$$

Demostración. Se hará uso de la siguiente caracterización para vectores aleatorios gaussianos:

$$(X_1, \ldots, X_n) \text{ vector aleatorio gaussiano } \Leftrightarrow \forall a_1, \ldots, a_n \in \mathbb{R},\ \sum_{i=1}^n a_i X_i \text{ es v.a. gaussiana.}$$

Observar que $T_{n,\gamma} \sim \mathcal{N}(0, 1/n + \gamma)$, entonces, del Ejemplo 1.18,

$$h(T_{n,\gamma}) = \frac{1}{2} \ln\left(2\pi e (1/n + \gamma)\right). \tag{1.11}$$

Por lo anterior, $X^n \sim \mathcal{N}(\mathbf{0}_n, I_n)$ y $(X_1, \ldots, X_n, T_{n,\gamma}) \sim \mathcal{N}_{n+1}(\mathbf{0}_{n+1}, \Sigma_{n,\gamma})$ donde I_n es la matriz identidad de tamaño n, $\mathbf{0}_n = (0, \ldots, 0) \in \mathbb{R}^n$ y

$$\Sigma_{n,\gamma} = \begin{bmatrix} I_n & 1/n \cdot \mathbf{1_n} \\ 1/n \cdot \mathbf{1_n^t} & 1/n + \gamma \end{bmatrix}.$$

De la Proposición 1.21,

$$h(X^n) = \frac{1}{2} \log(2\pi e)^n \tag{1.12}$$

y se puede verificar que $\det(\Sigma_{n,\gamma}) = \gamma$, por lo que,

$$h(X_1, \ldots, X_n, T_{n,\gamma}) = \frac{1}{2} \log\left((2\pi e)^{n+1}\gamma\right). \tag{1.13}$$

De (1.11), (1.12) y (1.13) se tiene que

$$\begin{aligned} I(X^n; T_{n,\gamma}) &= h(X^n) + h(T_{n,\gamma}) - h(X^n, T_{n,\gamma}) \\ &= \frac{1}{2} \log\left(1 + \frac{1}{n\gamma}\right). \end{aligned} \tag{1.14}$$

□

Observemos que, como $I(X^n; T_{n,\gamma}) = \frac{1}{2} \log\left(1 + \frac{1}{n\gamma}\right)$, si hacemos que $\gamma \to 0^+$, entonces $I(X^n; T_{n,\gamma}) = \infty$.

Esto es, al agregar un ruido pequeño al promedio de las variables aleatorias $X_1, \ldots, X_n$, la información que hay en común entre ellas y su media empírica crece al infinito. Lo anterior es intuitivo, ya que el promedio empírico es un estimador insesgado que depende completamente de la muestra.

Por otra parte, si hacemos que $\gamma \to \infty$, entonces $I(X^n; T_{n,\gamma}) = 0$.

Es decir, cuando se agrega un ruido grande, se pierde la información que hay en común entre la media empírica y las variables. El término de ruido aleatorio le gana en magnitud al término de la media empírica y eso provoca que $T_{n,\gamma}$ y la muestra ya no compartan información en común.

De acuerdo a lo anterior, se enuncia el siguiente corolario:

Corolario 1.26. *Sean $X^n = (X_1, \ldots, X_n)$ con $X_1, X_2, X_3, \ldots$ v.a.i.i.d. como $\mathcal{N}(0,1)$ y $T_{n,\gamma} = \frac{1}{n}\sum_{i=1}^n X_i + \sqrt{\gamma}Z$ donde $\gamma > 0$, $Z \sim \mathcal{N}(0,1)$ y Z es independiente de $X_1, \ldots, X_n$. Entonces*

$$\lim_{\gamma \to 0} I(X^n; T_{n,\gamma}) = \infty$$

y

$$\lim_{\gamma \to \infty} I(X^n; T_{n,\gamma}) = 0$$

El resultado del corolario se ilustra en la gráfica de la Figura 1.1.

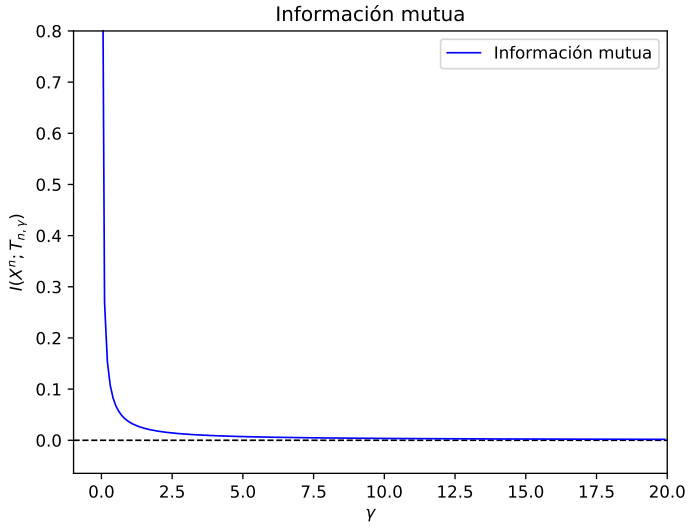

Figura 1.1: Gráfica de la información mutua entre X^n y $T_{n,\gamma}$, $I(X^n; T_{n,\gamma})$ con respecto a $\gamma > 0$, donde $X_1, \ldots, X_N$ v.a.i.i.d. como $\mathcal{N}(0,1)$ y $T_{n,\gamma} = \frac{1}{n}\sum_{i=1}^{n} X_i + \sqrt{\gamma}Z$, $Z \sim \mathcal{N}(0,1)$ y es independiente de X_1, X_2, $X_3, \ldots$.

Cabe resaltar que si en lugar de considerar $T_{n,\gamma}$ se hubiese considerado $T_n = T_{n,0}$, es decir, el promedio de las variables aleatorias sin ruido agregado, la información mutua no se habría podido calcular. Lo anterior debido a que (X^n, T_n) es vector gaussiano $n+1$-variado, con media $\mu = \mathbf{0}_n$ y matriz de covarianza $\Sigma_{n,0}$, de la cual se puede ver que el renglón $n+1$ es una combinación lineal de los n renglones anteriores. Es decir, $\Sigma_{n,0}$ es matriz singular y de esta manera se concluye que la distribución de $(X_1, \ldots, X_n, T_n)$ no tiene densidad, por lo tanto no se puede calcular la información mutua de X^n y T_n de acuerdo a la Definición 1.22.

Lo anterior da un primer acercamiento a uno de los problemas que se presentan al realizar cálculos de información mutua: no se tiene un comportamiento adecuado cuando se analizan algoritmos deterministas. Esto sugiere la necesidad de la presencia de aleatoriedad, que en este caso se da mediante la adición de un ruido gaussiano.

Capítulo 2

Machine Learning y Teoría de la Información

Como parte de las técnicas que son de interés analizar con la metodología de IB, se encuentran aquellas propias del aprendizaje máquina, algunas muy utilizadas en la actualidad por su grado de efectividad. Es por ello que en este capítulo veremos que se pueden analizar modelos de *machine learning* por medio de teoría de la información. Esta es una de las relaciones que se tienen entre estas dos áreas de conocimiento y está dada por medio de resultados contemporáneos que tienen implicaciones en la *generalización* de algoritmos.

Parte de este capítulo se basa en el libro de Shai Shalev-Shwartz & Shai Ben-David [10].

2.1. Modelo de aprendizaje supervisado

En esta sección vamos a considerar un problema de clasificación binaria. Pensemos, por ejemplo, que se tienen varios clientes en una compañía que otorga créditos. Es de interés para la compañía saber cuáles clientes son ideales para la aprobación de uno. La empresa cuenta con ciertos criterios en base a los cuáles asigna un puntaje a los clientes. De este puntaje es que se decide aprobar o rechazar la solicitud de crédito. Este es un problema que clásicamente se aborda con un modelo de aprendizaje supervisado y existen una gran variedad de herramientas estadísticas que dan solución a este tipo de modelos.

De manera general, para poder tener un **modelo formal de aprendizaje supervisado**, se necesita de lo siguiente:

- Un conjunto arbitrario $\mathcal{X}$ al que se conoce como dominio, que es el conjunto de características que se desea clasificar.
- Un conjunto arbitrario $\mathcal{Y}$ que es el conjunto de etiquetas que se asignarán a los objetos $x \in \mathcal{X}$.

- Una muestra $S_n = ((x_1, y_1), (x_2, y_2), \ldots, (x_n, y_n))$ donde $x_i \in \mathcal{X}$ y $y_i \in \mathcal{Y}$ para $i = 1, \ldots, n$, los cuales se consideran como datos de entrenamiento.

- Una familia $\mathcal{H}$ de funciones $h : \mathcal{X} \to \mathcal{Y}$ conocida como clase de hipótesis, la cual contiene a los clasificadores que pueden ser utilizados para predecir las etiquetas basadas en las características de los objetos.

- Una función de pérdida $l : \mathcal{H} \times \mathcal{Z} \to \mathbb{R}_{\geq 0}$, $\mathcal{Z} = \mathcal{X} \times \mathcal{Y}$, donde $l(h, z)$ representa el error de utilizar a la hipótesis h en los datos $z = (x, y)$.

Se denota por $\mathcal{A}(S_n)$ a la hipótesis que regresa un algoritmo de aprendizaje $\mathcal{A}$ al darle una muestra S_n.

Para el contexto del ejemplo de los créditos, un posible modelo de aprendizaje supervisado sería el descrito a continuación.

- $\mathcal{X} = \mathbb{R}$ el conjunto dominio, es decir, el conjunto en el que se encuentran los puntajes de los clientes;

- $\mathcal{Y} = \{-1, 1\}$ el conjunto de etiquetas: el cliente se cataloga con un -1 si el crédito es rechazado y con 1 en caso contrario;

- $\mathcal{H} = \{h_\theta, \theta \in \mathbb{R}\}$ la clase de hipótesis, con $h_\theta(x) = sgn(x - \theta)$.

Evaluar la precisión que se tiene al utilizar cierta hipótesis requiere de lo siguiente:

Definición 2.1. *Sea $l(h, z)$ función de pérdida. Se define el error en la predicción que da h como*

$$L(h) = \mathbb{E}\left[l(h, (X, Y))\right].$$

Sea $S_n = ((x_1, y_1), (x_2, y_2), \ldots, (x_n, y_n))$ una muestra. Se define el error empírico o error de entrenamiento como

$$L_{S_n}(h) = \frac{1}{n} \sum_{i=1}^{n} l(h, (x_i, y_i)).$$

Observemos que

$$\mathbb{E}\left[L_{S_n}(h)\right] = L(h),$$

lo cual se utiliza en la práctica debido a que para calcular el error de predicción, se necesita hacer uso de la distribución subyacente, la cual suele ser desconocida.

2.2. Desempeño y Generalización de Algoritmos

Una propiedad deseable en la salida de un algoritmo de aprendizaje, es que la hipótesis sea capaz de predecir las etiquetas de datos nuevos a partir del entrenamiento con la muestra. Por lo que una situación que se desea evitar es que la hipótesis sufra un sobre-ajuste y que sea incapaz de dar estimaciones certeras para datos distintos a la muestra. Es común que esta situación suceda debido a que la muestra es poco variada o inadecuada de acuerdo al problema que se quiere resolver. Cuando el error empírico es pequeño pero se tiene un caso como el mencionado, se suele decir que el algoritmo sufrió de ***overfitting*** o sobre-entrenamiento.

Para efectos de evitar el fenómeno de *overfitting*, es necesario medir la capacidad de predecir correctamente las etiquetas de datos nuevos dada alguna muestra. Por lo cual se define el error de generalización, el cual mide el nivel de sobre-entrenamiento de la salida del algoritmo. A mayor *overfitting*, mayor error de generalización.

Definición 2.2. *Se define al error de generalización de la hipótesis que devuelve un algoritmo $\mathcal{A}$ dada una muestra S_n como el valor esperado de la diferencia entre el error de predicción y el error empírico. Es decir,*

$$gen(\mathcal{A}(S_n)) = |\,\mathbb{E}\,[L(\mathcal{A}(S_n))] - \mathbb{E}\,[L_{S_n}(\mathcal{A}(S_n))]\,|.$$

Ejemplo 2.3. Retomando el ejemplo de la aprobación de créditos, supongamos que a los clientes se les asigna un puntaje en base a los criterios evaluados por la compañía. Por estrategia de mercado, si ese puntaje es mayor o igual que su valor esperado, el crédito se aprueba, de lo contrario se niega la autorización.

Supongamos que la distribución de los puntajes $X_1, X_2, \ldots, X_n$ es $\mathcal{N}(\mu, 1)$, pero permanece desconocida para la compañía.

De acuerdo a lo descrito, la clasificación respectiva a X_i sería

$$Y_i = sgn(X_i - \mu),$$

es decir, a los clientes se les otorgan los créditos si su puntaje está por encima de la media y se les rechaza en caso contrario.

Sea $S_n = ((x_1, y_1), \ldots, (x_n, y_n))$ una muestra de puntajes de clientes y su respectiva clasificación, obtenida del historial de la empresa.

En este contexto, tenemos las siguientes expresiones para la pérdida empírica y esperada.

Proposición 2.4. *Sea $\mathcal{X} = \mathbb{R}$ el conjunto dominio, $\mathcal{Y} = \{-1, 1\}$ el conjunto de etiquetas, $\mathcal{H} = \{h_\theta, \theta \in \mathbb{R}\}$ la clase de hipótesis, con $h_\theta(x) = sgn(x-\theta)$, $X_1, X_2, \ldots, X_n$ v.a.i. distribuidas como $X \sim \mathcal{N}(\mu, 1)$ y $S_n = ((X_1, Y_1), \ldots, (X_n, Y_n))$ una muestra donde $Y_i = sgn(X_i - \mu)$. Sea $l(h_\theta, (X, Y)) = (Y - h_\theta(X))^2$ función de pérdida. Entonces*

$$L_{S_n}(h_\theta) = \frac{4}{n} \sum_{i=1}^{n} \mathbf{1}_{X_i \in [\mu \wedge \theta,\, \mu \vee \theta]}. \tag{2.1}$$

En particular,

$$L(h_\theta) = 4\,\mathbb{P}\left(X \in [\mu \wedge \theta,\, \mu \vee \theta]\right).$$

Demostración. Por definición, el error empírico está dado por

$$\begin{aligned}
L_{S_n}(h_\theta) &= \frac{1}{n} \sum_{i=1}^{n} l(h_\theta, (X_i, Y_i)) \\
&= \frac{1}{n} \sum_{i=1}^{n} (Y_i - h_\theta(X_i))^2 \\
&= \frac{1}{n} \sum_{i=1}^{n} (1 - 2Y_i sgn(X_i - \theta) + (sgn(X_i - \theta))^2) \\
&= 2\left(1 - \frac{1}{n} \sum_{i=1}^{n} Y_i\, sgn(X_i - \theta)\right) \\
&= 2\left(1 - \frac{1}{n} \sum_{i=1}^{n} sgn(X_i - \mu)\, sgn(X_i - \theta)\right) \\
&= 2\left(1 - \frac{1}{n} \sum_{i=1}^{n} (1 - 2\,\mathbf{1}_{X_i \in [\mu \wedge \theta,\, \mu \vee \theta]})\right) \\
&= \frac{4}{n} \sum_{i=1}^{n} \mathbf{1}_{X_i \in [\mu \wedge \theta,\, \mu \vee \theta]}.
\end{aligned}$$

De lo anterior y debido a que $X_1, X_2, \ldots, X_n$ son i.i.d., se obtiene que

$$\begin{aligned}
L(h_\theta) &= \mathbb{E}\left[L_{S_n}(h_\theta)\right] \\
&= \frac{4}{n} \sum_{i=1}^{n} \mathbb{P}\left(X_i \in [\mu \wedge \theta,\, \mu \vee \theta]\right) \\
&= 4\,\mathbb{P}\left(X \in [\mu \wedge \theta,\, \mu \vee \theta]\right).
\end{aligned}$$

□

Es deseable que un algoritmo de aprendizaje tenga como salida una hipótesis h_θ con θ lo más parecido posible a μ. Como la entrada del algoritmo de aprendizaje es una muestra, es natural considerar a θ como el promedio empírico de los datos, ya que este es un estimador insesgado del parámetro μ.

Para $\gamma > 0$, consideremos el algoritmo

$$\mathcal{A}_\gamma(S_n) = h_{T_{n,\gamma}} \tag{2.2}$$

donde $T_{n,\gamma} = \frac{1}{n}\sum_{i=1}^n X_i + \sqrt{\gamma}Z$, $Z \sim \mathcal{N}(0,1)$ y es independiente de $X_1, \ldots, X_n$.

Observar que $T_{n,0}$ es el promedio de las variables y $T_{n,\gamma}$ es el promedio más un ruido gaussiano independiente.

A pesar de que el ruido agregado en (2.2) puede parecer artificial, podemos ver que si se supone que hubo errores en el muestreo, los datos X_i son en realidad $X_i' + \sqrt{\alpha}\, Z_i$ donde X_i' es real, $\alpha > 0$ y $Z_i \sim \mathcal{N}(0,1)$ independiente de X_1, X_2, …, X_n.

Definimos a $\tilde{T}_n^\alpha = \frac{1}{n}\sum_{i=1}^n X_i$.

Observemos que

$$\begin{aligned}
\tilde{T}_n^\alpha &= \frac{1}{n}\sum_{i=1}^n (X_i' + \sqrt{\alpha}\, Z_i) \\
&= \frac{1}{n}\sum_{i=1}^n X_i' + \frac{\sqrt{\alpha}}{n}\sum_{i=1}^n Z_i \\
&\overset{d}{=} \frac{1}{n}\sum_{i=1}^n X_i' + \frac{\sqrt{\alpha}}{n}\sqrt{n}\, Z, \quad Z \sim \mathcal{N}(0,1) \\
&= \frac{1}{n}\sum_{i=1}^n X_i' + \sqrt{\frac{\alpha}{n}}\, Z.
\end{aligned}$$

Haciendo $\gamma = \frac{\alpha}{n}$ se tiene que $\tilde{T}_n^\alpha = T_{n,\gamma}$ en distribución, es decir,

$$\mathcal{A}_0(S_n^\alpha) \overset{d}{=} \mathcal{A}_{\frac{\alpha}{n}}(S_n) \tag{2.3}$$

donde S_n^α es una muestra con ruido implícito en los datos.

Lo anterior nos dice que considerar solamente ruido implícito en los datos (por ejemplo errores en los cálculos del puntaje de los clientes) es equivalente a considerar el algoritmo con ruido gaussiano agregado, lo cual es conveniente dada la naturaleza imprecisa de los datos del mundo real.

Se tiene interés en acotar el error de estimación y el error de generalización de dicho algoritmo para tener una idea del comportamiento entre la precisión de este y su capacidad para generalizar.

Proposición 2.5. *Suponer el modelo de la Proposición 2.4 con el algoritmo*

$$\mathcal{A}_\gamma(S_n) = h_{T_{n,\gamma}}, \quad \gamma > 0,$$

donde $T_{n,\gamma} = \frac{1}{n}\sum_{i=1}^n X_i + \sqrt{\gamma}Z$, $Z \sim \mathcal{N}(0,1)$ *independiente de* $X_1, \ldots, X_n$. *Se cumple que*

$$\lim_{\gamma\to\infty} L(\mathcal{A}_\gamma(S_n)) = 2.$$

Más aún,

$$\mathbb{E}\left[L(\mathcal{A}_\gamma(S_n))\right] \leq 2,$$

es decir, el error de estimación está acotado superiormente por 2.

Demostración. Observemos que,

$$\begin{aligned} L(\mathcal{A}_\gamma(S_n)) &= 4\,\mathbb{P}\left(X \in [\mu \wedge T_{n,\gamma},\, \mu \vee T_{n,\gamma}] \mid S_n,\, Z\right) \\ &= 4\,\mathbb{P}\left(X - \mu \in [0 \wedge (T_{n,\gamma}-\mu),\, 0 \vee (T_{n,\gamma}-\mu)] \mid S_n,\, Z\right) \\ &= 4\int_{0\wedge(T_{n,\gamma}-\mu)}^{0\vee(T_{n,\gamma}-\mu)} \frac{1}{\sqrt{2\pi}}\, e^{-x^2/2} dx \\ &\leq \begin{cases} 4\int_0^\infty \frac{1}{\sqrt{2\pi}}\, e^{-x^2/2}dx = 2 & \text{cuando } T_{n,\gamma} \geq \mu, \\ 4\int_{-\infty}^0 \frac{1}{\sqrt{2\pi}}\, e^{-x^2/2}dx = 2 & \text{cuando } T_{n,\gamma} < \mu. \end{cases} \end{aligned} \tag{2.4}$$

Por lo que $\mathbb{E}\left[L(\mathcal{A}_\gamma(S_n))\right] \leq 2$.

Luego, para $\omega \in \Omega$ dado,

$$\begin{aligned} L(\mathcal{A}_\gamma(S_n)) &= 4\int_{0\wedge(T_{n,\gamma}(\omega)-\mu)}^{0\vee(T_{n,\gamma}(\omega)-\mu)} \frac{1}{\sqrt{2\pi}}\, e^{-x^2/2}dx \\ &= 4\int \mathbf{1}_{(0\wedge(T_{n,\gamma}(\omega)-\mu),\, 0\vee(T_{n,\gamma}(\omega)-\mu))}(x)\frac{1}{\sqrt{2\pi}}\, e^{-x^2/2}dx. \end{aligned}$$

Para $\gamma > 0$, definamos a

$$f_\gamma(x) = 4\,\mathbf{1}_{(0\wedge(T_{n,\gamma}(\omega)-\mu),\, 0\vee(T_{n,\gamma}(\omega)-\mu))}(x)\frac{1}{\sqrt{2\pi}}\, e^{-x^2/2}$$

y observemos que se tienen los siguientes cuatro casos:

I) $T_{n,\gamma}(\omega) \geq \mu \quad$ y $\quad Z(\omega) > 0.$

Dado $x \in \mathbb{R}$,

$$\lim_{\gamma\to\infty} \mathbf{1}_{(0,\, T_{n,\gamma}(\omega)-\mu)}(x) = \mathbf{1}_{(0,\infty)}(x),$$

por lo que

$$\lim_{\gamma\to\infty} f_\gamma(x) = 4\,\mathbf{1}_{(0,\infty)}(x)\frac{1}{\sqrt{2\pi}}\,e^{-x^2/2}.$$

II) $T_{n,\gamma}(\omega) \geq \mu \quad$ y $\quad Z(\omega) < 0.$

En este caso se satisface que, para $x \in \mathbb{R}$,

$$\lim_{\gamma\to\infty} \mathbf{1}_{(0\wedge(T_{n,\gamma}(\omega)-\mu),\, 0\vee(T_{n,\gamma}(\omega)-\mu))}(x) = \mathbf{1}_{(-\infty,\,0)}(x),$$

por lo que

$$\lim_{\gamma\to\infty} f_\gamma(x) = 4\,\mathbf{1}_{(-\infty,\,0)}(x)\frac{1}{\sqrt{2\pi}}\,e^{-x^2/2}.$$

III) $T_{n,\gamma}(\omega) \leq \mu \quad$ y $\quad Z(\omega) > 0.$

De manera similar a los casos anteriores,

$$\lim_{\gamma\to\infty} f_\gamma(x) = 4\,\mathbf{1}_{(0,\infty)}(x)\frac{1}{\sqrt{2\pi}}\,e^{-x^2/2}.$$

IV) $T_{n,\gamma}(\omega) \leq \mu \quad$ y $\quad Z(\omega) < 0.$

$$\lim_{\gamma\to\infty} f_\gamma(x) = 4\,\mathbf{1}_{(-\infty,\,0)}(x)\frac{1}{\sqrt{2\pi}}\,e^{-x^2/2}.$$

En cualquiera de los casos,

$$\int \lim_{\gamma\to\infty} f_\gamma(x)\,dx = 2.$$

Como $f_\gamma(x)$ es Borel medible y acotada para toda γ, por el Teorema de Convergencia Dominada se obtiene que

$$\lim_{\gamma\to\infty} L(\mathcal{A}_\gamma(S_n)) = 2 \quad c.\,s.$$

□

Por otro lado, se tiene la siguiente proposición.

Proposición 2.6 (Cota inferior para la pérdida esperada)**.** *Para el modelo de la Proposición 2.5 se cumple que*

$$\mathbb{E}\left[L(\mathcal{A}_\gamma(S_n))\right] \geq 2 - \frac{4}{\sqrt{1+\frac{1}{n}+\gamma}}.$$

Demostración. Haciendo $X' = X - \mu \sim \mathcal{N}(0,1)$ y $Y' = T_{n,\gamma} - \mu \sim \mathcal{N}(0, \frac{1}{n}+\gamma)$, de (2.4),

$$\begin{aligned} L(\mathcal{A}_\gamma(S_n)) &= 4\,\mathbb{P}\left(X' \in [0 \wedge Y',\, 0 \vee Y'] \mid S_n,\, Z\right) \\ &= 4\,\mathbb{P}(X' \in [0, |Y'|] \mid S_n,\, Z) \\ &= 4\left(\frac{1}{2} - \mathbb{P}(X' > |Y'| \mid S_n,\, Z)\right). \end{aligned}$$

Entonces

$$\begin{aligned} \mathbb{E}\left[L(\mathcal{A}_\gamma(S_n))\right] &= 2 - 4\,\mathbb{P}(X' > |Y'|) \\ &= 2 - 4\int_{-\infty}^{\infty} \mathbb{P}(X' > |y'|) \frac{1}{\sqrt{2\pi\left(\frac{1}{n}+\gamma\right)}} \exp\left(\frac{-y'^2}{2\left(\frac{1}{n}+\gamma\right)}\right) dy'. \end{aligned} \tag{2.5}$$

Dado que, para todo $\epsilon > 0$, si $Z \sim \mathcal{N}(0,1)$, $\mathbb{P}(Z > \epsilon) \leq e^{-\epsilon^2/2}$, se tiene que

$$\begin{aligned} \mathbb{E}\left[L(\mathcal{A}_\gamma(S_n))\right] &\geq 2 - 4\int_{-\infty}^{\infty} e^{-y'^2/2} \frac{1}{\sqrt{2\pi\left(\frac{1}{n}+\gamma\right)}} \exp\left(\frac{-y'^2}{2\left(\frac{1}{n}+\gamma\right)}\right) dy'. \\ &= 2 - 4\,\mathbb{E}\left[e^{-Y'^2/2}\right]. \end{aligned}$$

Como $Y' \overset{d}{=} \sqrt{\frac{1}{n}+\gamma}\, Z$, con $Z \sim \mathcal{N}(0,1)$,

$$\begin{aligned} 2 - 4\,\mathbb{E}\left[e^{-Y'^2/2}\right] &= 2 - 4\,\mathbb{E}\left[e^{-(1/n+\gamma)Z^2/2}\right] \\ &= 2 - 4M_{\chi^2_{(1)}}\left(-\frac{1}{2}\left(\frac{1}{n}+\gamma\right)\right), \end{aligned}$$

donde $M_{\chi^2_{(1)}}(t)$ es la función generadora de momentos de una variable aleatoria chi-cuadrada con un grado de libertad[1] $\left(\chi^2_{(1)}\right)$. Por lo tanto,

$$\mathbb{E}\left[L(\mathcal{A}_\gamma(S_n))\right] \geq 2 - \frac{4}{\sqrt{1+\frac{1}{n}+\gamma}}.$$

□

[1] $M_{\chi^2_{(1)}}(t) = (1-2t)^{-1/2}$ para $2t < 1$.

En la Figura 2.1 se observa la gráfica de una simulación del valor de la pérdida esperada y la cota de la Proposición 2.6. Para la realización de la simulación, se tomó en cuenta que, de (2.5),

$$\begin{aligned}\mathbb{E}\left[L(\mathcal{A}_\gamma(S_n))\right] &= 2 - 4\int \mathbb{P}(X > |y|)\frac{1}{\sqrt{2\pi(1/n+\gamma)}}\, e^{-\frac{y^2}{2(1/n+\gamma)}}\, dy \\ &= 2\int \operatorname{erf}\left(\frac{|y|}{\sqrt{2}}\right)\frac{1}{\sqrt{2\pi(1/n+\gamma)}}\, e^{-\frac{y^2}{2(1/n+\gamma)}}\, dy.\end{aligned}$$

La función $\operatorname{erf}(x)$ es la llamada función de error y está dada por $\operatorname{erf}(x) = \frac{2}{\sqrt{\pi}}\int_0^x e^{-t^2}dt$.

Así, el valor de $\mathbb{E}\left[L(\mathcal{A}_\gamma(S_n))\right]$ se calculó numéricamente con ayuda de la paquetería `scipy` en python.

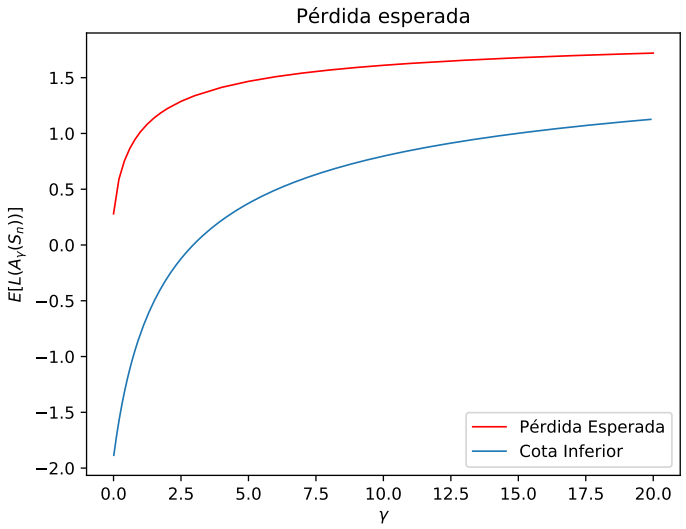

Figura 2.1: Gráfica de la pérdida esperada del algoritmo $\mathcal{A}_\gamma$ con respecto a γ y una cota inferior.

Por otro lado, el desempeño de un algoritmo se puede acotar con el error de generalización y el error empírico de la siguiente manera:

$$\mathbb{E}\left[L(\mathcal{A}(S_n))\right] \leq gen(\mathcal{A}(S_n)) + \mathbb{E}\left[L_{S_n}(\mathcal{A}(S_n))\right].$$

Entonces, es posible que se den casos en los cuales un algoritmo tiene buen desempeño en la muestra pero mal desempeño en general y por consiguiente una mala generalización. O también se puede dar el caso en el que el algoritmo generaliza adecuadamente pero el desempeño en general es pobre. En otras palabras, una buena generalización no es sinónimo de un buen desempeño. En la práctica, el error empírico suele ser cercano a cero; es por ello que acotar al error de generalización es la alternativa más viable

para tener un control sobre el desempeño del modelo, ya que un cálculo directo no es realizable en la mayoría de los casos. Un enfoque contemporáneo para llevar a cabo dicha tarea es mediante información mutua, lo cual se verá en la siguiente sección.

2.3. Generalización e Información Mutua

Ahora lo que se busca es acotar el error de generalización del algoritmo $\mathcal{A}_\gamma(S_n)$. Para esto, se recurre al siguiente teorema, el cual es un resultado atribuido a Xu A. y Raginsky M. (2017).[2]

Teorema 2.7. *Sea $\mathcal{H} = \{h_w : w \in \mathcal{W}\}$ clase de hipótesis. Si $l(h_w, z) \in [0, c]$ para toda $w \in \mathcal{W}$ y W es parámetro del algoritmo $\mathcal{A}$ dada una muestra, entonces*

$$gen(\mathcal{A}(S_n)) \leq \sqrt{\frac{c^2}{2n} I(S_n; W)},$$

para toda $z \in \mathcal{Z}$.

Aplicando el teorema anterior al modelo considerado a lo largo de esta sección, se obtiene lo siguiente.

Proposición 2.8 (Cota superior para el error de generalización)**.** *Una cota para el error de generalización del modelo en la Proposición 2.5, está dada como sigue:*

$$gen(\mathcal{A}_\gamma(S_n)) \leq \sqrt{\frac{4}{n} \log\left(1 + \frac{1}{n\gamma}\right)}$$

para $\gamma > 0$.

Demostración. Observemos que, de (2.1),

$$L_{S_n}(\mathcal{A}_\gamma(S_n)) = \frac{1}{n} \sum_{i=1}^{n} l(h_{T_{n,\gamma}}, (x_i, y_i)) \tag{2.6}$$

donde $l(h_{T_{n,\gamma}}, (x_i, y_i)) = 4\,\mathbf{1}_{x_i \in [\mu \wedge T_{n,\gamma},\, \mu \vee T_{n,\gamma}]}$. Como $0 \leq l(h_{T_{n,\gamma}}, (x_i, y_i)) \leq 4$, por el Teorema 2.7

$$|\,\mathbb{E}\,[L(\mathcal{A}_\gamma(S_n))] - \mathbb{E}\,[L_{S_n}(\mathcal{A}_\gamma(S_n))]\,| \leq \sqrt{\frac{8}{n} I(S_n; T_{n,\gamma})}.$$

Recordemos la Proposición 1.25, la cual nos dice que $I(S_n; T_{n,\gamma}) = \frac{1}{2} \log\left(1 + \frac{1}{n\gamma}\right)$. Entonces se obtiene que

$$|\,\mathbb{E}\,[L(\mathcal{A}_\gamma(S_n))] - \mathbb{E}\,[L_{S_n}(\mathcal{A}_\gamma(S_n))]\,| \leq \sqrt{\frac{4}{n} \log\left(1 + \frac{1}{n\gamma}\right)}.$$

□

[2]La versión general de este teorema hace uso de un tipo de variables aleatorias llamadas subgaussianas, el cual puede ser consultado en [6].

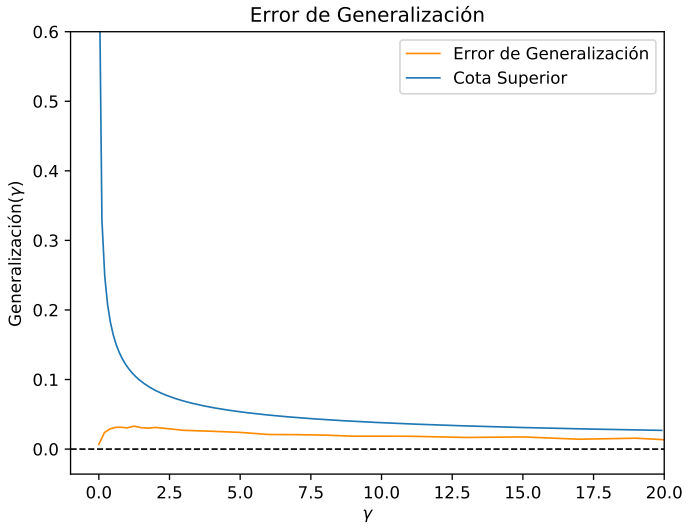

Figura 2.2: Gráfica del error de generalización del algoritmo $\mathcal{A}_\gamma$ con respecto a γ.

En la Figura 2.2 se ilustra el resultado de la Proposición 2.8. El valor del error de generalización se obtuvo mediante el cálculo numérico de la pérdida esperada (Figura 2.1) y mediante la simulación de la pérdida empírica: de (2.6), se obtiene que

$$\mathbb{E}\left[L_{S_n}(\mathcal{A}_\gamma(S_n))\right] = \frac{1}{n}\sum_{i=1}^{n} 4\,\mathbb{P}\left(X_i \in [\mu \wedge T_{n,\gamma},\, \mu \vee T_{n,\gamma}]\right).$$

Entonces, para la estimación de $\mathbb{P}\left(X_i \in [\mu \wedge T_{n,\gamma},\, \mu \vee T_{n,\gamma}]\right)$ se consideró el promedio de $k = 500,000$ repeticiones y se tomaron en cuenta $n = 20$ muestras por cada repetición, con $\mu = 2$. El tiempo computacional fue de aproximadamente 20 minutos en un equipo con 4 GB de memoria RAM y un procesador 2.5 GHz Intel Core i5.

De la Proposición 2.8 y la Figura 2.2 se puede concluir que controlando la información mutua entre la entrada y la salida del algoritmo mediante la adición de ruido aleatorio, se puede mejorar la generalización del mismo. Esto permite encontrar un balance entre la generalización del algoritmo y la cantidad de información que comparten la entrada y la salida del mismo. Más aún, la relación en (2.3) acerca de ruido implícito en la muestra, permite proponer una posible explicación al por qué de la buena generalización que presentan los algoritmos de aprendizaje máquina en general. Es decir, podría ser que en la práctica los algoritmos suelen no sufrir de *overfitting* dado que los datos que se pueden extraer del mundo real tienden a tener errores de medición y/o ruido debido a la naturaleza, etc.

Poder relacionar información mutua y generalización se logró gracias al resultado del Teorema 2.7, el cual utiliza técnicas contemporáneas de teoría de la información. En

ese mismo sentido es que se desarrolla la metodología de Information Bottleneck, en la cual se trata de entender la evolución de técnicas estadísticas a través de información mutua entre entrada y salida de algoritmos de aprendizaje. Una de las técnicas de machine learning que se han analizado bajo una perspectiva de información son las redes neuronales, las cuales se introducen en la siguiente sección. Debido a estos trabajos la metodología de IB se ha posicionado en la mira de la comunidad científica.

2.4. Redes Neuronales

Una de las técnicas más utilizadas en la actualidad y que son características de aprendizaje máquina, son las redes neuronales. Están inspiradas en el funcionamiento del cerebro humano: las neuronas se entrelazan unas con otras e intercambian la información para procesarla y así obtener resultados.

El modelo de red neuronal más sencillo se denomina *perceptrón* y se define a continuación.

Definición 2.9. *Sea $\sigma : \mathbb{R} \to \mathbb{R}$ una función dada. Para $w = (w_1, \ldots, w_n) \in \mathbb{R}^n$ y $b \in \mathbb{R}$, el perceptrón con pesos w y bias b es la función $f_{w,b}(x) : \mathbb{R}^n \to \mathbb{R}$ dada por*

$$f_{w,b}(x) = \sigma(w \cdot x + b)$$

para $x = (x_1, \ldots, x_n)$.

En la Figura 2.3 se ilustra una representación usual de un modelo perceptrón. Cada dato de entrada x_i se asigna a una *neurona* que a su vez está conectada con otra que arroja un resultado, es decir, se tienen n neuronas de entrada y una de salida. En la neurona de salida se realiza una suma ponderada de los datos de entrada y al resultado obtenido se le aplica la función σ, obteniendo $f_{w,b}(x)$ como salida de la red.

A la función $\sigma : \mathbb{R} \to \mathbb{R}$ se le llama *función de activación*. Las funciones de activación comúnmente utilizadas son las siguientes:

$\sigma(x) = sign(x)$	función signo,
$\sigma(x) = \text{máx}\{0, x\}$	función ReLu,
$\sigma(x) = \dfrac{1}{1+e^{-x}}$	función sigmoide,
$\sigma(x) = \tanh(x)$	tangente hiperbólica,
$\sigma(x) = \mathbf{1}_{x \geq 0}$	función umbral.

Observar que en el modelo de perceptrón, si la función de activación es la función sigmoide, el modelo resultante es una **regresión logística** (ver por ejemplo [10], Sección 9.3).

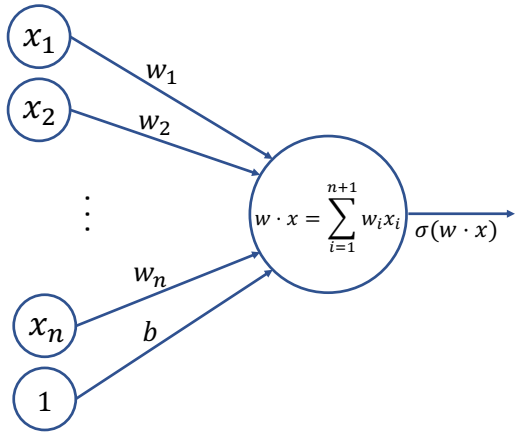

Figura 2.3: Modelo perceptrón. Los datos de entrada son $x_1, \ldots, x_n$ y $x_{n+1} = 1$; los pesos de cada vértice son $w_1, \ldots, w_n$ y $w_{n+1} = b$; $\sigma : \mathbb{R} \to \mathbb{R}$ es función de activación. En la neurona se efectúa una suma ponderada y como salida se tiene la función de activación aplicada a dicha suma.

El modelo de perceptrón se puede extender a un modelo de red neuronal más complejo en el cual las neuronas tienen un mayor número de conexiones entre ellas.

Definición 2.10 (Red neuronal (*fully connected feedforward*))**.** *Dadas* $\sigma : \mathbb{R} \to \mathbb{R}$ *función de activación,* $d \in \mathbb{N}$, $V_0, V_1, \ldots, V_d \in \mathbb{N}$ *con* $V_0 = m$ *y* $V_d = n$, *la red neuronal (fully connected feedforward) con pesos* $w = (w_1, w_2, \ldots, w_d) \in \mathbb{R}^{V_1 \times V_0} \times \cdots \times \mathbb{R}^{V_d \times V_{d-1}}$ *y bias* $b = (b_1, b_2, \cdots, b_d) \in \mathbb{R}^{V_1} \times \cdots \times \mathbb{R}^{V_d}$ *es la función* $f_{w,b} : \mathbb{R}^m \to \mathbb{R}^n$ *dada por*

$$f_{w,b}(x) = \sigma.\left(w_d\, \sigma.(\cdots \sigma.(w_2\, \sigma.(w_1 \cdot x + b_1)) \cdots) + b_d\right),$$

donde $\sigma.$ *es aplicar la función* σ *entrada por entrada.*

Al conjunto de componentes que participan en un modelo de red neuronal (neuronas, conexiones, función de activación) se le denomina *arquitectura* de la red.

Para tener una mejor organización de la red, las neuronas se dividen en subconjuntos $T_0, \ldots, T_d$ disjuntos entre sí, con cardinalidades $V_0 + 1, \ldots, V_{d-1} + 1, V_d$ respectivamente. A cada subconjunto T_i se le denomina *capa* de la red y a las capas $T_1, \ldots, T_{d-1}$ se les llama *capas ocultas*. El número d es conocido como la *profundidad* de la red. Cuando la red tiene profundidad mayor o igual a 3 se le denomina *red neuronal profunda*.

Cada neurona perteneciente a la capa T_i se conecta solamente con cada una de las neuronas pertenecientes a la capa T_{i+1}, para $i = 0, \ldots, d-1$. A este tipo de redes se les denomina *fully connected feedforward*, que es el tipo de red neuronal en el que nos centraremos.

En la Figura 2.4 podemos observar una representación de una red neuronal, la cual tiene una *capa oculta*.

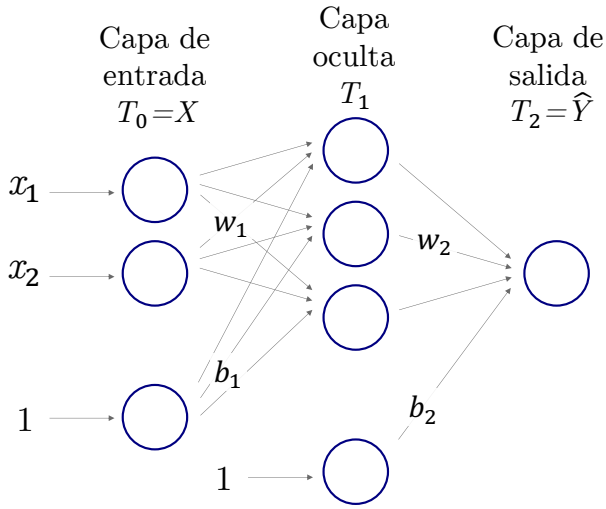

Figura 2.4: Red neuronal (*fully connected feedforward*), con una capa oculta. Su arquitectura consta de 3 capas con $3-4-1$ neuronas respectivamente. Los datos de entrada son $x = (x_1, x_2)$; las matrices de pesos $w_1 \in \mathbb{R}^{3\times 2}$ y $w_2 \in \mathbb{R}^{1\times 3}$; los bias $b_1 \in \mathbb{R}^3$ y $b_2 \in \mathbb{R}$.

Las redes neuronales como modelos de aprendizaje supervisado, pueden dar solución a problemas de clasificación y predicción. Los datos de entrada de la primer capa son las características a clasificar. En todas las neuronas pertenecientes a una capa, la red realiza transformaciones de estos datos y transmite la información a las neuronas de la siguiente capa. En cada neurona receptora se realiza una suma ponderada de la información proveniente de las neuronas de la capa anterior y que están conectadas a ella. La información de la última capa es la salida de la red neuronal.

En la práctica, para poder encontrar los pesos w, una red neuronal se somete a un proceso de entrenamiento, para el cual se debe contar con una muestra de características junto con sus respectivas clasificaciones. Las características en la muestra son las entradas de la red, los pesos se fijan arbitrariamente y la salida de la red se compara con las clasificaciones verdaderas mediante una función de pérdida. Luego, mediante un método de optimización se ajustan los pesos de tal manera que el valor de la pérdida empírica se minimice. Es decir, se busca que la salida de la red coincida lo mejor posible con la clasificación verdadera de los datos de entrada.

2.4.1. *Stochastic Gradient Descent*

El método de optimización más común mediante el cual se ajustan los pesos de una red neuronal se llama *Stochastic Gradient Descent*. Es una versión aleatorizada del algoritmo de *Gradient Descent*. El algoritmo en general está dado a continuación.

Dado $K \subset \mathbb{R}^d$ conjunto convexo y cerrado, definimos a $\pi_k : \mathbb{R}^d \to K$ como $\pi_k(y) \in arg\min_{x_0 \in K} \|y - x_0\|$.

Sean además (X_1, Y_1), (X_2, Y_2), ..., (X_n, Y_n) i. i. d. $\sim (X, Y)$ y $f_t(x) = l(x, (X_t, Y_t))$.

Algoritmo *Stochastic Gradient Descent*

Parámetros: $T \in \mathbb{N}$ (iteraciones), $\eta > 0$ (tasa de aprendizaje), $x_0 \in \mathbb{R}^d$ (inicialización)

1. Para $t = 0, 1, \ldots, T$
2. $\quad y_{t+1} = x_t - \eta \bigtriangledown f_{t+1}(x_t)$
3. $\quad x_{t+1} = \pi_k(y_{t+1})$
4. Regresar $(x_0, x_1, \ldots, x_T)$

Para el caso de redes neuronales, la función objetivo podría no ser convexa, lo cual es fuertemente utilizado en el algoritmo anterior. Por lo tanto, se utiliza una versión ligeramente modificada. El vector $w^{(0)}$ con el cual se inicializa, debe ser cercano a cero y elegido aleatoriamente.

Algoritmo *Stochastic Gradient Descent* para redes neuronales

---

Parámetros: $T \in \mathbb{N}$ (iteraciones), $\eta > 0$ (tasa de aprendizaje), $\lambda \geq 0$ (regularización)

1. Tomar aleatoriamente $w^{(0)} \in \mathbb{R}^{V_0 \cdot V_1 + \cdots + V_{d-1} \cdot V_d}$
2. Para $i = 0, \ldots, T-1$
3. $\quad$ Tomar aleatoriamente $(x, y) \in S_n$
4. $\quad w^{(i+1)} = w^{(i)} - \eta(\nabla_w L_{(x,y)}(w^{(i)}) + \lambda w^{(i)})$
5. Salida: $w^{(T)}$

A cada iteración se le suele denominar *época*. En este algoritmo se calcula el gradiente de la función de pérdida empírica en busca de los pesos w que tengan mayor poder de predicción. Para esto, la función de activación debe ser diferenciable. Dicho gradiente no tiene una forma explícita, por lo cual se calcula utilizando un algoritmo llamado *backpropagation*. Cuando la función de activación no es diferenciable se suelen emplear otros algoritmos de optimización[3].

Después del proceso de entrenamiento, se suele comprobar el poder de predicción de la red con un conjunto de prueba, cuyos datos no se utilizan durante el proceso de entrenamiento. En el proceso de prueba se utilizan los pesos calculados durante el entrenamiento para así comprobar qué tan acertada es la predicción realizada sobre datos nuevos.

Podemos observar que el algoritmo que se utiliza para redes neuronales cuenta con aleatoriedad agregada al inicializar y durante el paso de las épocas. Esto nos da la sugerencia de que la información mutua podría tener un comportamiento deseado al analizar redes neuronales entrenadas con este algoritmo.

Por último, cabe resaltar que en el contexto de computación, puede pensarse a una red neuronal como un grafo dirigido en el cual cada nodo representa una neurona y los vértices, las conexiones entre ellas.

[3]Ver por ejemplo Walia, A. S. (2017). *Types of Optimization Algorithms used in Neural Networks and Ways to Optimize Gradient Descent*. Towards Data Science.

Capítulo 3

Metodología de Information Bottleneck

En este capítulo se describe en qué consiste la metodología de *Information Bottleneck* y la manera en que se ha llevado a cabo su aplicación a redes neuronales. Se explican los experimentos realizados por los autores en [1] y [2] y se mencionan las conjeturas propuestas en sus trabajos.

3.1. Curva de Information Bottleneck

En 1999, Tishby et al. propusieron la metodología de Information Bottleneck (IB) como un marco común de trabajo en el que se pudieran analizar diferentes problemas de procesamiento de información, como predicción, aprendizaje, filtrado, etc. La filosofía detrás de esta metodología es la siguiente:

Para una variable, denotada por X, la cual representa una señal, se quiere extraer información relevante sobre otra señal, la cual se denotará por Y y será la variable de relevancia. Ambas variables deben tener información mutua positiva, es decir, no deben ser independientes (Corolario 1.11). Se asume que se tiene acceso a la distribución conjunta $p(x, y)$. El problema es encontrar una representación de X, denotada por T, que extraiga la mínima información de X (es decir, que comprima la información de X) y preserve la máxima información acerca de Y (que estime correctamente a Y).

Recordar que lo anterior se puede escribir en términos de información mutua con ayuda de la desigualdad de Fano y debido a que las variables X, Y y T forman una cadena de Markov $Y - X - T - \hat{Y}$.

El límite fundamental entre compresión y precisión es la curva de Information Bottleneck, la cual se define a continuación y se ilustra en la Figura 3.1.

Definición 3.1 (Curva de Information Bottleneck)**.** *Para un nivel de compresión $\epsilon > 0$ de la variable aleatoria X, se define*

$$IB(\epsilon) = \sup_{T:I(X;T)\leq\epsilon} I(Y;T),$$

donde T es una representación de X y Y es la variable de relevancia.
La curva de Information Bottleneck está dada por el conjunto $\{IB(\epsilon) : \epsilon > 0\}$.

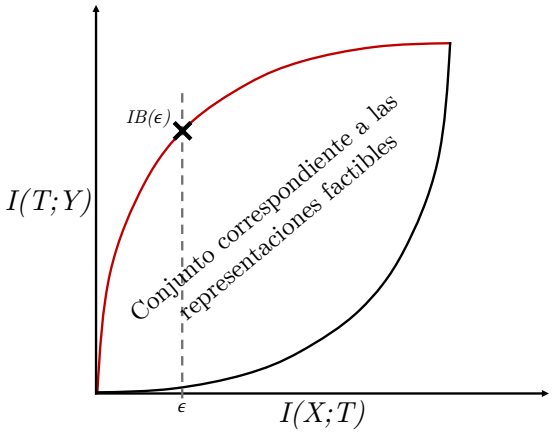

Figura 3.1: Plano de información en el cual se muestra la región correspondiente a posibles representaciones de X. En rojo se muestra la curva de Information Bottleneck.

En la Figura 3.1 se ilustra el plano de información en el cual se grafica la información entre la variable de entrada y su representación contra la información entre la variable de relevancia y la representación, es decir, $I(X;T)$ contra $I(T;Y)$. A cada representación factible T, le corresponde un único punto en el plano y al considerar a todas las representaciones, se forma el área que se muestra en la figura. Así, si se quiere obtener una compresión de X de a lo más ϵ, la representación que tiene el mayor poder predictivo de Y es la que se encuentra sobre la curva de IB.

La idea de la metodología es graficar sobre el plano de información, $I(X;T)$ e $I(T;Y)$, lo cual nos da un marco común de trabajo para cualquier problema de aprendizaje que se desee analizar. De esta forma se podría tener una idea de la manera en la que procesan la información distintas técnicas estadísticas.

3.2. Information Bottleneck y Redes Neuronales

En esta sección se analizarán los resultados propuestos por Tishby et al. (2017) y Saxe et al. (2018) ([1] y [2] respectivamente).

Como se puede notar, el contexto de la metodología de IB es muy general, por lo que es posible aplicarla a cualquier problema en el que se deba estimar o predecir cierta información a partir de otra. Es por ello, que en 2015, Tishby y Zaslavsky [4] proponen analizar la dinámica de redes neuronales a través del marco de trabajo de IB, dado que estas siguen siendo cajas negras en cuanto a su funcionamiento.

Como la información generada por cada capa de la red depende solamente de la anterior, estas forman una cadena de Markov

$$X - T_1 - T_2 - \cdots - T_m - \hat{Y}$$

en donde a cada capa se le asigna una variable aleatoria T_i.

Durante el periodo de entrenamiento, una red neuronal aprende a crear representaciones T_i de X que contienen las características necesarias para maximizar $\Pr(Y = \hat{Y})$. La desigualdad de Fano, en su versión dada por (1.9),

$$P_e \geq \frac{H(Y) - I(Y;T_i) - 1}{\log|\mathcal{Y}|},$$

sugiere que simultáneamente se maximiza $I(Y;T_i)$.

La manera de representar una red neuronal en el plano de información es a través de la información que contienen las variables de interés y de entrada con respecto a las capas de la red. Para cada capa T_i se tendrá una trayectoria en la cual se compara $I(X;T_i)$ contra $I(Y;T_i)$, cantidades que se calculan por medio de estimaciones, dada una muestra finita de $p(X,Y)$ (datos de entrenamiento). De esta manera, se puede comparar la dinámica en el plano de información de cada capa de la red con la curva de IB, que como ya se mencionó, es el límite fundamental entre compresión y predicción. Lo anterior podría revelar nuevos comportamientos de las redes neuronales de acuerdo a sus arquitecturas, nuevos criterios de optimización, nuevas cotas para generalización, etc.

Luego, en 2017, Tishby y Schwartz realizan un trabajo [1] en el cual siguen las ideas propuestas en 2015 [4] para visualizar el comportamiento de las capas de redes neuronales en el plano de información. Ellos proponen un modelo, al cual se hará referencia como el modelo SZT, en el cual estiman la información mutua $I(X;T_i)$ e $I(Y;T_i)$. Después grafican las estimaciones en el plano de información para cada capa T_i y proponen varias conjeturas de acuerdo a la dinámica observada.

El modelo SZT es una red neuronal *fully connected feed forward*, cuya arquitectura es la siguiente: 7 capas con tamaños de $12 - 10 - 7 - 5 - 4 - 3 - 2$ neuronas, con función de activación tanh para todas las neuronas, excepto para la capa final, en la cual se

utiliza la activación sigmoidal. La red se entrenó con *stochastic gradient descent* y la función de pérdida logarítmica. Los datos utilizados para el entrenamiento de la red fueron 12 puntos distribuidos uniformemente en una esfera 2D etiquetados con 0 o 1 según cierta regla de simetría dada.

En base a las visualizaciones obtenidas (un ejemplo se muestra en la Figura 3.2), algunas de las conjeturas que proponen son las siguientes:

(I) El entrenamiento de la red muestra dos fases, una de compresión de la información y otra de ajuste de las etiquetas. La mayoría de las épocas de entrenamiento son ocupadas en la primer fase.

(II) La compresión comienza cuando el error de entrenamiento se hace pequeño y el SGD cambia de una deriva rápida a una relajación estocástica.

(III) La dinámica de las capas de la red se acerca a la curva de IB conforme avanzan las épocas de entrenamiento.

(IV) Las redes profundas muestran una buena generalización en menor número de épocas que las redes con una capa oculta.

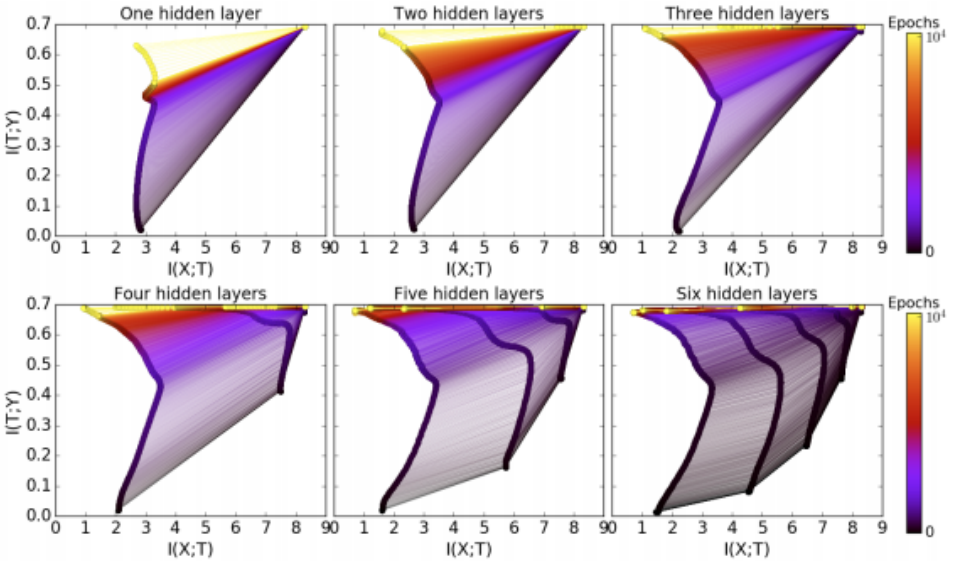

Figura 3.2: Resultados mostrados en [1]. Dinámica de aprendizaje de redes neuronales con distintas arquitecturas en el plano de información. Cada una de las trayectorias observadas corresponde a la dinámica promedio de cada capa de la red para 50 entrenamientos con inicializaciones y muestras distintas.

El trabajo anterior ha generado polémica debido a que las conjeturas están basadas en lo observado en experimentos para algunos ejemplos de arquitectura de red neuronal.

En 2018, Saxe et al. [2] muestran ejemplos de redes neuronales que no cumplen con algunas de las conjeturas de Tishby.

(i) Ellos reproducen sus experimentos con el modelo SZT y muestran comportamientos similares. Luego, cambian la función de activación tanh por una ReLu y las trayectorias observadas no muestran compresión en su comportamiento.

(ii) Entrenan otra red con arquitectura distinta y con datos de la MNIST, nuevamente utilizando activación tanh y observan fase de compresión. Repiten el experimento pero con activación ReLu y nuevamente se observa que no hay fase de compresión.

(iii) Realizan otro experimento, ahora con una red cuya arquitectura es solo de 3 neuronas y entrenan con solo el 30 % de los datos considerados. Observan que existe compresión, pero la red no generaliza.

(iv) Otro experimento que muestran en su trabajo, es el de una red entrenada con SGD, con activaciones tanh y luego ReLu; entrenan esa misma red pero ahora con Batch Gradient Descent (BGD) y las mismas activaciones. Observan que las dinámicas de las capas en el plano de información muestran comportamientos similares tanto para las redes con SGD como para las entrenadas con BGD. Las redes con tanh muestran compresión.

(v) Concluyen que la fase de compresión observada por Tishby depende de la activación utilizada, no es consecuencia de la estocasticidad de SGD y no está relacionada con la generalización de la red.

3.3. Information Bottleneck y Estimación de Información Mutua

Una de las principales dificultades que surgen al aplicar la metodología de IB, es el problema de estimar la información mutua, ya que en la práctica no se conocen las distribuciones tanto conjuntas como marginales de las variables que participan en el modelo. En este capítulo se ilustrará mediante simulaciones el porqué de las variantes en los resultados que se han obtenido a lo largo del estudio sobre IB, por las cuales han surgido polémicas al respecto.

3.3.1. Estimación de información mutua mediante discretizaciones de las variables

La primera de las maneras en que se propuso estimar la información mutua entre las variables de entrada y sus representaciones en redes neuronales es mediante una

discretización de las variables en cuestión. Esta es la forma principal en la que se realizaron los experimentos en [1] y [2].

Supongamos que se tiene una red neuronal *fully connected feed forward* de n capas ocultas. Denotamos por X a la variable de entrada y por T_i a la representación de X que surge en la capa i. Dichas variables se consideran continuas ya que el rango de valores que puede tomar cada neurona en una capa se encuentra en la recta real. Pero al entrenar la red, lo que se obtiene en cada capa son valores discretos dependiendo de las neuronas con que se cuente. Estos valores se toman como una muestra de la variable.

Por lo que, para realizar la estimación de la información mutua, se considera al valor más pequeño que arroja la capa y al más grande, entonces el rango en que se encuentran los valores de la capa se divide en intervalos de tamaño Δ. Se cuenta el número de valores que caen dentro de cada intervalo y de esta manera se estima la densidad de una versión discretizada de la variable.

Entonces, a grandes rasgos, en los artículos [1] y [2], están considerando la información mutua de variables aleatorias continuas como una estimación de la información mutua entre versiones discretizadas de las variables. Dichas versiones discretizadas son obtenidas mediante muestras generadas durante el entrenamiento de la red, es decir, mediante los valores que se obtienen en cada neurona.

La manera en como se discretiza la variable y el fundamento teórico que hay detrás de lo descrito anteriormente se menciona en la Proposición 1.24 que dice lo siguiente:

Sean X y Y variables aleatorias con densidad conjunta $f(x, y)$. Entonces la información mutua entre las dos variables aleatorias es el límite de la información mutua entre las variables discretizadas correspondientes. Es decir,

$$\lim_{\Delta \to 0} I(X^{\Delta}; Y^{\Delta}) = I(X; Y).$$

De la definición de información mutua

$$I(X;Y) = \int f(x,y) \log \frac{f(x,y)}{f(x)f(y)}\, dx\, dy,$$

se aprecia que para calcular su valor se consideran solamente a las distribuciones de las variables. Más aún, cada valor posible de las variables contribuyen en promedio al valor de la información. Es por ello que se esperaría que de una sola muestra de las variables no se obtengan estimaciones acertadas de la información.

A continuación se muestran varios ejemplos en los cuales se estima la información mutua de versiones discretizadas de variables para las cuales se conocerán sus distribuciones.

Esto con el fin de ilustrar que este procedimiento de estimación se debe tratar con mayor rigor del que podría aparentar.

Ejemplo 3.2. Sean X y Y v.a. $\mathcal{N}(0,1)$ correlacionadas con $cov(X, Y) = \rho$. De acuerdo a la Proposición 1.21 y a la relación 1.10,

$$I(X; Y) = -\frac{1}{2}\log(1-\rho^2).$$

Considerando $\rho = 0{,}5$, se realizan varias estimaciones del valor anterior en base a muestras de las variables X y Y, generadas mediante la librería `scipy.stats.multivariate_normal` de python.

En la Figura 3.3 se ilustra el valor real de $I(X; Y)$ (IM teórica); el valor de $I(X^\Delta; Y^\Delta)$ para distintos valores[1] de Δ (IM discretizada); y para los mismos valores de Δ, se grafica $\hat{I}(X^\Delta; Y^\Delta)$. Para calcular cada curva de $\hat{I}(X^\Delta; Y^\Delta)$, las funciones de probabilidad conjuntas y marginales han sido estimadas de acuerdo a una muestra de tamaño $N = 10$ de (X, Y) para distintos valores de Δ (IM estimada).

Se grafican 30 curvas de información mutua estimada para generar una banda empírica que ilustre la aleatoriedad que se tiene al estimar la información mutua de esta manera y con una sola muestra. Se grafica además el promedio de las 30 curvas (Promedio muestral) y una banda del promedio $\pm$ menos 3 desviaciones estándar.

Se observa que existe mucha variabilidad entre una muestra y otra en cuanto a la estimación de la información mutua, además de que para una sola muestra se tienen muchas fluctuaciones en la estimación. La curva de la información mutua $I(X^\Delta; Y^\Delta)$ (color azul), la cual se calculó haciendo uso de las funciones de probabilidad correspondientes, converge al valor de la información mutua $I(X; Y) = -\frac{1}{2}\log(1-\rho^2)$, que es justo lo que enuncia el Teorema 1.24. Se esperaría que una curva de estimación de $I(X;Y)$, se parezca lo más posible a la curva de $I(X^\Delta; Y^\Delta)$, lo cual no se aprecia en este caso.

Se realizó el mismo experimento varias veces pero con distintos tamaños de las muestras: $N = 50, 100, 200$. En las gráficas de las Figuras 3.4, 3.5 y 3.6, se aprecian los resultados obtenidos. Nuevamente, las bandas se realizaron con 30 muestras (de tamaño N).

[1]Para los valores de Δ se consideró una partición del intervalo de 0.005 a 3 con tamaño de particiones de 1/20.

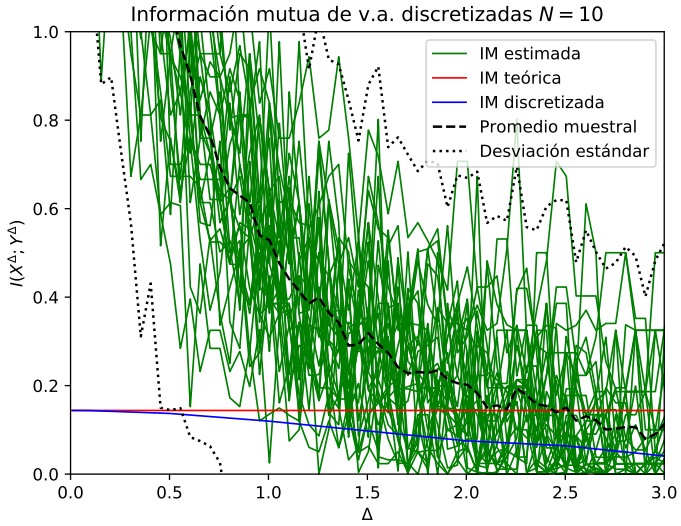

Figura 3.3: Gráfica de la información mutua entre X^Δ y Y^Δ (discretizaciones de las variables unidimensionales X y Y respectivamente) con respecto a Δ, así como también $\hat{I}(X^\Delta; Y^\Delta)$ que es una estimación de $I(X^\Delta; Y^\Delta)$ con 30 muestras de (X, Y), cada una de tamaño $N = 10$.

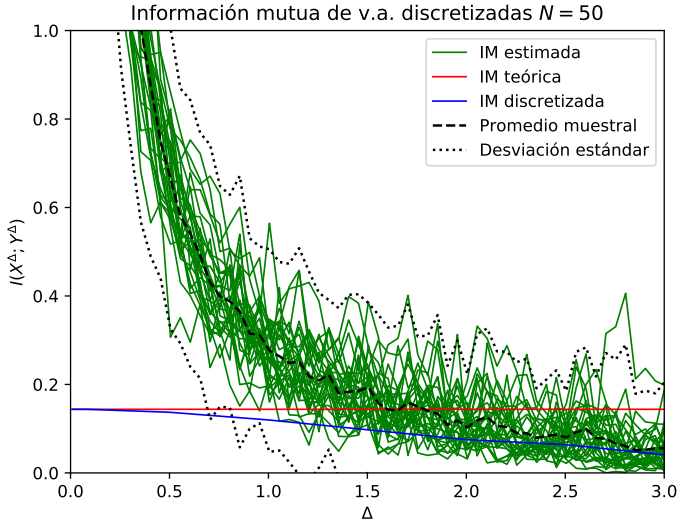

Figura 3.4: Gráfica de la información mutua entre X^Δ y Y^Δ con respecto a Δ, así como también $\hat{I}(X^\Delta; Y^\Delta)$ que es una estimación de $I(X^\Delta; Y^\Delta)$ con 30 muestras de las variables unidimensionales (X, Y), cada una de tamaño $N = 50$.

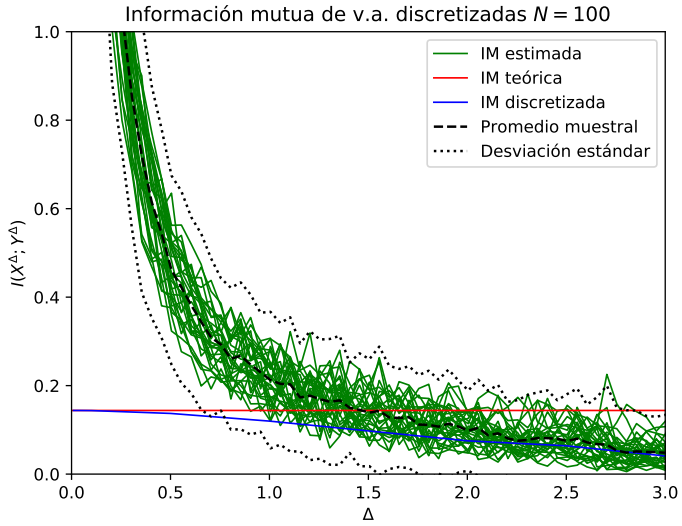

Figura 3.5: Gráfica de la información mutua entre X^Δ y Y^Δ con respecto a Δ, así como también $\hat{I}(X^\Delta; Y^\Delta)$ que es una estimación de $I(X^\Delta; Y^\Delta)$ con 30 muestras de las variables unidimensionales (X, Y), cada una de tamaño $N = 100$.

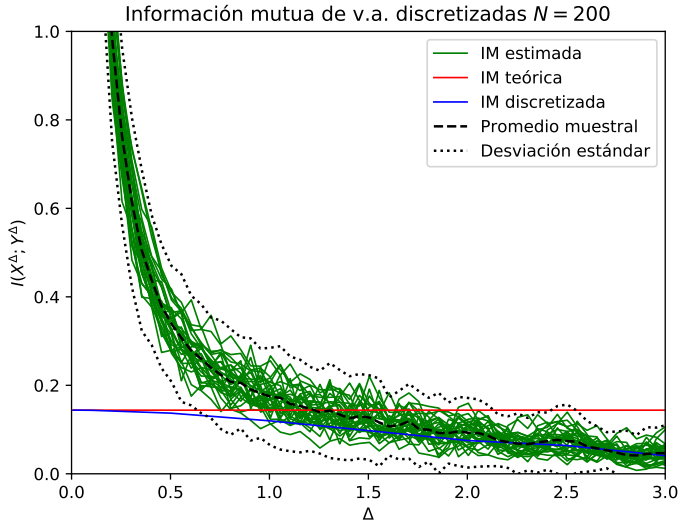

Figura 3.6: Gráfica de la información mutua entre X^Δ y Y^Δ con respecto a Δ, así como también $\hat{I}(X^\Delta; Y^\Delta)$ que es una estimación de $I(X^\Delta; Y^\Delta)$ con 30 muestras de las variables unidimensionales (X, Y), cada una de tamaño $N = 200$.

Se puede observar que al tener mayor número de muestras, las fluctuaciones de la estimación se vuelven menores. Además, el intervalo de valores de Δ para los cuales

la información mutua estimada iguala al valor real de $I(X;\,Y)$ es más pequeño que el intervalo para el caso de muestras de tamaño $N = 10$ y $N = 50$. La banda empírica se reduce y se parece más a la curva de $I(X^\Delta;\,Y^\Delta)$. Cuando el valor de Δ se vuelve más pequeño, es cada vez menos probable que más de una realización de la muestra se encuentren en el mismo intervalo $(i\Delta,\,(i+1)\Delta)$. Es por ello que la información mutua se acerca a la entropía de una variable aleatoria uniforme, que es el logaritmo de la cantidad de subintervalos en que se haya dividido el intervalo.

Lamentablemente, en la práctica no siempre se cuenta con acceso a una cantidad grande de muestras. O de ser así, no se tiene la capacidad computacional para procesar muestras lo suficientemente grandes como para que las estimaciones se parezcan a lo que realmente ocurre en la teoría. Además, es complicado tener conocimiento previo sobre el valor adecuado de Δ para estimar el valor de $I(X;Y)$.

Ahora se realizarán simulaciones similares pero aumentando la dimensionalidad de los datos. Esto con la finalidad de obtener indicios sobre la relación de la complejidad del problema con los límites fundamentales de estimación de información mutua.

Ejemplo 3.3. Sea (X, Y) v.a. $\mathcal{N}_4(\mathbf{0}, \Sigma)$ con matriz de covarianza $\Sigma = \begin{bmatrix}\Sigma_1 & \Sigma_2\\ \Sigma_2 & \Sigma_1\end{bmatrix}$, donde $\Sigma_1 = \begin{bmatrix}1 & 0{,}5\\ 0{,}5 & 1\end{bmatrix}$ y $\Sigma_2 = \rho\Sigma_1$. Realizando cálculos análogos al ejemplo anterior se tiene que

$$I(X;\,Y) = -\log(1-\rho^2).$$

Considerando $\rho = 0{,}3$, se realizan varias estimaciones del valor anterior en base a muestras de las variables X y Y, generadas mediante la librería `scipy.stats.multivariate_normal` de python.

En la Figura 3.7 se ilustra el valor real de $I(X;\,Y)$ (IM teórica); el valor de $I(X^\Delta;\,Y^\Delta)$ para distintos valores[2] de Δ (IM discretizada); y para los mismos valores de Δ, se grafica $\hat{I}(X^\Delta;\,Y^\Delta)$. Para calcular cada curva de $\hat{I}(X^\Delta;\,Y^\Delta)$, las funciones de probabilidad conjuntas y marginales han sido estimadas de acuerdo a una muestra de tamaño $N = 10$ de $(X,\,Y)$ para distintos valores de Δ (IM estimada).

Se grafican 30 curvas de información mutua estimada para generar una banda empírica que ilustre la aleatoriedad que se tiene al estimar la información mutua de esta manera y con una sola muestra. Se grafica además el promedio de las 30 curvas y una banda del promedio $\pm$ 3 desviaciones estándar.

[2]Para los valores de Δ se consideró una partición del intervalo de 0.005 a 3 con tamaño de particiones de 1/20.

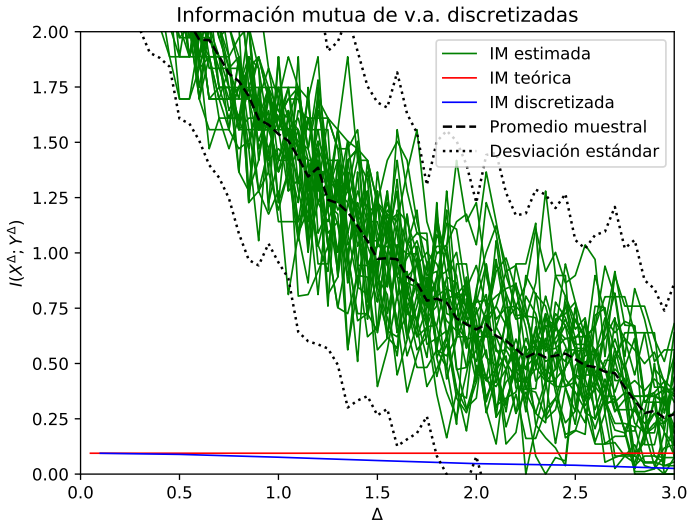

Figura 3.7: Gráfica de la información mutua entre X^Δ y Y^Δ (discretizaciones de las variables bidimensionales X y Y respectivamente) con respecto a Δ, así como también $\hat{I}(X^\Delta; Y^\Delta)$ que es una estimación de $I(X^\Delta; Y^\Delta)$ con 30 muestras de (X, Y), cada una de tamaño $N = 10$.

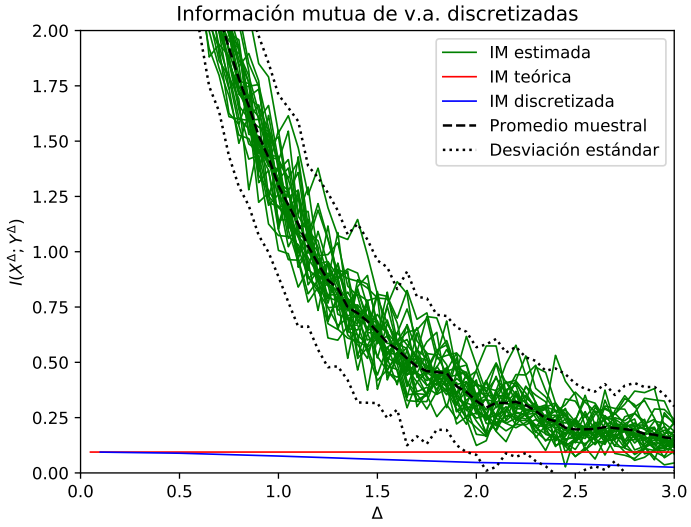

Figura 3.8: Gráfica de la información mutua entre X^Δ y Y^Δ con respecto a Δ, así como también $\hat{I}(X^\Delta; Y^\Delta)$ que es una estimación de $I(X^\Delta; Y^\Delta)$ con 30 muestras de las variables bidimensionales (X, Y), cada una de tamaño $N = 50$.

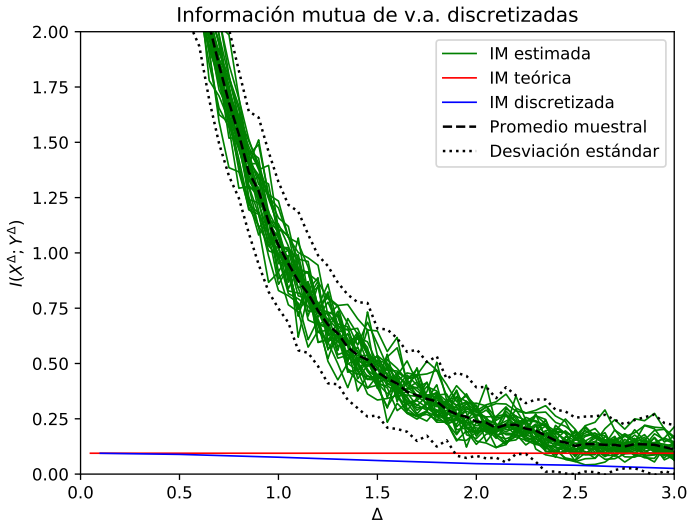

Figura 3.9: Gráfica de la información mutua entre X^Δ y Y^Δ con respecto a Δ, así como también $\hat{I}(X^\Delta; Y^\Delta)$ que es una estimación de $I(X^\Delta; Y^\Delta)$ con 30 muestras de las variables bidimensionales (X, Y), cada una de tamaño $N = 100$.

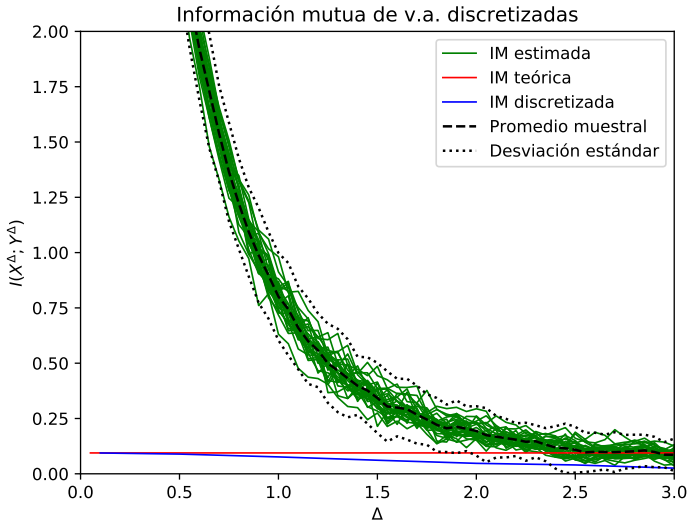

Figura 3.10: Gráfica de la información mutua entre X^Δ y Y^Δ con respecto a Δ, así como también $\hat{I}(X^\Delta; Y^\Delta)$ que es una estimación de $I(X^\Delta; Y^\Delta)$ con 30 muestras de las variables bidimensionales (X, Y), cada una de tamaño $N = 200$.

Se realizó el mismo experimento con distintos tamaños de las muestras: $N = 50, 100, 200$. En las gráficas de las Figuras 3.8, 3.9, 3.10, se aprecian los

resultados obtenidos. Nuevamente, las bandas se realizaron con 30 muestras (cada una de tamaño N).

Observamos que nuevamente las bandas empíricas se reducen con el aumento en el tamaño de la muestra. Pero a pesar de ello, los valores de Δ para los cuales se tiene una estimación correcta de $I(X;Y)$ forman un intervalo. En dicho intervalo, el aumento en la dimensionalidad de los datos no hace más sencillo obtener un solo valor que pueda funcionar para distintas muestras, tal como en el caso para una dimensión.

Así pues, estas gráficas nos ayudan a tratar de dar una explicación sobre la dificultad que conlleva adoptar esta forma de estimación de IM para llevar a cabo un análisis con la metodología de IB:

i) En el caso de los experimentos en [1] y [2], los resultados de cada capa se agruparon en 30 intervalos iguales entre -1 y 1. Es decir, para el modelo SZT se consideró $\Delta = 1/15 = 0{,}0666$, valor para el cual las simulaciones realizadas no arrojan resultados que puedan funcionar para más de una muestra.

ii) El equipo en el cual se llevaron a cabo las simulaciones cuenta con dos piezas de procesadores de 3.40GHz y una memoria RAM de 128GB. Solamente para el caso de dos dimensiones, la obtención de las gráficas de este ejemplo conllevó un tiempo computacional de 621 horas. Y aunque se cuenta con varias muestras, se tiene incertidumbre sobre una elección favorable del valor de Δ.

iii) En general, los datos que se suelen manipular en redes neuronales son de dimensionalidad alta y si en un ejemplo en apariencia sencillo se tienen dificultades, es de suponer que las dificultades aumentarán en modelos más complejos. Además de que se necesitaría de un equipo potente para tal vez poder tener varias muestras de las variables en juego.

iv) Como ya se mencionó en el Capítulo 1, es importante recordar que los conceptos de entropía e información mutua consideran los valores en promedio. Es decir que, en este contexto, no se puede decir algo meramente acertado sobre la variable por medio de una estimación de la información mutua.

3.3.2. Otros métodos de estimación de información mutua

En este trabajo solo se analiza un método de estimación de información mutua, ya que es el utilizado en los experimentos realizados por Tishby et al. (2017). Sin embargo, en años recientes se han publicado trabajos en donde se proponen distintos métodos para

realizar estimaciones de la información mutua.[3]

En [2] replican los experimentos de Tishby considerando un método basado en Kernel Density Estimation, propuesto por Kolchinsky et al. (2017). Para aplicar el método se asume que la actividad que realizan las redes neuronales en sus capas ocultas tiene una distribución de mezcla de gaussianas. La distribución que se asume tiene un parámetro de varianza, el cual juega un rol similar a la adición de ruido que se genera al discretizar las variables en subintervalos. Las conclusiones a las que llegan con este método son similares a las ya mencionadas en el método de discretización.

Otro estimador de información mutua utilizado, el cual fue propuesto por Kraskov et al. (2004), se basa en distancias entre las muestras, calculadas mediante k-nearest neighbor. Al igual que en los enfoques de discretización y de kernel density estimation, se deben elegir algunos parámetros debido a la naturaleza del estimador. El comportamiento cualitativo que los autores observan es similar al observado con los métodos mencionados anteriormente.

[3]Ver por ejemplo Chelombiev, I., Houghton, C., O'Donnell, C. (2019). *Adaptive estimators show information compression in deep neural networks.* Conference paper at International Conference on Learning Representations 2019.
Goldfeld, Z., Greenewald, K., Weed, J., Polyanskiy, Y., (2019) *Optimality of the Plug-in Estimator for Differential Entropy Estimation under Gaussian Convolutions.* IEEE International Symposium on Information Theory.

Conclusiones

En base a la teoría presentada, a los ejemplos realizados en este trabajo y a las simulaciones de la Sección 3.3, se concluye que

- Resultados contemporáneos de teoría de la información se pueden utilizar para el análisis de la generalización de algoritmos de aprendizaje. En particular pueden ayudar a explicar el porqué algunos modelos de aprendizaje máquina generalizan de una manera satisfactoria (Sección 2.3).

- Algunos autores que han estudiado el IB (por ejemplo [1], [2]), consideraron de manera no evidente los límites fundamentales en la estimación de la información mutua: la relación entre la complejidad del problema y el tamaño de la muestra. Por lo tanto sus experimentos pueden no reflejar con fidelidad lo que realmente ocurre en el plano de información, lo cual es una de las problemáticas potenciales por las cuales se ha desencadenado polémica en torno a la metodología. Se requiere de trabajo futuro para tratar de aclarar este punto.

- En los experimentos realizados (Sección 3.3), se observa que, para una sola realización, el valor de la información mutua tiene muchas fluctuaciones (por ejemplo, Figura 3.3). Más aún, la variación entre realizaciones es lo suficientemente grande como para tener incertidumbre sobre el valor de Δ adecuado para realizar la estimación.

- La metodología de IB puede servir para obtener nuevas perspectivas sobre la evolución de las técnicas estadísticas que se analicen. Sin embargo, hay que tratarla con cuidado, considerando los límites fundamentales en la estimación de la información mutua. Además, como se mostró en este trabajo su implementación y costo computacional ha sido significativo, lo cual se realizó en modelos mucho más sencillos que los que se manipulan en la práctica.

Bibliografía

[1] Shwartz-Ziv R., & Tishby N. (2017) *Opening the black box of deep neural networks via information.* CoRR, abs/1703.00810.

[2] Saxe A. M., Bansal Y., Dapello J., Advani M., Kolchinsky A., Tracey B. D., & Cox D. D. (2018) *On the information bottleneck theory of deep learning.* Conference paper at International Conference on Learning Representations 2018.

[3] Goldfeld Z., Berg E. V. D., Greenewald K., Melnyk I., Nguyen N., Kingsbury B. & Polyanskiy Y. (2018) *Estimating information flow in neural networks.* Submitted at International Conference on Learning Representations 2019.

[4] Tishby N. & Zaslavsky N. (2015) *Deep learning and the information bottleneck principle.* In IEEE Information Theory Workshop (ITW) 2015.

[5] Tishby N., Pereira F. C., & Bialek W. (1999) *The information bottleneck method*, in Proceedings of the 37-th Annual Allerton Conference on Communication, Control and Computing, 1999, pp. 368-377.

[6] Xu A. & Raginsky M. (2017) *Information-theoretic analysis of generalization capability of learning algorithms.* In Advances in Neural Information Processing Systems 2017, pp. 2524-2533.

[7] Feder M. & Merhav N. (1994) *Relations between entropy and error probability.* In IEEE Transactions on Information Theory, vol. 40, no. 1, pp. 259-266, Jan. 1994.

[8] Goldfeld Z., Greenewald K., Weed J. & Polyanskiy Y. (2019) *Optimality of the Plug-in Estimator for Differential Entropy Estimation under Gaussian Convolutions.* In IEEE International Symposium on Information Theory, July, 2019.

[9] Cover T. M., & Thomas J. A. (2012) *Elements of information theory.* John Wiley & Sons.

[10] Shalev-Shwartz S. & Ben-David S. (2014) *Understanding Machine Learning: From Theory to Algorithms.* Cambridge University Press.