

# Informe sobre la Comparación de aprendizaje e inferencia entre Deep Learning y Modelos Gráficos Probabilísticos

Bryan Rodriguez Meza  
Universidad Peruana de Ciencias Aplicadas  
Lima, Perú  
u201611137@upc.edu.pe

Edwin Alfaro Paredes  
Universidad Peruana de Ciencias Aplicadas  
Lima, Perú  
u201611810@upc.edu.pe

**Abstract**—En este informe se realiza un resumen comparativo entre los modelos gráficos probabilísticos (MGP) y el Deep learning (DL) para determinar sus diferencias tanto en el aprendizaje de estructuras, como aprendizaje de parámetros y finalmente en la manera que realizan la inferencia para lograr predicciones. Para ambos casos nos basaremos en conocimientos previos obtenidos en sesiones de clase y charlas brindadas por la Universidad, pero para datos más específicos, en MGP utilizaremos como guía el libro “*Probabilistic graphical models: principles and techniques*” de Daphne Koller y Nir Friedman. Por otra parte, para el campo del DL y redes neuronales nos basaremos en el libro “*Deep learning*” de Ian Goodfellow, Yoshua Bengio y Aaron Courville. Finalmente, este comparativo nos ayudara a resaltar las diferencias entre ambos campos y poder determinar sus utilidades en problemas reales.

**Index Terms**—Deep Learning, Modelos gráficos probabilísticos

## I. INTRODUCCIÓN

Hoy en día, la inteligencia artificial es un término que se aplica a casi todas las áreas imaginables de cualquier campo. Es un método muy versátil ya que se adapta a cualquier escenario donde se quiera aplicar y obtiene resultados deseados en casi todos los intentos realizados, y en caso no se obtuviesen se sigue intentando de diversos métodos hasta que se logra conseguir algo que pueda aportar algo relevante. Pero la inteligencia artificial no es algo que haya nacido de la noche a la mañana, o sea algo con que se presiona un botón y se obtienen resultados, ha ido evolucionando desde sus orígenes a base de esquemas teóricos o fundamentos matemáticos que no necesariamente fueron pensados para ser aplicados a la computación como los orígenes del teorema de Bayes en 1763 [1] hasta computadoras Go que vencen a jugadores profesionales humanos a base de técnicas de machine learning y búsqueda de árboles en 2016 [2]. Todo este proceso evolutivo ha llevado a la obtención de diversos resultados y contribuciones al área de inteligencia artificial, para que así esta crezca y genere más posibilidades de solucionar problemas.

En el área de la inteligencia artificial, se pueden identificar 2 modelos que son muy recurrentes en diversas soluciones. En primer lugar, tenemos los modelos gráficos probabilísticos que son un framework en donde se encuentran las distribuciones

probabilísticas de diversas variables relacionadas entre sí [3]. Por otro lado, tenemos al Deep Learning, es un método de aprendizaje automático donde las computadoras aprenden mediante la experiencia y entienden el mundo a base de la jerarquía de conceptos, donde cada concepto está definido mediante su relación con conceptos más simples. Este acercamiento evita la necesidad de ayuda humana para especificar lo que necesita la computadora [4].

A continuación, en la sección 2 se mencionará con mayor detalle la literatura utilizada para este informe, mientras que en la sección 3 se hablará con más detalle los puntos donde compararemos al Deep Learning con los Modelos gráficos probabilísticos describiendo como se realiza el aprendizaje de las estructuras, seguido con el aprendizaje de parámetros para finalizar con el proceso de la inferencia de resultados. Finalmente, en la sección 4 se presentarán las conclusiones sobre el informe comparativo en general.

## II. MÉTODOS Y PROCEDIMIENTOS

El propósito de este informe es recolectar la información más importante sobre los modelos gráficos probabilísticos y el Deep learning. Esto se logrará mediante 2 libros: para los MGP, se utilizará “*Probabilistic graphical models: principles and techniques*” mientras que, para DL, “*Deep learning*”. Cabe destacar que también se utilizaran otros artículos, libros o publicaciones para información adicional pero la principal fuente de este resumen son los 2 libros mencionados previamente y también la información recolectada mediante charlas o workshops asistidos en la Universidad.

## III. RESULTADOS

### A. Aprendizaje de estructuras

1) *Modelos gráficos probabilísticos*: La estructura de los modelos gráficos probabilísticos consiste en conjuntos de variables aleatorias. Cada valor que pueda tomar cualquiera de estas variables defina una propiedad del mundo o escenario donde se encuentra. En el libro [3] se propone un ejemplo donde la influenza pueda ser una variable en un dominio. Esta puede tomar 2 valores: presente o ausente. A su vez, se encuentra una variable Fiebre, que también puede tomar

otros valores que pueden llegar a ser incluso continuos. Estas variables se toman en cuenta y son importantes para lograr tomar una decisión. Para la toma de esta, se necesita construir una distribución conjunta de acuerdo con las relaciones entre variables para tener una distribución resultante y así poder realizar la toma de la decisión. Este es un ejemplo pequeño, a comparación de la realidad donde hay datos con cientos o miles de variables que pueden tomar miles de valores diferentes, elevando la complejidad del problema a incluso convertirlo en algo intratable.

A continuación, se verá un grafo acíclico dirigido (DAG) que servirá como un ejemplo de MGP. Cabe resaltar que hay otros tipos de grafos, como las redes de Márkov que utilizan grafos no dirigidos.

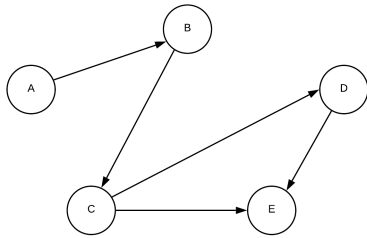


Fig. 1. Una red bayesiana es un claro ejemplo de MGP

La figura 1 nos muestra una estructura típica de un MGP, en este caso mediante una red bayesiana. Para comenzar, tenemos los nodos A, B, C, D, E; y estos se conectan mediante aristas de una manera específica sin generar ciclos. Podemos clasificar estos nodos en 2 tipos: el primer tipo son los nodos sin padres, en este caso el nodo A no tiene padre, lo que significa que es independiente. El segundo tipo de nodos son los que cuentan con uno o más padres. Aquí se encuentran los nodos B, C, D. Esto significa que, los nodos que conectan hacia estos tienen un efecto directo en los nodos hijos.

Con esta estructura, se pueden comenzar a calcular los parámetros necesarios para realizar las inferencias posteriormente.

2) *Deep Learning*: La estructura del Deep Learning, en sí de las redes neuronales, consiste en neuronas de entrada conectadas a múltiples capas que se encuentran conectadas de manera horizontal hasta llegar a una capa de salida. Cada una de estas neuronas se les llama perceptrones, y al estar enfocados en Deep Learning, se tienen múltiples capas llamándolo así una red neuronal profunda o Perceptrón Multicapa (MLP). Cada capa consta de funciones de activación que surgen desde las más simples cuando están más cerca de las entradas hasta las más complejas que ya colindan con los perceptrones de salida. Cada una de estas conexiones entre perceptrones en capas diferentes tienen un peso, el cual se va calculando y recalculando en cada iteración de entrenamiento del MLP.

Un ejemplo clásico de los MLP son las redes neuronales convolucionales. Estas reciben imágenes de entrada, pero como se mencionó anteriormente, se empieza desde lo más simple, es decir, componentes pequeños, y progresivamente

se van juntando para formar componentes más complejos [4]. En el caso presente, las entradas son los píxeles de la imagen, cada capa oculta junta los componentes pequeños hasta llegar a la capa de salida que puede lograr conseguir la imagen y, por ejemplo, lograr clasificar un dígito si se quisiera dar un ejemplo específico.

A continuación, se verá cómo es la estructura de un MLP. El ejemplo es un MLP básico, hay casos donde se cuentan con muchas más capas, muchas más entradas y muchos más perceptrones.

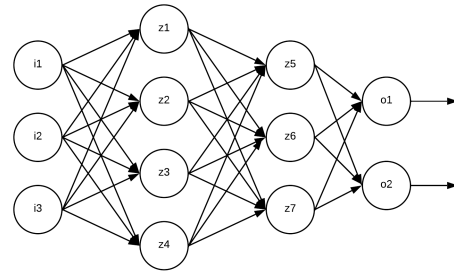


Fig. 2. Un ejemplo de un MLP

La figura 2 nos muestra las partes del MLP mencionado anteriormente. De i1 a i3 se tienen las entradas, mientras que de z1 a z4 se tiene la primera capa oculta y de z5 a z7 se tiene la segunda. Finalmente, o1 y o2 son las salidas donde se obtienen las probabilidades que se escoja una salida o la otra.

#### B. Aprendizaje de parámetros

1) *Modelos gráficos probabilísticos*: Los parámetros para los modelos gráficos probabilísticos son las distribuciones condicionales o marginales de cada nodo, dependiendo si son independientes o cuentan con padres. Estas distribuciones se almacenan en factores. Estos nos permiten realizar operaciones entre las distribuciones para hallar las probabilidades totales y lograr realizar una inferencia teniendo un nuevo dataset como prueba después de haber entrenado el modelo. Para calcular estas distribuciones se necesitan la cantidad de instancias de los datos y las veces que se repiten ciertos criterios.

A continuación, se observará un factor, o tabla de probabilidades condicionales (CPT) donde B puede tomar como valores 0, 1, 2 y A puede tomar 0, 1; y A es padre de B.

TABLE I  
CPT DE  $P(B|A)$

B	A	$P(B A)$
0	0	0.3
1	0	0.3
2	0	0.4
0	1	0.1
1	1	0.6
2	1	0.3

En la Tabla 1 se observa cómo se ve un CPT. Se aprecia que hay 2 distribuciones y que cada una de estas suma 1. Para

un cálculo de parámetros cuando hay data faltante, se aplica el alfa de Dirichlet para que la propiedad de que todas las distribuciones sumen 1 se cumpla para cualquier caso. Este alfa puede ser cualquier valor deseado.

2) *Deep Learning*: Los parámetros para el Deep Learning se calculan con la sumatoria de las multiplicaciones de los perceptrones con los pesos que salen de él, y así se halla el valor de un nuevo perceptrón que está en la capa siguiente [4]. A esta sumatoria, se le aplica una función de activación no lineal, como por ejemplo la Sigmoide, la TANH o la ReLU, lo cuales sirven para poder lograr separaciones no lineales. A estas funciones también se les agrega la neurona Bias que sirve para otorgarle mayor flexibilidad a los resultados.

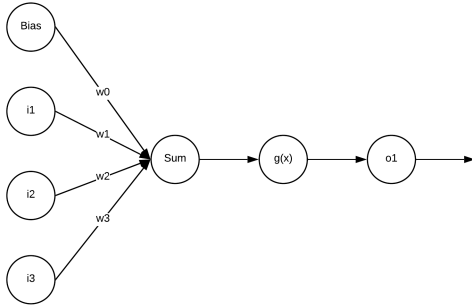


Fig. 3. Funcionamiento del perceptrón

A continuación, se vera la fórmula con la cual se calcula el valor de las neuronas de las capas posteriores:

$$\hat{o} = g \left( \sum_{j=1}^m i_j w_j \right) \quad (1)$$

Dicho proceso se repite por cada neurona de cada capa posterior. El proceso de la aplicación de (1) para cada neurona desde la capa de entrada hasta la capa de salida tiene como nombre Forward Propagation, el cual se puede definir como el producto punto entre los perceptrones de la capa n-1 contra los pesos que salen de dicha neurona hasta una neurona ubicada en la capa n. En resumen, cada perceptrón se calcula de la siguiente manera: se realiza un producto punto y ese resultado pasa por la función de activación.

### C. Inferencia

1) *Modelos gráficos probabilísticos*: Una vez se tienen obtenidas todas las tablas de probabilidad condicional, se proceden a multiplicar según el grafo para obtener la distribución conjunta del mismo. Con esto se puede comenzar a realizar las inferencias donde se tienen valores evidencia, es decir, valores que ya se conocen; y también se tienen los valores deseados. Estos se van a hallar a base de las evidencias y la distribución conjunta del grafo.

Lo que va a lograr la inferencia es, dado un conjunto de evidencias, retornarnos el valor más probable que se pueda obtener con esas condiciones dadas previamente. Lo que podemos hallar con esta inferencia, y a base del grafo son

las relaciones de las variables y esto se puede interpretar para tener una noción básica de por qué una variable tiene relevancia con el resultado de otra.

2) *Deep Learning*: Una vez se completa con el primer Forward Propagation, se comparan los resultados de salida obtenidos con los resultados de salida esperados y comparando ambos resultados se obtiene el error cuadrático medio, el cual se halla con la suma de la elevación del resultado deseado con el obtenido, multiplicado por el número de salidas (n) elevado a la -1 o 1/n.

$$E_{total} = \sum \frac{1}{n} (\text{target} - \text{output})^2 \quad (2)$$

Luego de obtener el error total mediante (2), se procede con el Backward Propagation. Lo que hace esto es actualizar los pesos de la red y hallar el error cometido por cada neurona. Esto se logra mediante las derivadas parciales de cada elemento que une un perceptrón con otro. Esto se realiza hasta actualizar los pesos iniciales de las primeras aristas. Lo que no se actualiza son los valores de entrada ya que son datos estáticos.

Ambos pasos, el Forward y Backward propagation se van a realizar las veces que sean necesarias para minimizar el error y lograr obtener resultados muy cercanos a las salidas esperadas.

## IV. CONCLUSIONES

En primer lugar, las estructuras de los MGP indican las relaciones entre las variables, es decir si una o varias afectan a una variable, o si alguna es un suceso independiente. Mientras que en el caso de DL no es así. Todos los perceptrones se conectan entre sí y tienen pesos en sus relaciones que se van reajustando durante las diversas operaciones. De los MGP obtenemos información más interpretable de las relaciones que de las redes neuronales en DL.

En segundo lugar, los parámetros en los MGP son un poco más entendibles y fáciles de calcular que en el caso de DL. En el primer caso, con funciones de conteo y condiciones se obtienen casi todos los parámetros para los modelos, mientras que en el segundo caso se requieren aplicar transformaciones de los productos de los pesos a través de funciones no lineales para lograr obtener clasificadores robustos.

En tercer lugar, las inferencias de los MGP logran brindar información más entendible para los agentes humanos que los DL. Si bien, se pueden obtener mejores resultados en diversos casos en DL, la información que se obtiene de estos puede ser que no aporte mucho al entendimiento del problema para las personas.

Finalmente, no necesariamente uno de los dos siempre va a ser mejor que el otro. Ambos, tanto los MGP, como las redes neuronales en DL, tienen sus usos específicos y campos donde resaltan. El modelo más recomendable para utilizar dependerá de la naturaleza del problema y lo que se necesite recolectar o descubrir de los datos en cuestión.

## REFERENCES

- [1] T. Bayes, "An essay towards solving a problem in the doctrine of chances. 1763." *MD computing: computers in medical practice*, vol. 8, no. 3, p. 157, 1991.
- [2] B. News, "Google achieves AI 'breakthrough' by beating Go champion," *BBC News*, Jan 2016. [Online]. Available: <https://www.bbc.com/news/technology-35420579>
- [3] M. I. Jordan and Y. Weiss, "Graphical models: Probabilistic inference," *The handbook of brain theory and neural networks*, pp. 490–496, 2002.
- [4] I. Goodfellow, Y. Bengio, and A. Courville, *Deep Learning*. MIT Press, 2016, <http://www.deeplearningbook.org>.