JOURNAL BOLIVIANO DE CIENCIAS – Vol. 17 – Número 51
ISSN: 2075-8944
Universidad Privada del Valle – Bolivia
https://doi.org/10.52428/20758944.v17i51.110

Fecha de Recepción: 05/09/2021
Fecha de Aprobación: 01/12/2021
Fecha de Publicación: 10/12/2021

ARTÍCULO DE REVISIÓN BIBLIOGRÁFICA

Modelos Generativos en el Aprendizaje Automático y su aplicación a la generación de Imágenes Digitales

Generative Models in Machine Learning and their Applications to Digital Image Generation

Oscar Contreras Carrasco[1]

[1] Docente de la carrera de Ingeniería de Sistemas Informáticos, Univalle Cochabamba. ocontrerasc@univalle.edu

RESUMEN

En el ámbito del Aprendizaje Automático se identifican dos tipos de algoritmos desde el punto de vista de la naturaleza de las salidas que estos proporcionan. Los modelos discriminativos asocian un dato proporcionado a una respuesta, mientras que los modelos generativos generan nuevos datos en base a una distribución probabilística de respuestas o variables latentes. Por otra parte, en años recientes, se han dado importantes avances en el *Deep Learning,* que es el estudio de las redes neuronales profundas. En este contexto, las redes convolucionales han ganado considerable terreno en diversas tareas relacionadas al análisis de imágenes. Entre las varias aplicaciones de las redes convolucionales se mencionan: clasificación de imágenes, detección de objetos, segmentación de instancias, reconocimiento facial, entre otros varios. Sin embargo, en el ámbito del *Deep Learning,* no solamente se han tenido importantes avances con relación a los puntos aquí mencionados, sino también en cuanto a la habilidad de los modelos en generar nuevas imágenes. De ese modo, hoy en día se tiene a disposición una variedad de modelos generativos para diversos propósitos, con aplicaciones tales como la creación de imágenes faciales de personas que no existen en la realidad. Es así que el propósito del presente artículo es realizar un análisis de diversos modelos generativos para imágenes digitales, así como de las bases teóricas que sustentan a los modelos generativos en el *Deep Learning*. El enfoque de análisis se centra en

ISSN: 2075-8944
Universidad Privada del Valle – Bolivia
https://doi.org/10.52428/20758944.v17i51.110
dos modelos esenciales: Las redes adversarias generativas (GAN) y los autoencoders variacionales (VAE).

Palabras clave: *Deep Learning*. GAN. Modelos generativos. Redes convolucionales. Redes neuronales. VAE.

ABSTRACT

Within the Machine Learning field, two types of algorithms can be distinguished if the nature of their outputs as a perspective is acknowledged. Discriminative and generative models associate data with a response and create new data based on a probabilistic distribution of latent variables, respectively. In recent years, significant progress in the Deep Learning subject has been published, which is the study of deep neural networks. Hence, convolutional neural networks have gained significant territory in different tasks regarding image analysis processing. Among the applications of convolutional neural networks, a few of them are Image classification, object detection, instance segmentation, and facial recognition, among others. However, the field of Deep Learning has seen progress not just in these areas but also in the ability of models to generate new images. Thus, a wide variety of generative models for different purposes has been developed; facial images generation of people who do not exist in real life is an example of the latter. Thereupon, this article aims to analyze different generative models for digital image processing and the theoretical aspects that define the generative models in the field of Deep Learning. Twofold essential models are developed in this article: Generative Adversarial Networks (GAN) and Variational Autoencoders (VAE).

Keywords: Neural networks. Deep Learning. Generative models. Convolutional neural networks. GAN. VAE.

1. INTRODUCCIÓN

El aprendizaje automático (*Machine Learning*) es la rama de la Inteligencia Artificial en la cual un modelo adquiere la capacidad de aprender y generar conocimiento en base a datos que le son proporcionados (Janiesch et al., 2021). La naturaleza de estos datos está directamente ligada con el problema que se busca resolver. Por ejemplo, en un sistema de reconocimiento facial, la entrada se ha de constituir en la imagen facial de un individuo, y lo

JOURNAL BOLIVIANO DE CIENCIAS – Vol. 17 – Número 51
ISSN: 2075-8944
Universidad Privada del Valle – Bolivia
https://doi.org/10.52428/20758944.v17i51.110
que se busca, es que luego de una fase de entrenamiento, el modelo implementado tenga la capacidad de reconocer de manera autónoma dicha imagen facial (M. Wang & Deng, 2021). Asimismo, si se desea construir un sistema que tenga la capacidad de clasificar objetos pertenecientes a diversas categorías, la entrada será una imagen de la instancia respectiva, mientras que la salida será la asociación de dicha imagen a una clase determinada (Kotsiantis et al., 2006). Mayores detalles sobre clasificación se muestran en la Figura 1:

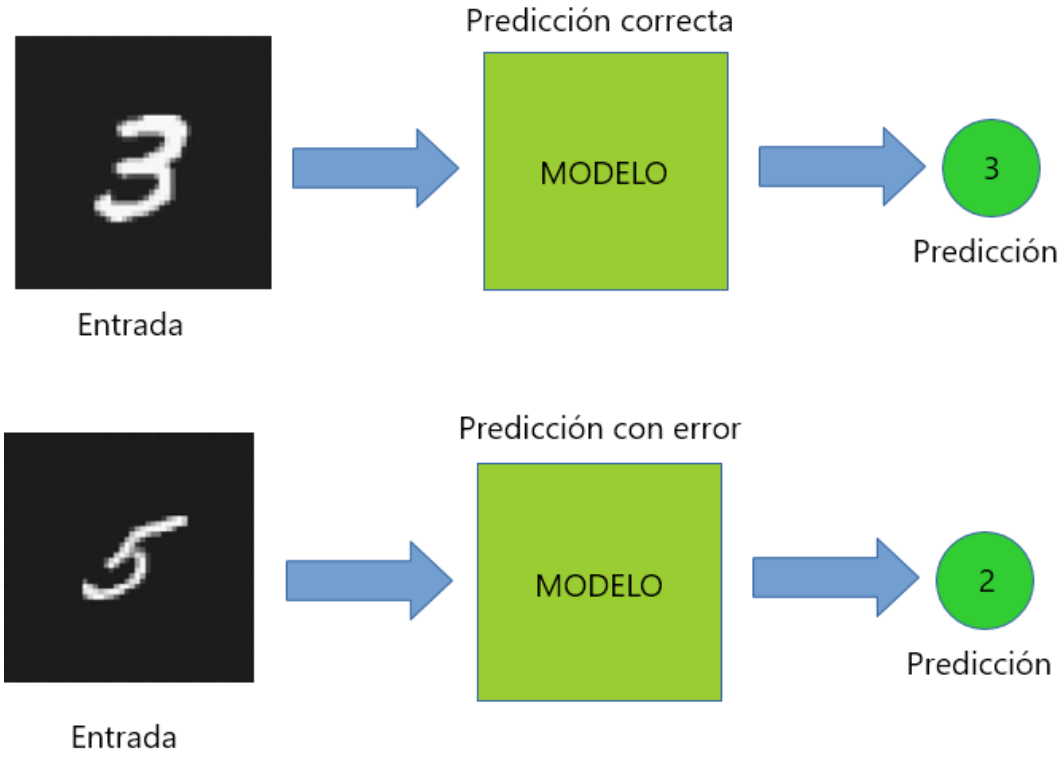

Figura 1. Ejemplos de clasificación correcta e incorrecta de un modelo predictivo
Fuente: Elaboración propia, 2021.

Aquí, se tiene el caso de un modelo clasificador de imágenes manuscritas del *dataset* MNIST (Deng, 2012). La tarea principal que debe cumplir tal modelo es asociar correctamente la imagen de entrada a alguna de las 10 categorías posibles en el conjunto de datos (0 al 9). En el primer caso, la imagen de un número 3 manuscrito es asociada correctamente a su correspondiente categoría. Sin embargo, en el segundo ejemplo, la imagen del número 5 es incorrectamente clasificada como un 2. De esta manera, se tendrá una diferencia entre la respuesta correcta y la predicción realizada por el modelo. Se define este tipo de diferencias como *errores*. A fin de poder minimizar el grado de error, primeramente se debe someter el

ISSN: 2075-8944
Universidad Privada del Valle – Bolivia
https://doi.org/10.52428/20758944.v17i51.110
modelo a un proceso de entrenamiento en el cual se hará uso de diversos métodos de optimización, tales como el *gradiente descendente* (gradient descent) (Ruder, 2017).

En la medida en que se entrena el modelo, se espera que el margen de error resultante de asociar cada dato a su categoría correcta vaya decreciendo paulatinamente. Habrá un momento en que dicho margen de error tendrá el mínimo valor posible. Se dirá entonces, que el modelo está en la capacidad de realizar nuevas predicciones de manera eficaz.

Como el propósito principal está en asociar datos a una categoría determinada, se dirá que el modelo anteriormente descrito es un *modelo discriminativo* (Ng & Jordan, 2001). Pero ¿qué tal si más bien lo que se desea lograr es lo inverso? Dada la respuesta correcta, se desea hacer que el modelo genere el dato. Es decir, una imagen entera que le corresponda a la categoría presentada. En este caso, se estaría hablando de un *modelo generativo* (Ng & Jordan, 2001), ya que este tendrá la posibilidad de generar nuevos datos en base a valores consignados a la respuesta misma o a una distribución de variables latentes.

Conviene destacar que los modelos discriminativos realizan inferencias en base a datos proporcionados. Esta característica de dichos modelos se vincula a los ámbitos de la regresión y la clasificación en el aprendizaje supervisado. Existe una amplia variedad de modelos con estas capacidades. Sin embargo, para efectos del presente análisis, se hará énfasis exclusivamente en las redes neuronales.

Tanto los modelos discriminativos como los generativos, requieren datos para poder realizar inferencias por sí mismos luego del entrenamiento. A fin de asegurar un adecuado aprovechamiento de estos datos, será necesario realizar un preprocesamiento de estos, lo cual estará en estrecha dependencia del problema que buscamos resolver y también de la arquitectura del modelo utilizado.

JOURNAL BOLIVIANO DE CIENCIAS – Vol. 17 – Número 51
ISSN: 2075-8944
Universidad Privada del Valle – Bolivia
https://doi.org/10.52428/20758944.v17i51.110

2. DESARROLLO Y ANÁLISIS

2.1. Características de los modelos generativos

Se ha descrito con anterioridad que los modelos de aprendizaje automático pueden ser de naturaleza *discriminativa* o *generativa*. Se desarrollarán ahora en mayor profundidad las características de los modelos de naturaleza generativa.

Como se indicó anteriormente, una de las principales características de los modelos generativos es la capacidad de crear nuevos datos en base a cierta información proporcionada (Gm et al., 2020), por ejemplo, una distribución probabilística latente. De esta manera, se identificará como **x** al dato ingresado al modelo, también entendido como una *instancia* del conjunto de datos. Asimismo, se denominará **y** a la respuesta de dicho modelo. Los modelos discriminativos se asocian una distribución probabilística de las siguientes características (Bishop, 2006):

$$p(C_k \mid \mathbf{x}) \text{ o bien } p(y = C_k \mid \mathbf{x})$$

Donde C_k es la clase a la cual pertenece la instancia **x**. Claramente, podemos ver que la tarea principal reside en asociar dicha instancia a la categoría a la cual esta corresponde. Esto es lo que ocurre en problemas de clasificación tradicional, donde por ejemplo **x** representa la imagen de un objeto, mientras que C_k es la categoría a la cual dicho objeto pertenece. En la Figura 2, se ilustra dicho escenario para mayor claridad:

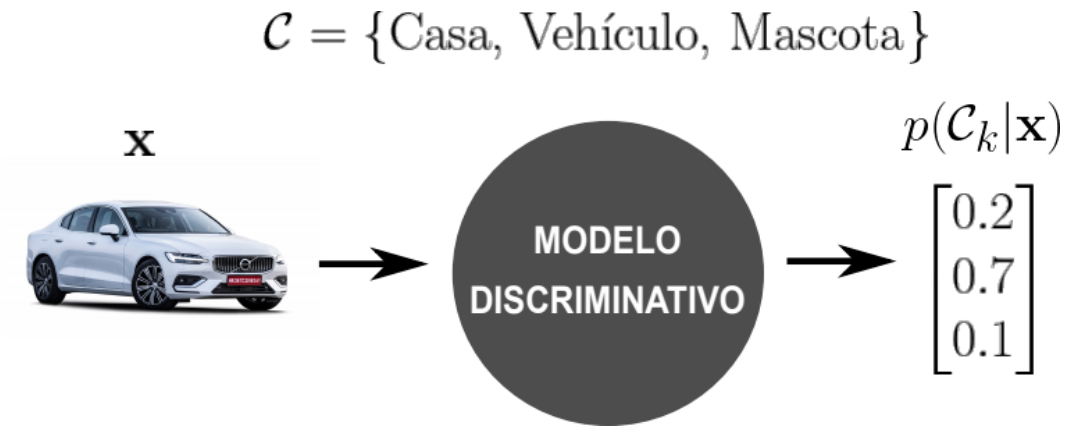

Figura 2. Estructura de un modelo discriminativo
Fuente: Elaboración propia, 2021.

JOURNAL BOLIVIANO DE CIENCIAS – Vol. 17 – Número 51
ISSN: 2075-8944
Universidad Privada del Valle – Bolivia
https://doi.org/10.52428/20758944.v17i51.110

En este ejemplo, **x** es la imagen de un vehículo, la cual es alimentada a un modelo discriminativo. Este a su vez, genera como resultado una distribución probabilística donde el primer elemento de valor 0,2 identifica la probabilidad de que **x** sea una casa. De la misma forma, los elementos de la segunda y tercera posición del vector resultante son las probabilidades de que **x** sea un vehículo y una mascota, respectivamente. Como es posible apreciar, la mayor probabilidad de 0,7 le corresponde a la clase *Vehículo*. Ahora bien, ¿cómo es que los modelos discriminativos adquieren esta capacidad de consignar mayor probabilidad a la categoría correcta? Para que esto sea posible, es necesario que de por medio exista un proceso de *entrenamiento del modelo*. Dicho proceso consistirá en realizar un ajuste de los parámetros de este. Por ejemplo, si el modelo es una función de probabilidad Gaussiana, habrá que encontrar los valores óptimos de la media y la covarianza de esta distribución. Si el modelo pertenece a la familia lineal, entonces habrá que realizar un ajuste de los pesos (*weights*) y el *bias* del mismo, donde la probabilidad condicional $p(C_k|\mathbf{x})$ puede ser expresada de la siguiente manera:

$$p(\mathrm{C}_k \mid \mathbf{x}) = g(z) = g(\mathbf{w}^{\bullet}\, \mathbf{x} + b) \qquad (1)$$

Donde **w** es el grupo de pesos, también conocidos como *pesos sinápticos*, *b* es el *bias* y *g* es una función de activación que transforma la función lineal del argumento en una predicción.

Ahora bien, la principal característica de los modelos generativos es que estos modelan la distribución de probabilidad **p(x|y)** en lugar de **p(y|x)** como en el caso de los discriminativos. Esto es, dada la respuesta, generan la instancia **x** a la cual corresponde dicha respuesta. Este mismo principio puede ser extendido a los modelos no supervisados[1], donde la respuesta es desconocida y por ende se afirmará que es más bien una variable latente, de ahora en adelante **z**. De esta manera, se dirá que los modelos generativos no supervisados modelan **p(x|z)**.

1 Un tipo de modelos de Machine Learning que infieren la estructura interna de los datos. En estos datos, la respuesta asociada a cada instancia es desconocida y buscamos que sea inferida por el modelo mismo. Por ejemplo: K-Means.

JOURNAL BOLIVIANO DE CIENCIAS – Vol. 17 – Número 51
ISSN: 2075-8944
Universidad Privada del Valle – Bolivia
https://doi.org/10.52428/20758944.v17i51.110

Una particularidad de los modelos generativos es que estos también pueden realizar predicciones basadas en la distribución de probabilidad condicional **p(y|x)** (o **p(z|x)**). Para tal efecto, usan el Teorema de Bayes (Ng & Jordan, 2001):

$$p(\mathbf{y} \mid \mathbf{x}) = \frac{p(\mathbf{x} \mid \mathbf{y})\, p(\mathbf{y})}{p(\mathbf{x})} \tag{2}$$

Aquí, **p(x|y)** es la función probabilística de *verosimilitud*, **p(y)** es la distribución de probabilidad a priori o *prior*, mientras que **p(x)** es la *evidencia*. Este razonamiento también puede ser extendido al caso de los modelos no supervisados:

$$p(\mathbf{z} \mid \mathbf{x}) = \frac{p(\mathbf{x} \mid \mathbf{z})\, p(\mathbf{z})}{p(\mathbf{x})} \tag{3}$$

La estrategia de modelado de la probabilidad **p(x|y)** o **p(x|z)** depende de la naturaleza del modelo, así como del dominio de aplicación. Más adelante, se realizará una descripción más pormenorizada de los modelos generativos para la creación de imágenes digitales, así como sus ámbitos aplicativos y desempeño en determinadas tareas.

2.2. Redes convolucionales

En el ámbito del Aprendizaje Automático o *Machine Learning*, una variedad de desarrollos interesantes ha tenido lugar en la última década donde las redes neuronales han venido jugando un papel protagónico. Sin embargo, más allá de las capacidades algorítmicas, el potenciamiento de las capacidades de cómputo, principalmente a través de unidades de procesamiento gráfico (Jeon et al., 2021), fue otro de los factores determinantes en los recientes avances del *Deep Learning*, una rama del Aprendizaje Automático que se dedica al estudio de las redes neuronales profundas (Janiesch et al., 2021).

Ya en 2012, se desarrolla una red convolucional capaz de sobrepasar las capacidades de modelos que hasta ese entonces eran conocidos. Se trataba de AlexNet (Krizhevsky et al., 2012), una red convolucional desarrollada por Alex Krizhevsky que además utilizaba una unidad de procesamiento gráfico para el entrenamiento. Esto le permitía importantes ganancias en cuanto al tiempo de optimización. Si bien el uso del GPU para entrenar modelos no era para entonces un concepto nuevo, los resultados obtenidos en la competencia

ISSN: 2075-8944
Universidad Privada del Valle – Bolivia
https://doi.org/10.52428/20758944.v17i51.110

ImageNet[2], a través de AlexNet fue algo que motivó un mayor interés de parte de la comunidad científica en torno a las redes convolucionales.

Pronto surgirían nuevos modelos de similares características y es así que, en 2015, el Grupo de Geometría Visual de la Universidad de Oxford publica las arquitecturas de redes VGG con considerables mejoras frente a modelos anteriores (Simonyan & Zisserman, 2015). En ese mismo año, el equipo de Kaiming He publica RESNET (He et al., 2015), un grupo de redes convolucionales capaces de lograr niveles muy altos de desempeño en tareas de clasificación complejas.

En el estudio del *Deep Learning,* las redes convolucionales juegan un papel instrumental pues demuestran resultados sobresalientes en la extracción de características relevantes a partir de imágenes digitales (Li et al., 2020). Es conveniente entender una imagen digital como un tensor cuyas dimensiones son el ancho, la altura y el número de canales de color que usan. Por ejemplo, una imagen a color de 100 píxeles de ancho y altura tendrá como dimensiones 100x100x3, con tres canales para cada color primario (rojo, verde y azul). Por otra parte, la razón por la cual se las conoce como redes convolucionales es porque emplean el operador de la convolución 2D para extraer información en la forma de características o *features* de una imagen de entrada. Matemáticamente, el operador de convolución para imágenes está definido por (I. Goodfellow et al., 2016):

$$C(i, j) = \sum_m \sum_n I(i-m, j-n)K(m,n) \tag{4}$$

Donde **C (*i, j*)** es la salida de la convolución en la fila *i* y columna *j*. *I* es la entrada de la convolución, mientras que *K* es un kernel convolutivo. Los kernels son matrices pequeñas que permiten extraer las características de una imagen determinada, y en las redes convolucionales, almacenan los parámetros del modelo (I. Goodfellow et al., 2016). Por convención general, estas tienen un número impar de filas y columnas. Una variante del anterior operador es la *correlación cruzada* (cross-correlation) que es usualmente implementada por los frameworks modernos de Deep Learning (I. Goodfellow et al., 2016):

[2] Competencia que tiene lugar periódicamente y busca establecer evaluaciones comparativas entre diversos modelos para resolver problemas de Visión Artificial con imágenes (Simonyan & Zisserman, 2015).

JOURNAL BOLIVIANO DE CIENCIAS – Vol. 17 – Número 51
ISSN: 2075-8944
Universidad Privada del Valle – Bolivia
https://doi.org/10.52428/20758944.v17i51.110

$$C(i,j) = \sum_m \sum_n I(i+m, j+n) K(m,n) \quad (5)$$

La razón principal por la que se usa la correlación cruzada es por temas de simplicidad en la implementación de los módulos programáticos para el entrenamiento de las redes convolucionales. Es posible reflejar de manera ilustrativa la operación de la convolución, de la siguiente manera:

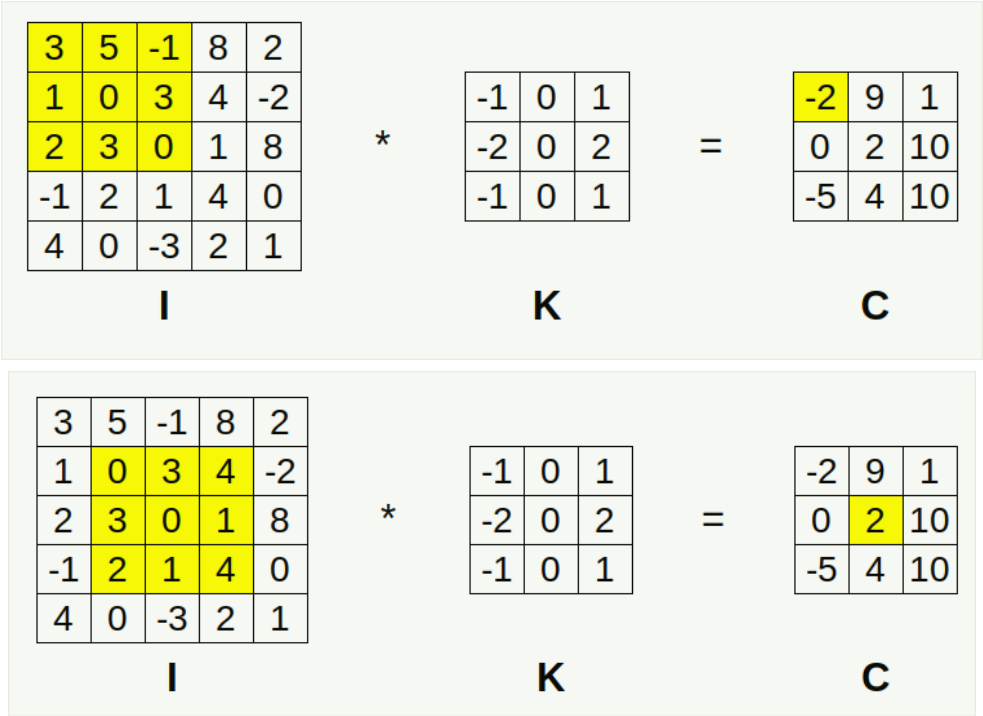

Figura 3. Proceso del cálculo de la convolución

Fuente: Elaboración propia, 2021.

En la Figura 3, **I** es la imagen de entrada, **K** es el kernel convolutivo, mientras que **C** es el resultado de la operación de convolución. Como es posible advertir, las celdas de la matriz **C** resultan de multiplicar cada elemento de **I** en el vecindario de color amarillo por las celdas correspondientes en el kernel **K**. Estos productos se totalizan para dar como resultado el primer elemento en **C**. La misma hermenéutica es aplicable a los otros elementos de ambas matrices. Las dimensiones de la salida de la convolución dependen de las dimensiones de la entrada, así como de la forma en que se aplican las convoluciones por elemento. De hecho, las dimensiones de la salida pueden variar en función al tamaño del kernel, así como el paso

JOURNAL BOLIVIANO DE CIENCIAS – Vol. 17 – Número 51
ISSN: 2075-8944
Universidad Privada del Valle – Bolivia
https://doi.org/10.52428/20758944.v17i51.110
(*stride*) de cada convolución y el uso de técnicas tales como el *padding*[3] para preservar el tamaño de la salida de la operación (O'Shea & Nash, 2015).

Los elementos de **K** forman parte del conjunto de parámetros optimizables de toda la red convolucional. De hecho, **K** contiene los pesos sinápticos del modelo. En este sentido, al entrenar la red, se busca que cada *kernel* tenga la capacidad de detectar una variedad de características lo más rica posible. En la Figura 4, es posible apreciar un esquema de los componentes esenciales de una red convolucional:

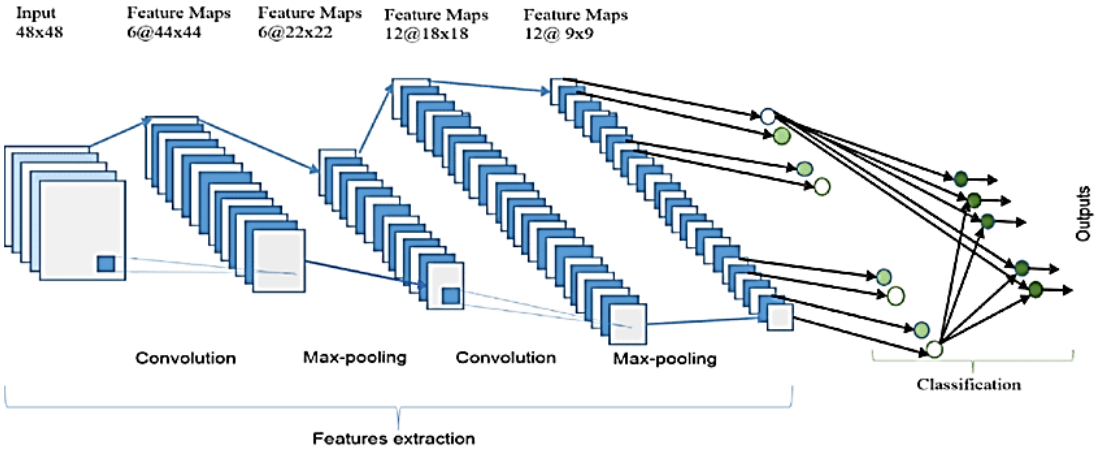

Figura 4. Elementos de una red convolucional
Fuente: A State-of-the-Art Survey on Deep Learning Theory and Architectures (Alom et al., 2019).

Entre los principales elementos de una red convolucional, es posible destacar (Khan et al., 2020):

- **Capa de extracción de características (feature extraction):** Consta de una serie consecutiva de capas convolucionales que se orientan a la extracción de datos relevantes de la imagen de entrada. Las capas tempranas de la red convolucional típicamente incluyen información muy general (Yamashita et al., 2018), tales como contornos y formas básicas, mientras que las capas más avanzadas obtienen detalles cada vez más finos. Es por lo que arquitecturas convolucionales de mayor

3 Es una técnica que agrega píxeles adicionales a los costados, así como en la parte superior e inferior de la entrada, a fin de hacer que la salida tenga las mismas dimensiones que la entrada de la convolución.

ISSN: 2075-8944
Universidad Privada del Valle – Bolivia
https://doi.org/10.52428/20758944.v17i51.110

profundidad son preferibles, ya que estas tienden a capturar detalles más ricos a partir de los datos de entrada. Dentro de la capa de extracción de características distinguimos dos tipos de elementos:

- **Convoluciones y *feature maps*:** Se encargan de la extracción de características de la entrada. Según lo indicado anteriormente, estas capas se componen de una entrada, un *kernel* convolutivo y un *feature map* que es la salida de cada convolución. Se tendrán tantos *feature maps* como *kernels* convolutivos se usan para extraer información. Por otra parte, las dimensiones de los *feature maps* están en estrecha relación con las dimensiones mismas de la entrada y del *kernel* convolutivo empleado. Se debe recordar también que los elementos de cada *kernel* son parámetros optimizables de la red (I. Goodfellow et al., 2016).
- ***Pooling:*** Luego de cada convolución y con carácter opcional, es posible adicionar capas de *pooling,* cuyo objetivo es el de disminuir el tamaño de la salida y, por ende, reducir el número total de parámetros del modelo, lo que se traduce también en una mayor facilidad de cómputo, así como una menor propensión al sobreajuste u *overfitting*[4]. Existen diversos tipos de pooling en redes convolucionales (Gholamalinezhad & Khosravi, 2020), pero los que más destacan son el *Average Pooling* y el *Max Pooling*.

- **Clasificador:** Se trata de una red neuronal del tipo *feed-forward* que se encarga de realizar la asociación del dato de entrada a la categoría que le corresponde (O'Shea & Nash, 2015). La última capa del clasificador tiene tantas unidades como categorías tiene el conjunto de datos utilizado para el entrenamiento. Las capas de convolución de la red se encargan de generar características o *features* que son alimentados al clasificador en forma vectorial.

4 Cuando entrenamos un modelo de Aprendizaje Automático, buscamos que este tenga la capacidad de realizar predicciones adecuadas, aún con datos con los que el mismo no fue entrenado. Overfitting es el escenario en el que el modelo realiza buenas predicciones solamente con los datos del conjunto de entrenamiento, pero no con datos que están fuera de él (Bishop, 2006).

ISSN: 2075-8944
Universidad Privada del Valle – Bolivia
https://doi.org/10.52428/20758944.v17i51.110

Durante el proceso de entrenamiento de la red neuronal, existirá una divergencia entre las categorías correctas asociadas al conjunto de datos y las predicciones del modelo. Esta divergencia es la medida del error de la red neuronal, también identificada como *función objetivo* o *función de pérdida* (*loss function*) y es una cantidad que se busca minimizar progresivamente (Q. Wang et al., 2020).

En problemas de clasificación de imágenes, es común usar funciones de pérdida tales como la entropía cruzada. Para la optimización, se emplean métodos basados en el descenso del gradiente (*gradient descent*). Si bien los aspectos mencionados aquí son elementos importantes en la arquitectura de las redes convolucionales, también lo son en otras arquitecturas, tales como las redes recurrentes y en arquitecturas *Transformer*. Sin embargo, estas arquitecturas van más allá de los alcances del presente artículo.

Finalmente, conviene así también destacar la *convolución transpuesta*, algunas veces también denominada *de-convolución* cuyo objetivo está en generar una representación en espacios de mayor dimensionalidad (Dumoulin & Visin, 2016). Se emplea este tipo de operadores para generar imágenes completas a partir de una distribución latente de probabilidad, tal como se abordará más adelante.

2.3. Métodos generativos para creación de imágenes

Pasaremos ahora a abordar las características esenciales de los diversos métodos generativos para imágenes digitales desde un punto de vista de sus características y aplicabilidad en diversos escenarios.

2.3.1. Caso particular: Naive Bayes

Uno de los modelos generativos más básicos es el de Naive Bayes (Bayes Ingenuo) (Ng & Jordan, 2001). Si bien, este no pertenece al ámbito de modelos de *Deep Learning*, la descripción de dicho modelo será necesaria para a tener una mejor comprensión de qué buscan lograr los modelos pertenecientes al ámbito del *Deep Learning*.

JOURNAL BOLIVIANO DE CIENCIAS – Vol. 17 – Número 51
ISSN: 2075-8944
Universidad Privada del Valle – Bolivia
https://doi.org/10.52428/20758944.v17i51.110

Con anterioridad se indicó que el principal propósito de un modelo generativo está en crear nuevos datos que provienen de una distribución de probabilidad del tipo **p(x|y)** ó **p(x|z)** en el caso de modelos no supervisados. Naive Bayes se utiliza comúnmente para problemas de clasificación (Ng & Jordan, 2001), pero esto involucra que como parte del proceso de entrenamiento del modelo es necesario primeramente modelar **p(x|y)**.

Según lo explicado en la sección 1 sobre modelos generativos para clasificación, es necesario emplear el Teorema de Bayes para realizar predicciones del tipo **p(y|x)**:

$$p(\mathbf{y} \mid \mathbf{x}) = \frac{p(\mathbf{x} \mid \mathbf{y})\, p(\mathbf{y})}{p(\mathbf{x})}$$

En el caso de Naive Bayes, **p(x|y)** puede ser una distribución Bernoulli para datos categóricos (Singh et al., 2019), o bien una distribución Gaussiana para datos en el dominio real. El supuesto principal del modelo radica en la independencia de las características (*features*) que integran **x**.

Por ejemplo, si **x** es una imagen digital, se asume que sus píxeles son estadísticamente independientes entre sí (Raschka, 2017). De ahí nace la denominación del modelo: Naive (ingenuo) en el sentido que se asume que las características o *features* son independientes desde un punto de vista estadístico. La denominación "Bayes" se refiere al uso del Teorema de Bayes como mecanismo para realizar predicciones.

Por razones obvias, el supuesto de independencia estadística entre las características de cada dato no es válida en muchos escenarios (Rish, 2001), por ejemplo el caso de imágenes digitales, pues aquí hay una variedad de interacciones entre un píxel y los que le rodean en su vecindario. Sin embargo, esto se constituye en una ganancia notoria desde el punto de vista computacional, pues no es necesario tener que calcular la inversa de matrices de gran escala para encontrar los parámetros del modelo optimizado.

JOURNAL BOLIVIANO DE CIENCIAS – Vol. 17 – Número 51
ISSN: 2075-8944
Universidad Privada del Valle – Bolivia
https://doi.org/10.52428/20758944.v17i51.110

Ahora bien, si lo que se desea es generar nuevos datos **x** a partir del sistema anteriormente descrito, bastará con muestrear aleatoriamente una nueva imagen a partir de la distribución que le corresponda a su respectiva categoría. En la Figura 5, es posible advertir algunos resultados con imágenes de números manuscritos del 0 al 9:

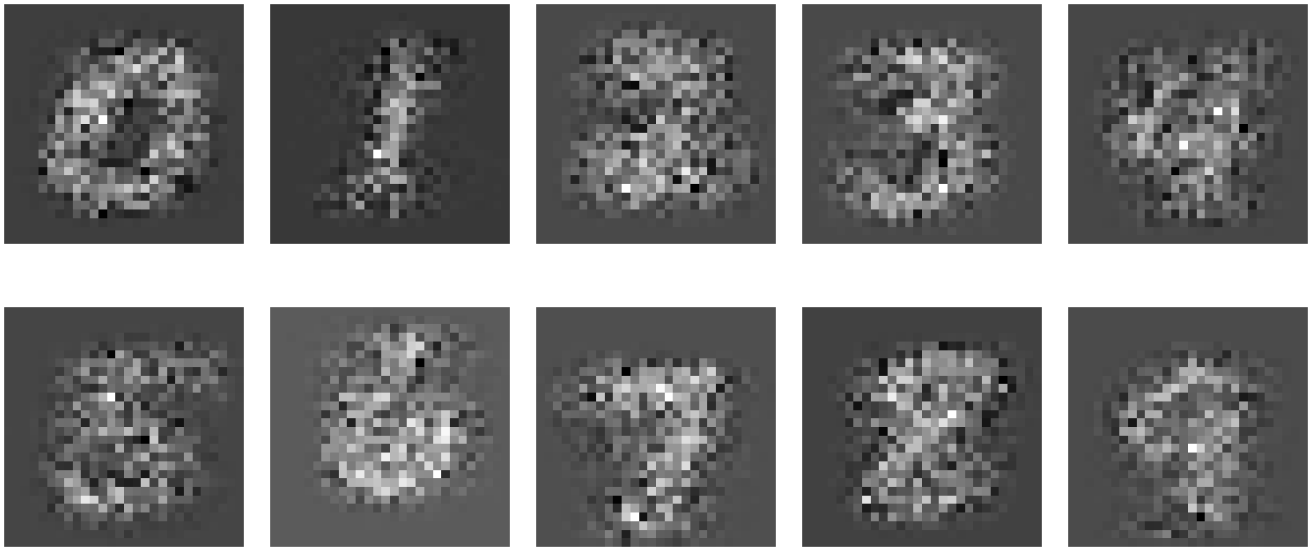

Figura 5. Ejemplos de imágenes generadas por Naive Bayes
Fuente: Elaboración propia, 2021.

Nótese que, en la mayoría de los casos, los píxeles de las imágenes contienen bastante ruido y en otros, es inclusive difícil poder distinguir el número generado. Esto se debe a que, en alusión al supuesto de independencia estadística de cada *feature*, los píxeles se generan aleatoriamente a partir de una distribución de probabilidades distinta. Si bien Naive Bayes otorga importantes ganancias desde el punto de vista de la simplicidad de entrenamiento y costo computacional, no es un modelo adecuado para generar imágenes. Sin embargo, la noción de muestrear nuevos datos a partir de una distribución de probabilidad es algo que también realizan los siguientes modelos a ser abordados.

2.3.2. Redes adversarias generativas (GAN)

Las redes adversarias generativas se constituyen en modelos capaces de generar nuevos datos en base a una distribución probabilística de variables latentes. Hoy por hoy, se trata de uno de los modelos más difundidos en diversas áreas, tales como la generación de nuevas

JOURNAL BOLIVIANO DE CIENCIAS – Vol. 17 – Número 51
ISSN: 2075-8944
Universidad Privada del Valle – Bolivia
https://doi.org/10.52428/20758944.v17i51.110
imágenes. Las redes adversarias generativas fueron propuestas por (I. J. Goodfellow et al., 2014) como una integración de dos modelos:

- **Discriminador:** Cumple la labor de diferenciar entre imágenes reales y falsas. Identifica si existen características en las imágenes que puedan sugerir si las mismas son auténticas o bien fueron generadas por un modelo computacional.
- **Generador:** Cumple la labor de crear nuevas imágenes. El propósito está en generar imágenes que sean tan reales que finalmente terminarán por "engañar" al discriminador.

Es posible entender el proceso de entrenamiento de este tipo de modelos como un juego *mini-max*, donde ambos componentes compiten entre sí, y de ahí es donde nace la definición de *redes adversarias*. Para una mejor comprensión de la arquitectura general del modelo, se ilustrará la interacción del discriminador y el generador a través del gráfico ilustrado en la Figura 6:

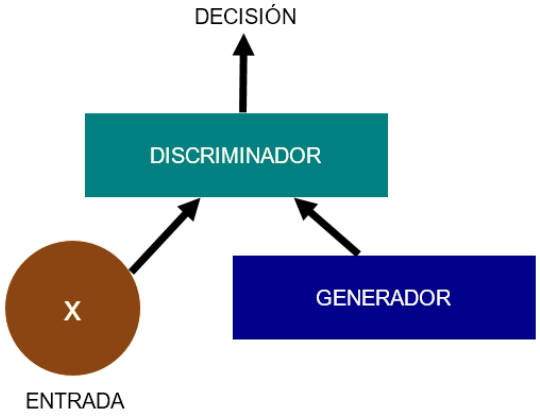

Figura 6. Arquitectura de redes adversarias generativas (GAN)
Fuente: Elaboración propia, 2021.

Aquí, la entrada es una imagen real, mientras que el generador muestrea aleatoriamente nuevos datos a partir de una distribución de probabilidad de variables latentes $p(\mathbf{z})$, por ejemplo, una distribución normal estándar. El discriminador $D(\mathbf{x})$ da como resultado una salida binaria que identifica si el dato ingresado es real o falso. $D(\mathbf{x})$ será muy próxima a 1 cuando el dato $\mathbf{x}$ es real, mientras que será casi cero cuando $\mathbf{x}$ es falso. De esta manera, $D(\mathbf{x})$

JOURNAL BOLIVIANO DE CIENCIAS – Vol. 17 – Número 51
ISSN: 2075-8944
Universidad Privada del Valle – Bolivia
https://doi.org/10.52428/20758944.v17i51.110
es una función sigmoidal, la cual está comprendida en el rango (0,1) (I. J. Goodfellow et al., 2014).

Si se identifica a la salida del generador como $\mathbf{x}' = G(\mathbf{z})$, entonces el propósito está en lograr que $D(\mathbf{x}') = D(G(\mathbf{z}))$ sea tan próximo a 1 como sea posible. Cuando eso ocurra, entonces será indicación de que el generador $G(\mathbf{z})$ está haciendo un excelente trabajo en generar imágenes falsas con una fidelidad tal que es capaz de engañar al discriminador. Luego, formalmente la función objetivo a optimizar puede ser definida de la siguiente manera (I. J. Goodfellow et al., 2014) (Gui et al., 2020):

$$J(\boldsymbol{\theta}) = -\min_D \max_G \{E_{\mathbf{x}\sim p(\mathbf{x})}[\log D(\mathbf{x})] + E_{\mathbf{z}\sim p(\mathbf{z})}[\log(1 - D(G(\mathbf{z})))]\} \qquad (6)$$

De esta manera, es posible comprender el proceso de optimización del modelo como una competencia donde el generador busca maximizar la probabilidad de imágenes falsas y el discriminador, la probabilidad de imágenes reales. Los signos de ambos logaritmos en la ecuación 6 se neutralizan con el signo negativo de la factorización de la función mini-max, por lo que ésta siempre tendrá un valor positivo. Es conveniente acotar que esta configuración de la función objetivo a optimizar puede fácilmente ser implementada en la mayoría de los *frameworks* modernos de *Deep Learning.*

Conviene destacar que tanto el generador como el discriminador pueden ser redes multicapa de cualquier tipo. Sin embargo, para que estas puedan ser empleadas con imágenes digitales, es menester emplear redes convolucionales. De este modo, se han investigado diversas arquitecturas de redes GAN para el ámbito de generación de imágenes, donde resaltan las DCGAN (Deep Convolutional Generative Adversarial Networks) propuestas por (Radford et al., 2016). De ahí se derivan otras arquitecturas más específicas, tales como las GAN condicionales (Mirza & Osindero, 2014), InfoGAN (Chen et al., 2016) y StackGAN (Zhang et al., 2016).

JOURNAL BOLIVIANO DE CIENCIAS – Vol. 17 – Número 51
ISSN: 2075-8944
Universidad Privada del Valle – Bolivia
https://doi.org/10.52428/20758944.v17i51.110

Las redes DCGAN se caracterizan por usar convoluciones de paso fraccional. Aunque la definición es incorrecta, coloquialmente también denomina a este tipo de operaciones como *de-convoluciones* ya que logran un resultado inverso al que se obtendría con bloques convolucionales regulares (Pu et al., 2016). A continuación, se presentará la arquitectura del generador propuesto por (Radford et al., 2016). Véase que, al contrario de una red convolucional tradicional, las salidas de cada capa consecutiva van incrementando en tamaño (Figura 7):

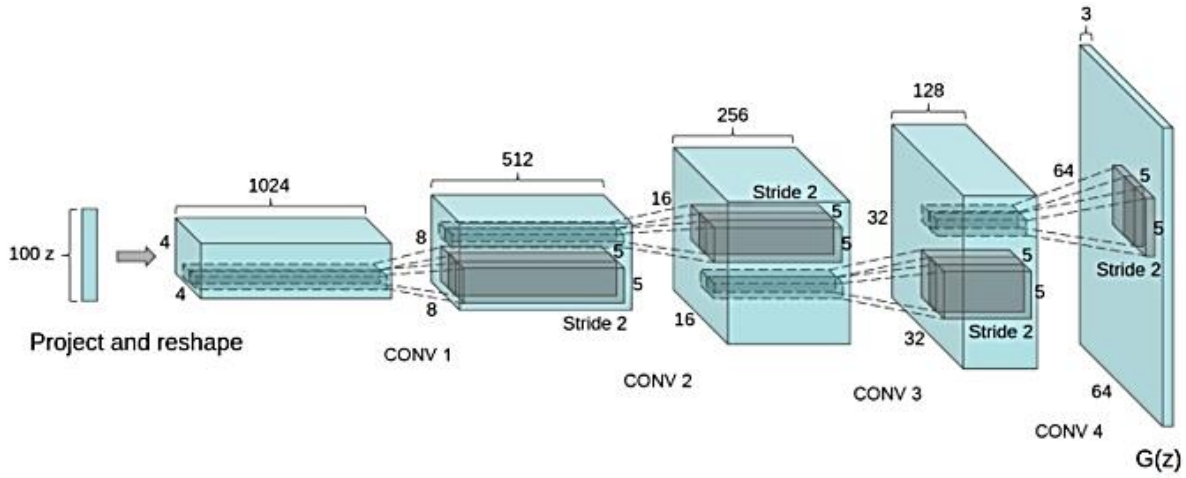

Figura 7. Arquitectura del generador de una DCGAN

Fuente: (Radford et al., 2016).

Aquí, cada capa convolucional incrementa el tamaño de la anterior, hasta que la salida final es la imagen generada de las dimensiones requeridas, la cual será posteriormente alimentada al discriminador. La entrada al generador es una distribución uniforme de 100 dimensiones. En general, es posible afirmar que se trata de un modelo no supervisado que en base a los datos de la distribución de **z** puede generar una variedad de imágenes con mucho detalle en los objetos que integran la escena. Ejemplos de diversos resultados obtenidos por Radford *et al*.(2016) son mostrados en la Figura 8:

JOURNAL BOLIVIANO DE CIENCIAS – Vol. 17 – Número 51
ISSN: 2075-8944
Universidad Privada del Valle – Bolivia
https://doi.org/10.52428/20758944.v17i51.110

Figura 8. *Resultados obtenidos luego de 5 etapas de entrenamiento de una DCGAN Fuente: (Radford et al., 2016).*

Como es posible advertir, las imágenes generadas tienen bastante detalle, y en algunos casos, objetos tales como ventanas son inclusive reemplazados por otros elementos. Esto sin duda es indicación de la riqueza visual de los objetos que el modelo genera luego de haber sido entrenado. En el trabajo original de (Radford et al., 2016), se especifica que también fue realizado un cuidadoso diseño del discriminador, a través del uso de funciones de activación específicas para el mismo.

Una de las principales limitaciones de las GAN es la falta de diversidad en las imágenes generadas que es común cuando el generador se especializa en imágenes para un tipo específico de discriminador. Este fenómeno también se conoce como *colapso de modo* (Bang & Shim, 2018). Por otra parte, la inestabilidad en el proceso de entrenamiento se constituye en un factor limitante, ya que el modelo es sensible a la configuración de sus *hiperparámetros*.

Asimismo, validar este tipo de modelos es una tarea compleja, lo cual compromete los estándares evaluativos para evitar el sobreajuste. Sin embargo, más allá de estas limitaciones, las redes GAN han venido demostrando importantes avances en diversas áreas. Sitios tales

JOURNAL BOLIVIANO DE CIENCIAS – Vol. 17 – Número 51
ISSN: 2075-8944
Universidad Privada del Valle – Bolivia
https://doi.org/10.52428/20758944.v17i51.110
como *thispersondoesnotexist.com* son un claro ejemplo de lo que es posible lograr a través de este tipo de modelos, y hay muchas más perspectivas en un futuro no muy lejano.

2.3.3. *Autoencoders* variacionales

Dentro de las grandes categorías de modelos generativos para imágenes, se tiene también a los *autoencoders variacionales*. Este tipo de modelos, son en realidad una extensión de los *autoencoders* tradicionales (Bank et al., 2021). En un contexto general, se afirma que un *autoencoder* está compuesto por dos elementos principales:

- Un codificador (*encoder*) que se encarga de comprimir los datos de entrada a una representación reducida. Es posible interpretar este proceso como una reducción de dimensionalidad, tal como la que se realiza a través de métodos como el PCA (Análisis de Componentes Principales) (Bishop, 2006).
- Un decodificador (*decoder*) que realiza un proceso inverso al del codificador. Este descomprime o decodifica la información reducida y genera nuevos datos a partir de ella.

Un esquema general de los *autoencoders* en el ámbito de generación imágenes se presenta a continuación en la Figura 9:

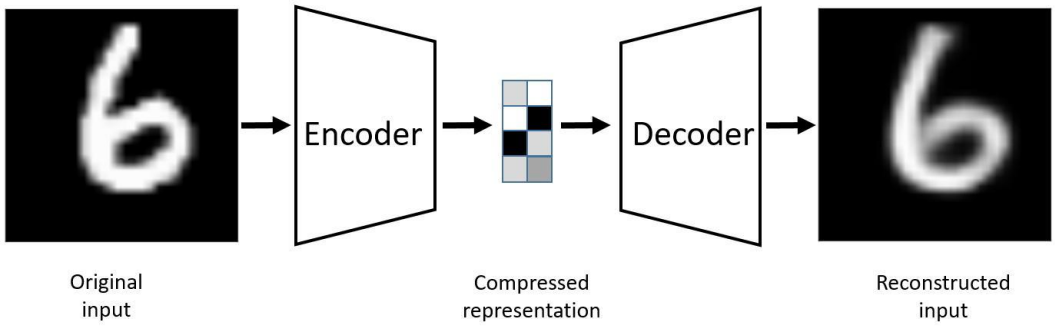

Figura 9. Arquitectura general de un autoencoder
Fuente: (Bank et al., 2021).

Un aspecto que conviene destacar con relación a los *autoencoders* es que la decodificación puede o no tener lugar con pérdidas. Asimismo, es posible interpretar la representación comprimida como una distribución probabilística de variables latentes. En ese sentido, el

JOURNAL BOLIVIANO DE CIENCIAS – Vol. 17 – Número 51
ISSN: 2075-8944
Universidad Privada del Valle – Bolivia
https://doi.org/10.52428/20758944.v17i51.110
decodificador actuará como un modelo generativo capaz de muestrear datos a partir de dicha distribución probabilística para crear nuevos datos.

Los *Autoencoders Variacionales* (VAE) fueron propuestos por el equipo de Kignma & Welling (Kingma & Welling, 2014) y a partir de ello se fueron desarrollando una variedad de implementaciones en diversos dominios, donde destacan la generación de imágenes e inclusive la creación automática de modelos tridimensionales (Tan et al., 2018).

La principal diferencia entre un *autoencoder* tradicional y uno variacional radica en la interpretación probabilística de cada componente. De esta manera, se establece que el codificador está asociado a una probabilidad del tipo $p(\mathbf{z}\,|\,\mathbf{x})$, mientras que el decodificador puede expresarse a través de $p(\mathbf{x}\,|\,\mathbf{z})$. Es posible relacionar el trabajo del decodificador con el generador en las redes GAN que se describirán en la sección 2.3.2. Por otra parte, $\mathbf{z}$ es una variable latente en el dominio real, y para el caso particular de los VAE se asumirá que pertenece a una distribución gaussiana estándar. Para el ámbito de generación de imágenes, el codificador será normalmente una red convolucional, mientras que el decodificador será una red de *deconvoluciones* que amplifican progresivamente las dimensiones del resultado (Pu et al., 2016).

De este modo, podemos expresar el codificador en función del decodificador haciendo uso del Teorema de Bayes, de la siguiente manera:

$$p(\mathbf{z}\,|\,\mathbf{x}) = \frac{p(\mathbf{x}\,|\,\mathbf{z})\,p(\mathbf{z})}{p(\mathbf{x})} \tag{7}$$

En esta expresión, el denominador también puede ser expresado de la siguiente manera:

$$p(\mathbf{x}) = \int_{\mathbf{z}} p(\mathbf{x},\mathbf{z})\,d\mathbf{z} \tag{8}$$

Lo cual también se interpreta como *marginalización respecto de* $\mathbf{z}$ (Spiegel et al., 2000). Desafortunadamente, no es posible evaluar esta integral de manera directa, ya que la misma es intratable desde un punto de vista matemático. Este problema también conlleva a que la distribución posterior $p(\mathbf{z}\,|\,\mathbf{x})$ también sea intratable y, por ende, no podrá ser evaluada analíticamente. De esta manera, la alternativa es usar métodos afines a la Inferencia

JOURNAL BOLIVIANO DE CIENCIAS – Vol. 17 – Número 51
ISSN: 2075-8944
Universidad Privada del Valle – Bolivia
https://doi.org/10.52428/20758944.v17i51.110

Aproximada para encontrar una distribución equivalente a $p(\mathbf{x})$ y $p(\mathbf{z}\,|\,\mathbf{x})$ que sí pueda ser evaluada y optimizada. De esta manera, se dirá que $q(\mathbf{z}\,|\,\mathbf{x})$ aproxima $p(\mathbf{z}\,|\,\mathbf{x})$ y por ende se busca que ambas distribuciones sean lo más parecidas posible. En otras palabras, se busca minimizar la Divergencia de Kullback-Leibler entre *p* y *q*, la cual formalmente puede ser expresada como:

$$KL(q(\mathbf{z}\,|\,\mathbf{x}) \,\|\, p(\mathbf{z}\,|\,\mathbf{x})) = \mathrm{E}\left[\ln \frac{q(\mathbf{z}\,|\,\mathbf{x})}{p(\mathbf{z}\,|\,\mathbf{x})}\right] = -\int_{\mathbf{z}} q(\mathbf{z}\,|\,\mathbf{x}) \ln \left|\frac{p(\mathbf{z}\,|\,\mathbf{x})}{q(\mathbf{z}\,|\,\mathbf{x})}\right| d\mathbf{z} \qquad (9)$$

Al reemplazar $p(\mathbf{z}\,|\,\mathbf{x})$ de la ecuación 7 en 9, se tendrá:

$$KL(q(\mathbf{z}\,|\,\mathbf{x}) \,\|\, p(\mathbf{z}\,|\,\mathbf{x})) = -\int_{\mathbf{z}} q(\mathbf{z}\,|\,\mathbf{x}) \ln \left|\frac{p(\mathbf{x}\,|\,\mathbf{z})\,p(\mathbf{z})}{q(\mathbf{z}\,|\,\mathbf{x})\,p(\mathbf{x})}\right| d\mathbf{z} \qquad (10)$$

Desarrollando el anterior término:

$$KL(q(\mathbf{z}\,|\,\mathbf{x}) \,\|\, p(\mathbf{z}\,|\,\mathbf{x})) = -\int_{\mathbf{z}} q(\mathbf{z}\,|\,\mathbf{x}) \ln \left|\frac{p(\mathbf{z})}{q(\mathbf{z}\,|\,\mathbf{x})}\right| d\mathbf{z} + \ln p(\mathbf{x}) \int_{\mathbf{z}} q(\mathbf{z}\,|\,\mathbf{x}) d\mathbf{z} - \int_{\mathbf{z}} q(\mathbf{z}\,|\,\mathbf{x}) \ln p(\mathbf{x}\,|\,\mathbf{z}) d\mathbf{z} \qquad (11)$$

Sabiendo que la integral que multiplica a $\ln p(\mathbf{x})$ equivale a 1 en la ecuación 11, y que las otras integrales son valores esperados de los logaritmos, finalmente es posible deducir que:

$$\ln p(\mathbf{x}) = KL(q(\mathbf{z}\,|\,\mathbf{x}) \,\|\, p(\mathbf{z}\,|\,\mathbf{x})) + \mathrm{E}\left[\ln \frac{p(\mathbf{z})}{q(\mathbf{z}\,|\,\mathbf{x})}\right] + \mathrm{E}[\ln p(\mathbf{x}\,|\,\mathbf{z})] \qquad (12)$$

Sabiendo que la Divergencia de Kullback-Leibler es siempre un valor positivo, una consecuencia de la anterior deducción es la siguiente:

$$\ln p(\mathbf{x}) \geq \mathrm{E}\left[\ln \frac{p(\mathbf{z})}{q(\mathbf{z}\,|\,\mathbf{x})}\right] + \mathrm{E}[\ln p(\mathbf{x}\,|\,\mathbf{z})] \qquad (13)$$

O bien:

$$\ln p(\mathbf{x}) \geq -\mathrm{E}\left[\ln \frac{q(\mathbf{z}\,|\,\mathbf{x})}{p(\mathbf{z})}\right] + \mathrm{E}[\ln p(\mathbf{x}\,|\,\mathbf{z})] \qquad (14)$$

La primera esperanza matemática a mano derecha es una Divergencia de Kullback-Leibler de $q(\mathbf{z}\,|\,\mathbf{x})$ y $p(\mathbf{z})$, por lo cual es posible reescribir la anterior expresión de la siguiente manera:

$$\ln p(\mathbf{x}) \geq -KL(q(\mathbf{z}\,|\,\mathbf{x}) \,\|\, p(\mathbf{z})) + \mathrm{E}[\ln p(\mathbf{x}\,|\,\mathbf{z})] \qquad (15)$$

JOURNAL BOLIVIANO DE CIENCIAS – Vol. 17 – Número 51
ISSN: 2075-8944
Universidad Privada del Valle – Bolivia
https://doi.org/10.52428/20758944.v17i51.110
La expresión a mano derecha en la desigualdad 15 corresponde a la función objetivo ELBO (*Evidence Lower-Bound*) y es una función que se busca maximizar (Shekhovtsov et al., 2021). Sin embargo, es posible convertir esta función en un objetivo de minimización simplemente cambiando el signo. De esta manera, la función de pérdida del VAE será:

$$\text{Loss} = KL(q(\mathbf{z}\,|\,\mathbf{x}) \,\|\, p(\mathbf{z})) - \text{E}[\ln p(\mathbf{x}\,|\,\mathbf{z})] \tag{16}$$

De esta manera, el primer término (la divergencia de Kullback-Leibler) de la función objetivo en 16 actúa como una regularización para el modelo (Kingma & Welling, 2014), y se puede deducir que esto le permite a los autoencoders variacionales incrementar su capacidad predictiva mientras reducen el sobreajuste. Por supuesto, esta es una característica esperada en este tipo de modelos, ya que ello les agregará mayor riqueza y variedad a las imágenes resultantes.

El segundo término (la esperanza matemática en 16) es el error de reconstrucción, y puede ser implementado en la forma de una media del error cuadrático medio de las salidas. En el caso de los *autoencoders variacionales*, la divergencia de Kullback-Leibler está asociada a una distribución gaussiana con parámetros μ_q y σ_q. De ese modo, es posible finalmente enunciar la función objetivo del modelo, de la siguiente manera (Kingma & Welling, 2014):

$$\text{Loss} = -\frac{1}{2}\left[1 + \ln \sigma_q^2 - \sigma_q^2 - \mu_q^2\right] - \text{E}[\ln p(\mathbf{x}\,|\,\mathbf{z})] \tag{17}$$

En esta expresión, μ_q y σ_q deben formar parte del grupo de parámetros optimizables del modelo, lo cual implica que debe calcularse la derivada de $\mathbf{z}$ respecto de μ_q y σ_q. Sin embargo, hay una dificultad con ello, ya que una variable aleatoria como $\mathbf{z}$ no es diferenciable. La forma en que es posible resolver esta limitación es introduciendo una nueva variable aleatoria $\varepsilon \sim \text{N}(0,1)$ de tal manera que:

$$\frac{\mathbf{z} - \mu_q}{\sigma_q} = \varepsilon \tag{18}$$

De lo anterior:

$$\mathbf{z} = \mu_q + \sigma_q \circ \varepsilon \tag{19}$$

JOURNAL BOLIVIANO DE CIENCIAS – Vol. 17 – Número 51
ISSN: 2075-8944
Universidad Privada del Valle – Bolivia
https://doi.org/10.52428/20758944.v17i51.110

Gracias a este *truco de reparametrización* (Kingma & Welling, 2014), es posible separar el proceso de generar los números aleatorios del entrenamiento mismo del modelo, y es algo que puede implementarse de manera muy sencilla a través de *frameworks* modernos de *Deep Learning*.

Los *autoencoders variacionales* pueden emplearse para una variedad de tareas, y aquí destaca la generación de imágenes digitales a partir de una adecuada distribución de variables latentes. Recientes avances en los modelos VAE (Razavi et al., 2019), permiten sobrellevar los problemas de falta de diversidad en las GAN, y explotan la riqueza de las distribuciones latentes de probabilidad en el modelo. Algunos ejemplos de generación son mostrados en la Figura 10:

Figura 10. Ejemplos de generación de imágenes a través del modelo VQ-VAE-2
Fuente: (Razavi et al., 2019).

Una de las limitaciones de los modelos VAE está en las imperfecciones de las imágenes que generan. Debido al ruido aleatorio, existen casos en los que los objetos de las imágenes generadas aparecen con artefactos tales como el difuminado. Sin embargo, estas limitaciones se han venido sobrellevando a través de arquitecturas más recientes basadas en VAE.

También se han dado trabajos interesantes que pretenden combinar VAE's con GAN's llegando a resultados que combinan lo mejor de ambos mundos (Larsen et al., 2016). En general es posible afirmar que el rubro de modelos generativos para imágenes aún está en

ISSN: 2075-8944
Universidad Privada del Valle – Bolivia
https://doi.org/10.52428/20758944.v17i51.110
franco desarrollo, y las diversas limitaciones de los modelos aquí descritos se están abordando con miras a mejores resultados. Otra interesante propuesta que surge de la investigación de (Razavi et al., 2019) es la combinación con redes del tipo Pixel CNN que son una extensión de las redes recurrentes de píxel (Pixel RNN) propuestas por (Oord et al., 2016) en las cuales, se incluye información contextual de vecindarios de píxel a través de redes recurrentes o convolucionales, llegando a resultados muy competitivos en generación de imágenes de alta fidelidad.

2.4. Implicaciones éticas de los modelos generativos

No cabe la menor duda que la capacidad de diversos modelos de generar nuevos datos en base a determinados parámetros, es uno de los avances más impresionantes en la Inteligencia Artificial moderna. Sin embargo, hoy en día no solamente se habla de modelos capaces de generar imágenes, según lo descrito en el presente artículo, sino también de aquellos que son capaces de generar secuencias de vídeo, así como textos completos e inclusive código fuente de programas.

Con relación a lo último, GPT-3 (Brown et al., 2020) es un modelo basado en la arquitectura *Transformer*[5] que cuenta con la capacidad de generar textos completos a partir de una instrucción específica. Pero inclusive estos modelos se están aplicando ya en la creación de código de programas computacionales. Un estudio reciente por (Jiang et al., 2021) explora el uso de *Transformers* para generación de imágenes de alta fidelidad, pero para imágenes de reducida resolución.

Para muchos, estos avances son motivo de mucho entusiasmo, ya que esto indica que la Inteligencia Artificial está dando pasos importantes y siempre habrá mucho espacio para continuar investigando y descubriendo cosas realmente interesantes. Pero para otros, esto es motivo de preocupación ya que, si no median aspectos éticos y legales, estas tecnologías pueden llegar a las manos equivocadas y ser usadas para fines negativos. Algunos de los

5 Un tipo de red neuronal que consta de un codificador y decodificador. Estos modelos se han implementado con gran éxito en el ámbito del Procesamiento del Lenguaje Natural (NLP) y ya son el estado del arte en varias áreas. GPT-3 es un decodificador capaz de generar texto e incluso código fuente de programas.

ISSN: 2075-8944
Universidad Privada del Valle – Bolivia
https://doi.org/10.52428/20758944.v17i51.110
aspectos éticos y legales que conviene tener en cuenta con relación a los modelos generativos son los siguientes:

- La capacidad de generar nuevos recursos, tales como imágenes e inclusive vídeo, ingresa en el ámbito de los *deep fakes* (falsos profundos). Si los modelos generativos son empleados con fines negativos, entonces estos podrían usarse para crear fotografías de personas que no existen e inclusive generar vídeos que contengan información falsa (Ali et al., 2021).
- Siguiendo esta misma línea, un uso inadecuado de modelos tales como GPT-3 puede llevar a la generación de noticias falsas, o emitir versiones alejadas de la realidad.
- En la misma línea de los aspectos éticos, existen sectores de la sociedad, que ya empiezan a mostrar preocupación en el sentido en que la Inteligencia Artificial reemplazará muchas de las actividades del cotidiano vivir de las personas (Abuselidze & Mamaladze, 2021). Si existen modelos capaces de generar textos completos, entonces ¿dónde quedan las habilidades innatas de los seres humanos?

Si bien estos son temas que evocan preocupación, también se debe tener en cuenta que los modelos requieren datos para poder ser entrenados y también deben orientarse a la resolución de problemas reales, así como a la innovación (Cockburn et al., 2018). Los modelos simplemente cumplirán el fin que el ser humano les haya consignado. Al igual que en cualquier otra tecnología, la Inteligencia Artificial puede ser utilizada en beneficio de la sociedad, o puede tener efectos negativos si se destina para tales fines. Es por ello que, al momento de crear aplicaciones relacionadas a la Inteligencia Artificial, es importante tomar en cuenta los aspectos legales y éticos, lo cual es un tema constante de debate en diversas esferas de la comunidad científica y la sociedad.

3. CONCLUSIONES

El Aprendizaje Automático ha venido teniendo importantes avances en los últimos años. Una parte importante de estos últimos avances se concentra en la implementación de modelos generativos para la creación de imágenes digitales. En este artículo se realizó la descripción

ISSN: 2075-8944
Universidad Privada del Valle – Bolivia
https://doi.org/10.52428/20758944.v17i51.110
de dos grandes modelos: Las Redes Adversarias Generativas (GAN) y los Autoencoders Variacionales (VAE).

Si bien ambos modelos tienen sus propias fortalezas y limitaciones, conviene destacar que estos tienen la capacidad de generar imágenes muy realistas en base a ruido generado por una distribución de probabilidad. Las GAN son sensibles a los hiperparámetros de configuración y adolecen de problemas tales como la falta de diversidad de las imágenes generadas. Los modelos VAE tienden a generar imágenes con artefactos no deseados, tales como el difuminado en los objetos. Otros modelos generativos tales como Pixel CNN se han usado en conjunción con los VAE para obtener mejores resultados. Inclusive, se han obtenido resultados muy prometedores combinando las GAN con VAE's. Esto sugiere que más allá de las limitaciones que uno u otro método pueda tener, a través de combinaciones de ellos se pueden lograr mejores resultados.

Algo interesante en un futuro, será ver qué resultados se obtienen utilizando modelos basados en la arquitectura *Transformer* que hoy por hoy ya son el estado del arte en ámbitos tales como el Procesamiento del Lenguaje Natural (Chernyavskiy et al., 2021). Si bien se realizaron algunos pasos en cuanto a generación de imágenes con el estudio de Jiang *et al.* (2021), aún queda por ver si a través de los *Transformers* es posible generar imágenes más complejas y de mayor resolución.

Finalmente, al momento de desarrollar modelos generativos para diversos contextos, también es importante considerar los aspectos éticos y legales que regulen el uso adecuado de los mismos.

4. RECOMENDACIONES Y TRABAJO FUTURO

Los modelos generativos para imágenes digitales han tenido desarrollos muy importantes que van a la par de los avances que la Inteligencia Artificial ha ostentado en la última década. Hoy en día, gracias a las redes GAN y VAE podemos generar fotografías con alta fidelidad. Sin embargo, ambos modelos tienen limitaciones características de la naturaleza de las arquitecturas que implementan, mayormente basadas en redes convolucionales. Hoy en día,

ISSN: 2075-8944
Universidad Privada del Valle – Bolivia
https://doi.org/10.52428/20758944.v17i51.110
los modelos *Transformer* son ya un éxito en varias tareas relacionadas al *Procesamiento del Lenguaje Natural*. Estudios recientes por (Jiang et al., 2021) muestran que estos modelos también pueden emplearse para generar imágenes.

El futuro de los modelos generativos estará dictado por el grado de éxito que se tenga con los modelos *Transformer* y su popularización en el trabajo con imágenes y vídeo. Sin embargo, en la medida que los modelos generativos se hacen más complejos, también será necesario apuntar a mejores alternativas de verificación de contenido para evitar la presentación de información falsa en la forma de imágenes o vídeo.

REFERENCIAS

Abuselidze, G., & Mamaladze, L. (2021). The impact of artificial intelligence on employment before and during pandemic: A comparative analysis. *Journal of Physics: Conference Series*, *1840*(1), 012040. https://doi.org/10.1088/1742-6596/1840/1/012040

Ali, S., DiPaola, D., Lee, I., Hong, J., & Breazeal, C. (2021). Exploring Generative Models with Middle School Students. En *Proceedings of the 2021 CHI Conference on Human Factors in Computing Systems* (pp. 1–13). Association for Computing Machinery. https://doi.org/10.1145/3411764.3445226

Bang, D., & Shim, H. (2018). MGGAN: Solving Mode Collapse using Manifold Guided Training. *arXiv.org:1804.04391 [cs]*. Obtenido de : http://arxiv.org/abs/1804.04391

Bank, D., Koenigstein, N., & Giryes, R. (2021). Autoencoders. *arXiv:2003.05991 [cs, stat]*. http://arxiv.org/abs/2003.05991

Bishop, C. (2006). *Pattern Recognition and Machine Learning*. Springer.

Brown, T. B., Mann, B., Ryder, N., Subbiah, M., Kaplan, J., Dhariwal, P., Neelakantan, A., Shyam, P., Sastry, G., Askell, A., Agarwal, S., Herbert-Voss, A., Krueger, G., Henighan, T., Child, R., Ramesh, A., Ziegler, D. M., Wu, J., Winter, C., … Amodei, D. (2020). Language Models are Few-Shot Learners. *arXiv:2005.14165 [cs]*. http://arxiv.org/abs/2005.14165

Chen, X., Duan, Y., Houthooft, R., Schulman, J., Sutskever, I., & Abbeel, P. (2016). *InfoGAN: Interpretable Representation Learning by Information Maximizing Generative Adversarial Nets*. https://arxiv.org/abs/1606.03657v1

ISSN: 2075-8944
Universidad Privada del Valle – Bolivia
https://doi.org/10.52428/20758944.v17i51.110
Chernyavskiy, A., Ilvovsky, D., & Nakov, P. (2021). Transformers: "The End of History" for NLP? *arXiv:2105.00813 [cs]*. http://arxiv.org/abs/2105.00813.

Cockburn, I. M., Henderson, R., & Stern, S. (2018). *The Impact of Artificial Intelligence on Innovation* (Working Paper Nº 24449; Working Paper Series). National Bureau of Economic Research. https://doi.org/10.3386/w24449.

Deng L., (2012) The MNIST Database of Handwritten Digit Images for Machine Learning Research [Best of the Web], *IEEE Signal Processing Magazine*, vol. 29, no. 6, pp. 141-142, doi: 10.1109/MSP.2012.2211477.

Dumoulin, V., & Visin, F. (2016). *A guide to convolution arithmetic for deep learning*. https://arxiv.org/abs/1603.07285v2

Gholamalinezhad, H., & Khosravi, H. (2020). Pooling Methods in Deep Neural Networks, a Review. *arXiv:2009.07485 [cs]*. http://arxiv.org/abs/2009.07485

Gm, H., Gourisaria, M. K., Pandey, M., & Rautaray, S. S. (2020). A comprehensive survey and analysis of generative models in machine learning. *Computer Science Review*, *38*, 100285. https://doi.org/10.1016/j.cosrev.2020.100285.

Goodfellow, I., Bengio, Y., & Courville, A. (2016). *Deep Learning*. MIT Press.

Goodfellow, I. J., Pouget-Abadie, J., Mirza, M., Xu, B., Warde-Farley, D., Ozair, S., Courville, A., & Bengio, Y. (2014). Generative Adversarial Networks. *arXiv:1406.2661 [cs, stat]*. http://arxiv.org/abs/1406.2661.

Gui, J., Sun, Z., Wen, Y., Tao, D., & Ye, J. (2020). A Review on Generative Adversarial Networks: Algorithms, Theory, and Applications. *arXiv:2001.06937 [cs, stat]*. http://arxiv.org/abs/2001.06937

He, K., Zhang, X., Ren, S., & Sun, J. (2015). Deep Residual Learning for Image Recognition. *arXiv:1512.03385 [cs]*. http://arxiv.org/abs/1512.03385.

Janiesch, C., Zschech, P., & Heinrich, K. (2021). Machine learning and deep learning. *Electronic Markets*, *31*(3), 685–695. https://doi.org/10.1007/s12525-021-00475-2.

Jeon, W., Ko, G., Lee, J., Lee, H., Ha, D., & Ro, W. W. (2021). Chapter Six—Deep learning with GPUs. En S. Kim & G. C. Deka (Eds.), *Advances in Computers* (Vol. 122, pp. 167–215). Elsevier. https://doi.org/10.1016/bs.adcom.2020.11.003.

ISSN: 2075-8944
Universidad Privada del Valle – Bolivia
https://doi.org/10.52428/20758944.v17i51.110
Jiang, Y., Chang, S., & Wang, Z. (2021). TransGAN: Two Pure Transformers Can Make One Strong GAN, and That Can Scale Up. *arXiv:2102.07074 [cs]*. http://arxiv.org/abs/2102.07074.

Khan, A., Sohail, A., Zahoora, U., & Qureshi, A. S. (2020). A survey of the recent architectures of deep convolutional neural networks. *Artificial Intelligence Review*, *53*(8), 5455–5516. https://doi.org/10.1007/s10462-020-09825-6.

Kingma, D. P., & Welling, M. (2014). Auto-Encoding Variational Bayes. *arXiv:1312.6114 [cs, stat]*. http://arxiv.org/abs/1312.6114.

Kotsiantis, S., Zaharakis, I., & Pintelas, P. (2006). Machine learning: A review of classification and combining techniques. *Artificial Intelligence Review*, *26*, 159–190. https://doi.org/10.1007/s10462-007-9052-3.

Krizhevsky, A., Sutskever, I., & Hinton, G. E. (2012). ImageNet classification with deep convolutional neural networks. *Proceedings of the 25th International Conference on Neural Information Processing Systems - Volume 1*, 1097–1105.

Larsen, A. B. L., Sønderby, S. K., Larochelle, H., & Winther, O. (2016). Autoencoding beyond pixels using a learned similarity metric. *arXiv:1512.09300 [cs, stat]*. http://arxiv.org/abs/1512.09300.

Li, Z., Yang, W., Peng, S., & Liu, F. (2020). A Survey of Convolutional Neural Networks: Analysis, Applications, and Prospects. *arXiv:2004.02806 [cs, eess]*. http://arxiv.org/abs/2004.02806

Mirza, M., & Osindero, S. (2014). *Conditional Generative Adversarial Nets*. https://arxiv.org/abs/1411.1784v1

Ng, A. Y., & Jordan, M. I. (2001). On discriminative vs. generative classifiers: A comparison of logistic regression and naive Bayes. *Proceedings of the 14th International Conference on Neural Information Processing Systems: Natural and Synthetic*, 841–848.

Oord, A. van den, Kalchbrenner, N., & Kavukcuoglu, K. (2016). Pixel Recurrent Neural Networks. *arXiv:1601.06759 [cs]*. http://arxiv.org/abs/1601.06759.

O'Shea, K., & Nash, R. (2015). An Introduction to Convolutional Neural Networks. *ArXiv e-prints*. https://arxiv.org/abs/1511.08458.

Pu, Y., Gan, Z., Henao, R., Yuan, X., Li, C., Stevens, A., & Carin, L. (2016). Variational Autoencoder for Deep Learning of Images, Labels and Captions. *arXiv:1609.08976 [cs, stat]*. http://arxiv.org/abs/1609.08976.

Radford, A., Metz, L., & Chintala, S. (2016). Unsupervised Representation Learning with Deep Convolutional Generative Adversarial Networks. *arXiv:1511.06434 [cs]*. http://arxiv.org/abs/1511.06434.

Raschka, S. (2017). Naive Bayes and Text Classification I - Introduction and Theory. *arXiv:1410.5329 [cs]*. http://arxiv.org/abs/1410.5329.

Razavi, A., Oord, A. van den, & Vinyals, O. (2019). Generating Diverse High-Fidelity Images with VQ-VAE-2. *arXiv:1906.00446 [cs, stat]*. http://arxiv.org/abs/1906.00446.

Rish, I. (2001). An Empirical Study of the Naïve Bayes Classifier. *IJCAI 2001 Work Empir Methods Artif Intell*, *3*. doi : 10.1.1.330.2788.

Ruder, S. (2017). An overview of gradient descent optimization algorithms. *arXiv:1609.04747 [cs]*. http://arxiv.org/abs/1609.04747.

Shekhovtsov, A., Schlesinger, D., & Flach, B. (2021). VAE Approximation Error: ELBO and Conditional Independence. *arXiv:2102.09310 [cs, stat]*. http://arxiv.org/abs/2102.09310

Simonyan, K., & Zisserman, A. (2015). Very Deep Convolutional Networks for Large-Scale Image Recognition. *arXiv:1409.1556 [cs]*. http://arxiv.org/abs/1409.1556

Singh, G., Kumar, B., Gaur, L., & Tyagi, A. (2019). Comparison between Multinomial and Bernoulli Naïve Bayes for Text Classification. *2019 International Conference on Automation, Computational and Technology Management (ICACTM)*, 593–596. https://doi.org/10.1109/ICACTM.2019.8776800

Spiegel, M. R., Schiller, J. J., & Srinivasan, R. A. (2000). *Schaum's outline of theory and problems of probability and statistics; 2nd ed.* McGraw-Hill. https://cds.cern.ch/record/450344

Tan, Q., Gao, L., Lai, Y.-K., & Xia, S. (2018). Variational Autoencoders for Deforming 3D Mesh Models. *arXiv:1709.04307 [cs]*. http://arxiv.org/abs/1709.04307.

Wang, M., & Deng, W. (2021). Deep Face Recognition: A Survey. *Neurocomputing*, *429*, 215–244. https://doi.org/10.1016/j.neucom.2020.10.081.

Wang, Q., Ma, Y., Zhao, K., & Tian, Y. (2020). A Comprehensive Survey of Loss Functions in Machine Learning. *Annals of Data Science*. https://doi.org/10.1007/s40745-020-00253-5

Yamashita, R., Nishio, M., Do, R. K. G., & Togashi, K. (2018). Convolutional neural networks: An overview and application in radiology. *Insights into Imaging*, *9*(4), 611–629. https://doi.org/10.1007/s13244-018-0639-9

Zhang, H., Xu, T., Li, H., Zhang, S., Wang, X., Huang, X., & Metaxas, D. (2016). *StackGAN: Text to Photo-realistic Image Synthesis with Stacked Generative Adversarial Networks*. https://arxiv.org/abs/1612.03242v2

Fuentes de financiamiento: Esta investigación fue financiada con fondos del autor.

Declaración de conflicto de intereses: El autor declara que no tiene ningún conflicto de interés.