

APUNTES DE CLASE

Inteligencia Artificial — Semana 2 — 12 de agosto

Priscilla Jiménez Salgado

*Escuela de Ingeniería en Computación, Tecnológico de Costa Rica
Cartago, Costa Rica — 2021022576@estudiantec.cr*

Abstract—El presente documento recopila los apuntes de la segunda semana del curso de Inteligencia Artificial. Se introducen conceptos de *machine learning* y *deep learning*, los principales tipos de aprendizaje, el rol de la calidad de los datos y el ciclo completo de desarrollo, validación y despliegue de modelos.

I. INTRODUCCIÓN A LA INTELIGENCIA ARTIFICIAL

Se hace un repaso de los fundamentos de la Inteligencia Artificial (IA), sus características de autonomía y adaptabilidad, y su capacidad para resolver problemas complejos. Se destacan ejemplos prácticos y analogías para comprender el comportamiento de sistemas inteligentes.

A. Definición de la Inteligencia Artificial

La Inteligencia Artificial (IA) es, en pocas palabras, un conjunto de tecnologías capaces de realizar tareas que requieren inteligencia humana, como ver, escuchar, aprender y adaptarse. Su objetivo es resolver problemas complejos de forma eficaz y generar valor.

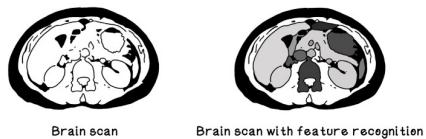
B. ¿Qué significa ser autónomo y adaptativo en esta área?

En IA, un sistema puede considerarse inteligente cuando combina dos capacidades clave: la autonomía para actuar sin instrucciones constantes y la adaptabilidad para modificar su comportamiento en respuesta a cambios del entorno o de las condiciones que se presenten.

C. Usos de la Inteligencia Artificial

La Inteligencia Artificial tiene aplicaciones en una gran variedad de áreas. Algunos ejemplos destacados son:

- **Medicina:** Uso de modelos de *machine learning* para analizar imágenes médicas, como escaneos cerebrales, y detectar patrones o anomalías que ayuden en el diagnóstico.



Brain scan with feature recognition

Fig. 1. Análisis de imágenes médicas con algoritmos de aprendizaje automático

- **Agricultura:** Análisis de datos sobre clima, tipo de planta, composición y humedad del suelo para optimizar qué cultivar, cuándo sembrar, cuándo regar y qué fertilizantes usar.

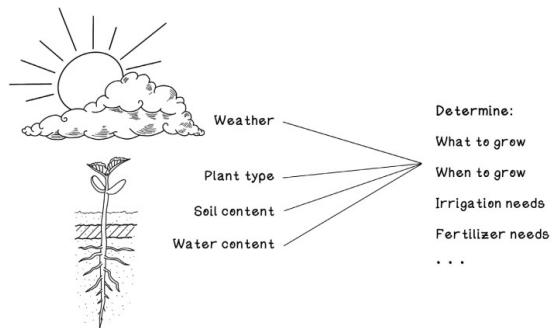


Fig. 2. Aplicación de datos para perfeccionar las prácticas de cultivo

- **Videojuegos:** En la imagen se muestra un ejemplo donde la IA analiza el entorno de un videojuego tipo *plataformas*, identificando obstáculos, enemigos y recompensas, para decidir qué acciones ejecutar mediante diferentes entradas y salidas de un modelo de red neuronal.

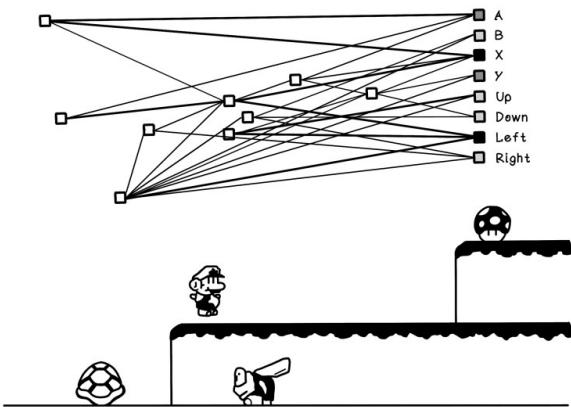


Fig. 3. IA identificando acciones óptimas en un entorno de plataformas

D. Modelos deterministas y estocásticos

En Inteligencia Artificial y Machine Learning, un modelo puede clasificarse según cómo responde a una entrada:

- **Determinista:** Para una entrada específica, siempre devuelve la misma salida. Ejemplo: determinar si hay luz de día a las 12 p.m.
- **Estocástico:** Para una entrada específica, puede devolver diferentes salidas de un conjunto de posibilidades, incorporando aleatoriedad. Ejemplo: predecir el clima exacto a las 12 p.m.

II. MACHINE LEARNING

El concepto de Machine Learning consiste en diseñar máquinas capaces de realizar tareas sin estar programadas de forma explícita, extrayendo la lógica directamente de los datos.

Por ejemplo, no se le indicará a la computadora qué es un perro ni las reglas que lo definen (cola, ojos, raza, etc.); en su lugar, se le proporcionará una imagen y retroalimentación para que aprenda por sí misma. Así, si se dispone de imágenes etiquetadas como X e Y, el modelo recibe cada imagen junto con su etiqueta y debe, de manera implícita, aprender a distinguir entre ambas, sin que se le explique el procedimiento.

En resumen, se le entrega únicamente la entrada y su etiqueta, y el modelo debe inferir o aproximar el resultado con la mayor precisión posible.

En este contexto, se cuenta con dos perspectivas: Ciencia e Ingeniería.

Machine Learning: Ciencia

- **Generar conocimiento:** Se refiere a investigar y desarrollar nuevas ideas para mejorar el aprendizaje automático. Esto puede incluir crear modelos desde cero, encontrar formas más eficientes de optimizar funciones o aprovechar datos sin etiquetar para entrenar sistemas.
- **Métricas:** Se emplean métricas para determinar qué modelo ofrece un mejor desempeño, evaluando su rendimiento en función del problema específico que se busca resolver.
- **Data Scientist:** Se encarga de trabajar con los datos desde su recolección hasta su preparación final, aplicando técnicas de análisis y manipulación para convertirlos en información comprensible y aplicable.
- **Research Scientist:** Centra su labor en investigar y proponer nuevos modelos, algoritmos y enfoques teóricos que impulsen el avance del Machine Learning. Esto incluye experimentar con aprendizaje no supervisado.



Fig. 4. Descripción general del uso de Machine Learning en la Ciencia.

En esta misma línea, Google publicó un artículo titulado *Towards an AI Co-Scientist*, en el que presenta una inteligencia artificial capaz de generar hipótesis científicas a partir del análisis de datos. Esta tecnología se probó en el ámbito médico y permitió obtener avances significativos en esa área.

Machine Learning: Ingeniería

En la etapa de **Puesta en producción de un modelo**, lo primero que se debe considerar es la necesidad de refactorizar el código, ya que será necesario seguir monitoreando el modelo e, incluso, volver a entrenarlo pasado cierto tiempo. Esto implica modificar su estructura para que pueda ofrecer la misma funcionalidad, pero de manera más eficiente. Por ejemplo, si se dispone de un modelo muy pesado, es fundamental **Transformar el modelo** para que pueda ser servido a un cliente de forma práctica.

Para lograrlo, existen diversas herramientas útiles. Una de ellas es **Onnx**, que permite tomar un modelo desarrollado en un framework específico y optimizarlo, generando una versión más ligera y compacta. También pueden aplicarse técnicas como el *model distillation*, donde un modelo más pequeño aprende a imitar el comportamiento de uno de mayor tamaño, logrando realizar las mismas tareas con menor peso. Otra estrategia es la que reduce la cantidad de bits en los parámetros del modelo. Estos parámetros, normalmente en punto flotante, se ajustan para mejorar el rendimiento, reduciendo el tamaño y acelerando su ejecución.

En el ámbito del **MLOps**, el objetivo es gestionar el modelo de forma más detallada. Un concepto clave es el *data shift*, que ocurre cuando el modelo fue entrenado con una distribución de datos específica, pero con el tiempo los patrones cambian, afectando su precisión. Por eso, una de las responsabilidades del equipo de **MLOps** es monitorear el comportamiento del modelo y, si detectan un cambio en los datos, volver a entrenarlo. Este monitoreo continuo no solo incluye el rendimiento del modelo, sino también la calidad y actualidad de los datos, la infraestructura de despliegue y la trazabilidad de las versiones utilizadas.



Fig. 5. Descripción general del uso de Machine Learning en la Ingeniería.

Ahora bien, consideremos el siguiente gráfico:

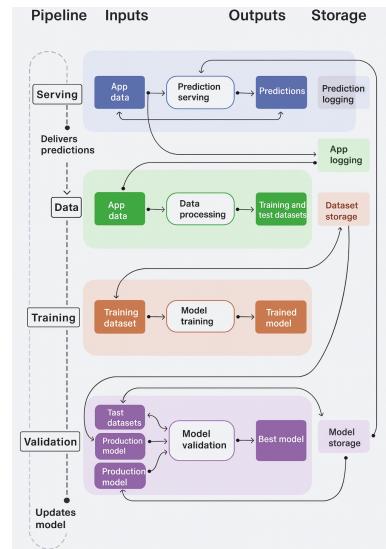


Fig. 6. Síntesis del Machine Learning en el campo de la Ingeniería.

El proceso comienza con la **recolección de datos**. A partir de ellos, se genera un conjunto estructurado (*dataset*) que se dividirá en dos partes: una para el **entrenamiento** y otra para las **pruebas**. Esta información se almacena para su uso posterior.

En la fase de **entrenamiento**, el modelo aprende a partir del conjunto de datos de entrenamiento, ajustando sus parámetros según el fenómeno que se desea modelar. Durante esta etapa también se define qué tipo de modelo utilizar, justificando la elección en función de las necesidades del problema.

Luego se pasa a la **validación**, donde se utilizan datos que el modelo no ha visto antes para evaluar su rendimiento en condiciones similares a las de producción. Esto permite identificar qué modelo obtiene los mejores resultados según los criterios de evaluación establecidos.

El modelo seleccionado se guarda en un repositorio o **almacenamiento de modelos**, desde el cual se prepara para su implementación en el entorno elegido.

Por último, se lleva a cabo el **despliegue o servicio de predicciones**, donde el modelo ya entrenado y validado comienza a procesar datos reales y a generar

resultados. Todo este flujo puede representarse mediante un diagrama que describe cada paso, desde la **obtención inicial de los datos** hasta la **puesta en marcha del sistema en producción**.

III. JERARQUÍA DE CONCEPTOS EN IA

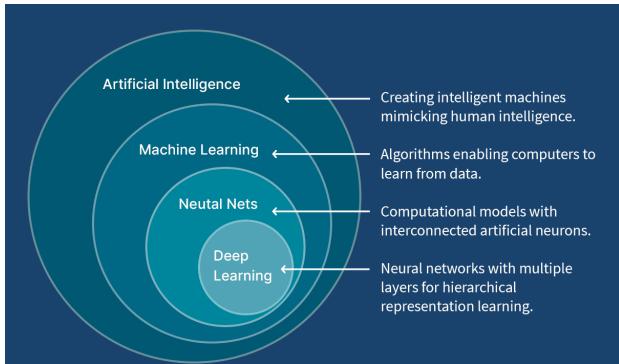


Fig. 7. Relación entre IA, Machine Learning y Deep Learning

- **Machine Learning**: utiliza datos para que un sistema pueda entrenarse y mejorar su rendimiento. Para lograrlo, emplea métodos como algoritmos estadísticos, regresión lineal, regresión logística o árboles de decisión. Más allá de la técnica usada, su base siempre es la misma: aprender a partir de los datos disponibles.
- **Deep Learning**: es una especialidad dentro del Machine Learning que utiliza redes neuronales profundas, formadas por varias capas conectadas. Este enfoque es muy efectivo para resolver tareas complejas, aunque requiere grandes volúmenes de datos para funcionar correctamente.

En resumen, las diferencias es que la Inteligencia Artificial es el concepto más amplio, que engloba todo lo relacionado con lograr que una máquina actúe de forma “inteligente”. El Machine Learning agrupa algoritmos que aprenden con datos, y el Deep Learning es una técnica dentro de este que está diseñada para trabajar con cantidades masivas de información y resolver tareas específicas con gran precisión.

IV. TIPOS DE APRENDIZAJE EN MACHINE LEARNING

Se hizo un repaso de la clase anterior y se retomaron los demás tipos de aprendizaje y se describen a

continuación.

- **Supervisado**: El modelo aprende a partir de datos que incluyen etiquetas, las cuales sirven como referencia durante el entrenamiento. Un ejemplo común es la clasificación de imágenes.
- **No supervisado**: El modelo trabaja con datos sin etiquetas y se encarga de encontrar patrones en los datos ocultos. Un ejemplo claro de esto son los clusters.
- **Semi-supervisado**: Combina datos etiquetados y no etiquetados, útil cuando el proceso de etiquetado es costoso o difícil de realizar.
- **Auto-supervisado**: El propio dato de entrada sirve como etiqueta. Se emplea en autoencoders y modelos de representación, principalmente para reducir el tamaño de vectores o extraer características relevantes.
- **Aprendizaje por refuerzo**: El modelo, llamado agente, aprende mediante un sistema de recompensas, mejorando su desempeño a través de la interacción con un entorno.
- **Few-Shot Learning**: El modelo necesita solo unos pocos ejemplos para aprender a realizar una tarea específica.
- **One-Shot Learning**: Basta con mostrarle una única vez cómo realizar la tarea para que el modelo pueda reproducirla.
- **Zero-Shot Learning**: El modelo es capaz de realizar tareas sin haber sido entrenado previamente para ellas de forma específica.

V. PIPELINE DE MACHINE LEARNING

El desarrollo de un modelo de Machine Learning pasa por varias etapas clave:

- 1) **Data Adquisition**: Este es muy importante ya que se necesitan obtener datos de calidad y representativos, evitando errores como valores faltantes y duplicados.
- 2) **Data preparation**: Para esta etapa se tiene que preparar los datos que se tienen para tener un

dataset de calidad, ya sea, eliminando datos duplicados o se descartan datos que no tienen utilidad.

- 3) **Feature Engineering:** Esta fase consiste en crear nuevas variables útiles a partir de los datos que se encuentran disponibles o eliminar aquellas que no aporten información relevante, especialmente en el caso de datos tabulares.
- 4) **Modelo Selection:** Elegir el modelo más adecuado según el problema y recursos disponibles, puede ir desde opciones simples como regresión logística hasta redes neuronales profundas para casos complejos.
- 5) **Model training:** Esta fase de entrenamiento del proceso del modelo, se dividen los datos en training set y validation set y se ajustan hiperparámetros con métodos como Grid Search.
- 6) **Model Deployment:** En este proceso de deployment se implementa el modelo en producción y se supervisa para poder garantizar un buen rendimiento.

VI. HIPERPARÁMETROS Y PARÁMETROS

Es importante diferenciar entre **parámetros** e **hiperparámetros**:

- **Parámetros:** son los valores internos que un modelo aprende automáticamente a partir de los datos durante el entrenamiento. Estos valores son los que el modelo ajusta para minimizar el error y mejorar su capacidad de predicción.
- **Hiperparámetros:** Son valores definidos manualmente antes de iniciar el entrenamiento, y controlan el comportamiento del algoritmo. Un ejemplo es el *batch size* (por ejemplo, 32 muestras por iteración). Estos no se aprenden del conjunto de datos, sino que se configuran para guiar el proceso de entrenamiento.

El ajuste de hiperparámetros requiere un proceso de **experimentación**, probando distintas configu-

raciones hasta encontrar la que produce el mejor rendimiento del modelo.

VII. PARADIGMAS DE RESOLUCIÓN DE PROBLEMAS EN MACHINE LEARNING

- 1) **Agrupamiento:** Busca patrones o relaciones ocultas en los datos para formar grupos, útil para descubrir conexiones que no se habían considerado.

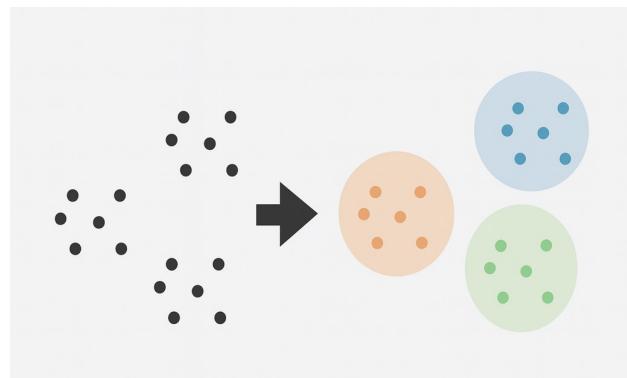


Fig. 8. Ejemplo del proceso de agrupamiento

2) Predicción y clasificación:

- **Predicción:** Estima un valor numérico basándose en patrones de los datos.
- **Clasificación:** Asigna datos a una categoría según sus características.



Fig. 9. Ejemplo del proceso de predicción y clasificación

3) **Optimización:** Encuentra la mejor solución entre muchas posibles.

- **Local:** La mejor en un área concreta.
- **Global:** La mejor en todo el espacio de búsqueda.

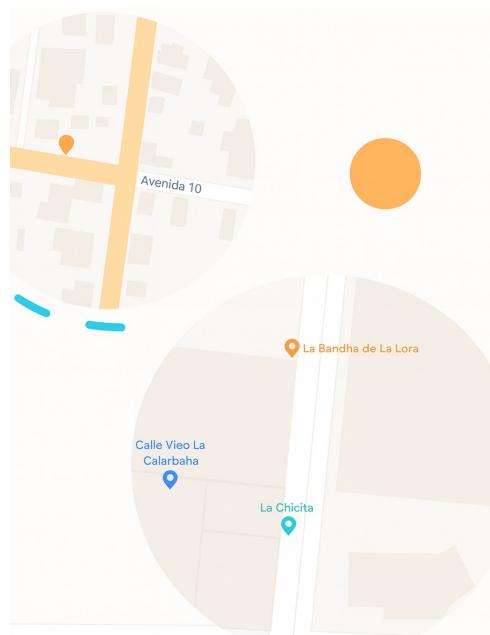


Fig. 10. Ejemplo del proceso de Optimización

4) **Búsqueda:** Encuentra el camino más eficiente hacia una solución, representando la mejor opción como el camino de menor costo.

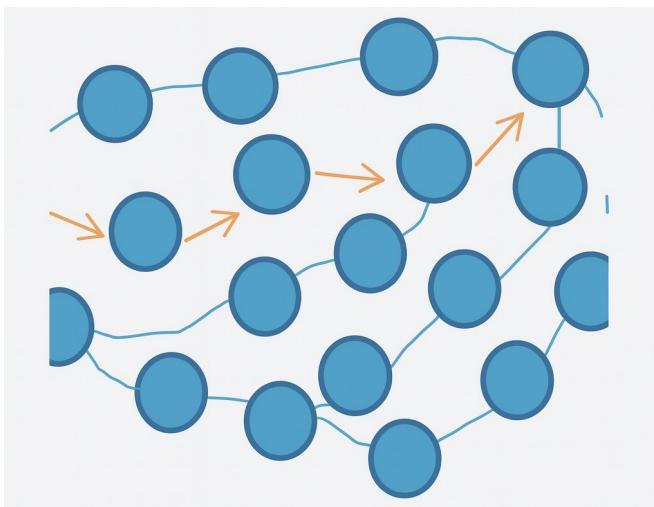


Fig. 11. Ejemplo del proceso de búsqueda

VIII. NOTA IMPORTANTE

• **Enunciado Tarea Moral** Mencionar un aporte de cada uno de ellos:

- Yann LeCun
- Yoshua Bengio
- Sam Altman
- Geoffrey Hinton
- Timnit Gebru
- Ian Goodfellow

Incluya un resumen del funcionamiento de las siguientes herramientas/conceptos:

- Onnx
- MLFlow
- Vertex
- LangChain
- HuggingFace
- Ollama
- Chain-of-Thought

Fecha de entrega: 19 de Agosto, pero esta es moral.

IX. ASPECTOS ADMINISTRATIVOS

La próxima sección corresponde a **Álgebra Lineal**, la cual es fundamental para los temas que se abordarán más adelante.

La próxima semana la modalidad será **virtual**. Hasta el **martes 26 de agosto** las clases serán **presenciales** y ese martes habrá **quiz acumulativo**.