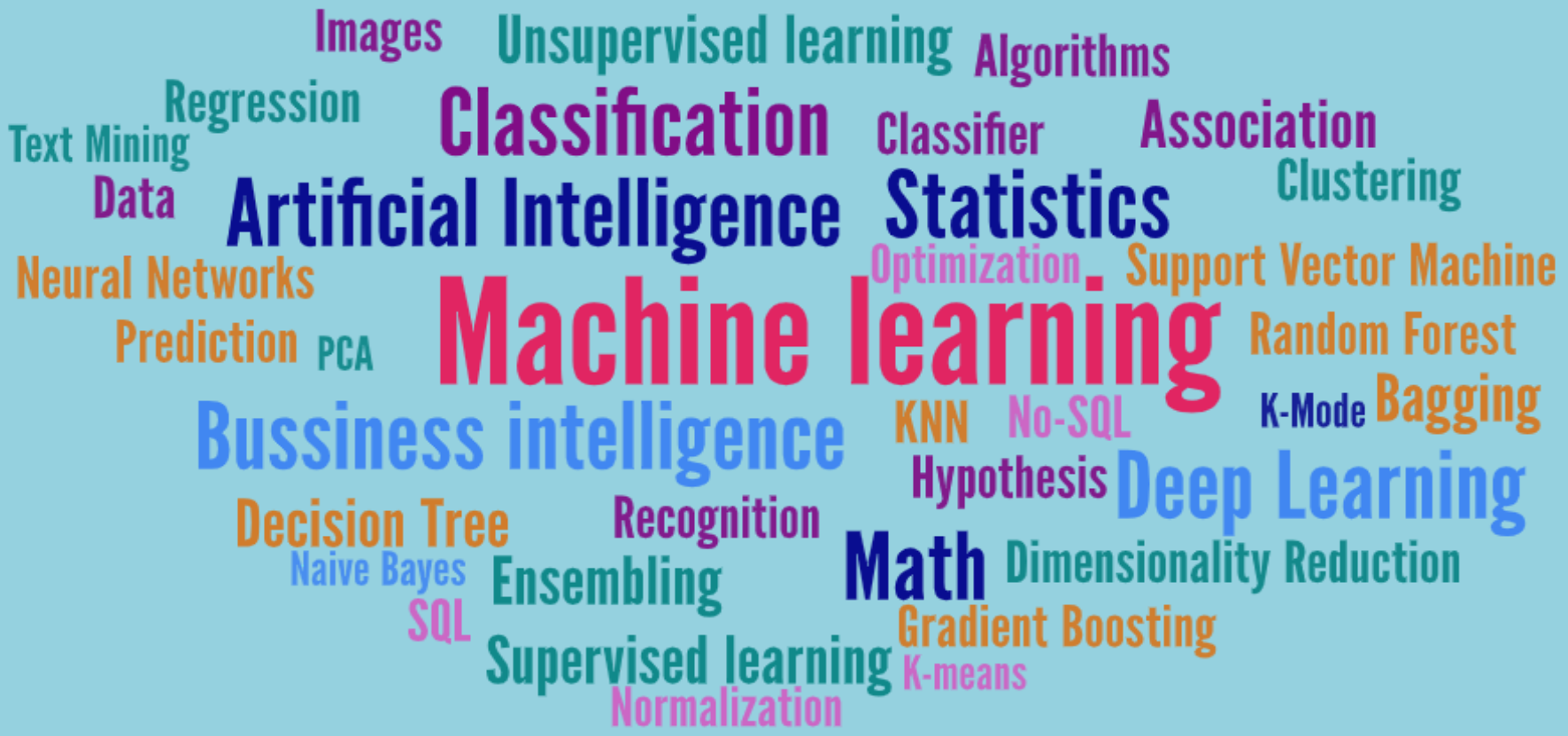
Clase 1: Introducción al Machine Learning

Herramientas Computacionales para Data Science

Jonathan Acosta - Danilo Alvares - Luis Castro

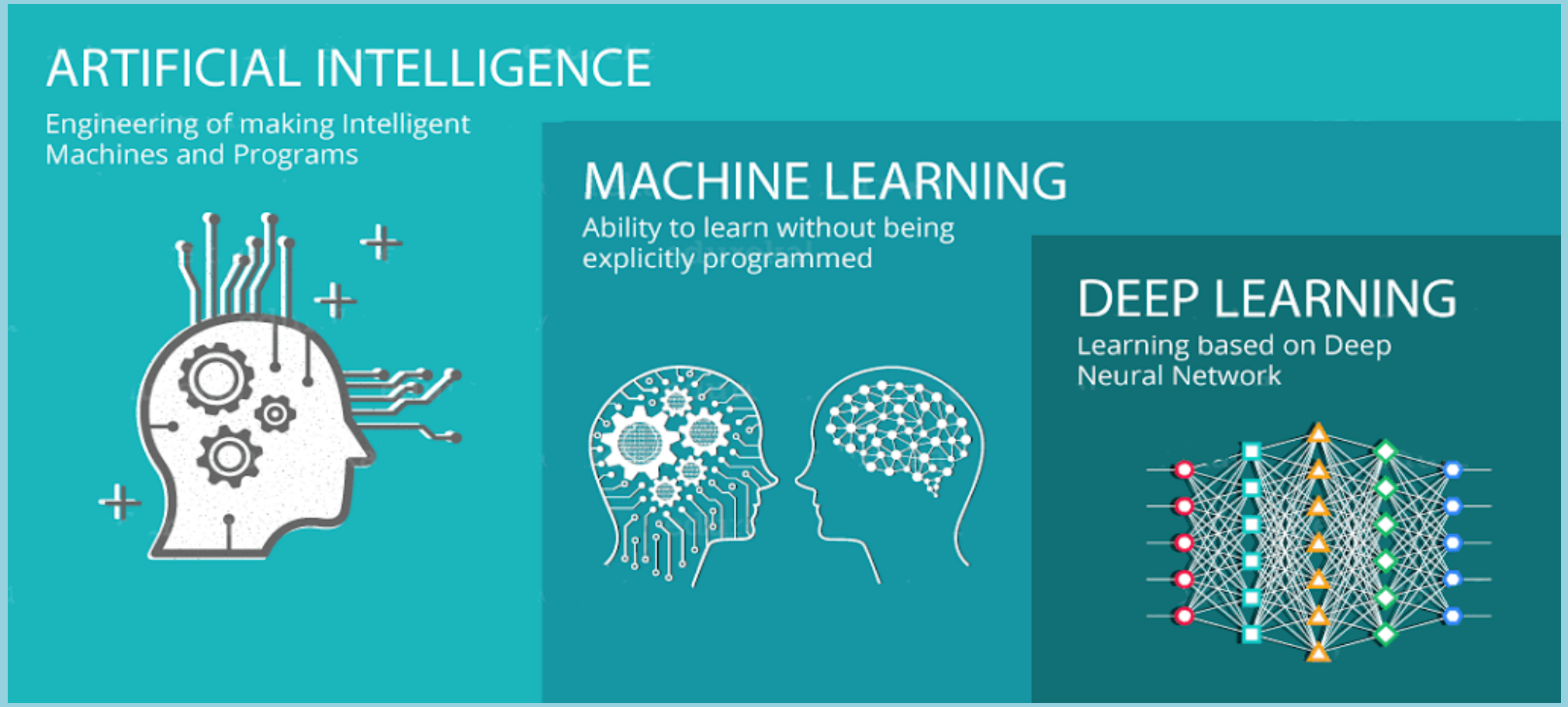
¿Qué aprenderemos?



Origen del Machine Learning

- Arthur Samuel (1901-1990) propone en 1959 darle a las máquinas y computadoras la habilidad de aprender sin ser explícitamente programadas.
- El informático fue pionero en el área de la Inteligencia Artificial y en comprender que observar patrones ayudaba a replicarlos.
- Gracias al desarrollo teórico de estadísticos e informáticos, el Machine Learning se hizo muy famoso en los años 90.
- La intersección de la informática y las estadísticas dio lugar a un enfoque probabilístico en la Inteligencia Artificial.

Machine Learning & Artificial Intelligence



El Machine Learning tiene como objetivo identificar patrones a partir de los datos con el fin de hacer predicciones, detecciones o clasificaciones.

Aplicaciones del Machine Learning

Área	Ejemplo
Reconocimiento de imágenes	Detección de personas sin mascarilla.
Clasificación	Análisis de clientes para evaluación de entrega de crédito bancario.
Reconocimiento de voz	Atención al cliente mediante llamadas telefónicas.
Optimización	Determinar rutas óptimas en tiempo real.
Reconocimiento de imágenes	Detección de espacios vacíos en góndolas de supermercados.
Predicción	Detección de vuelos con alta probabilidad de atraso.
Optimización	Determinar dotación de camiones óptima según demanda minera.

Entre otras diversas aplicaciones.

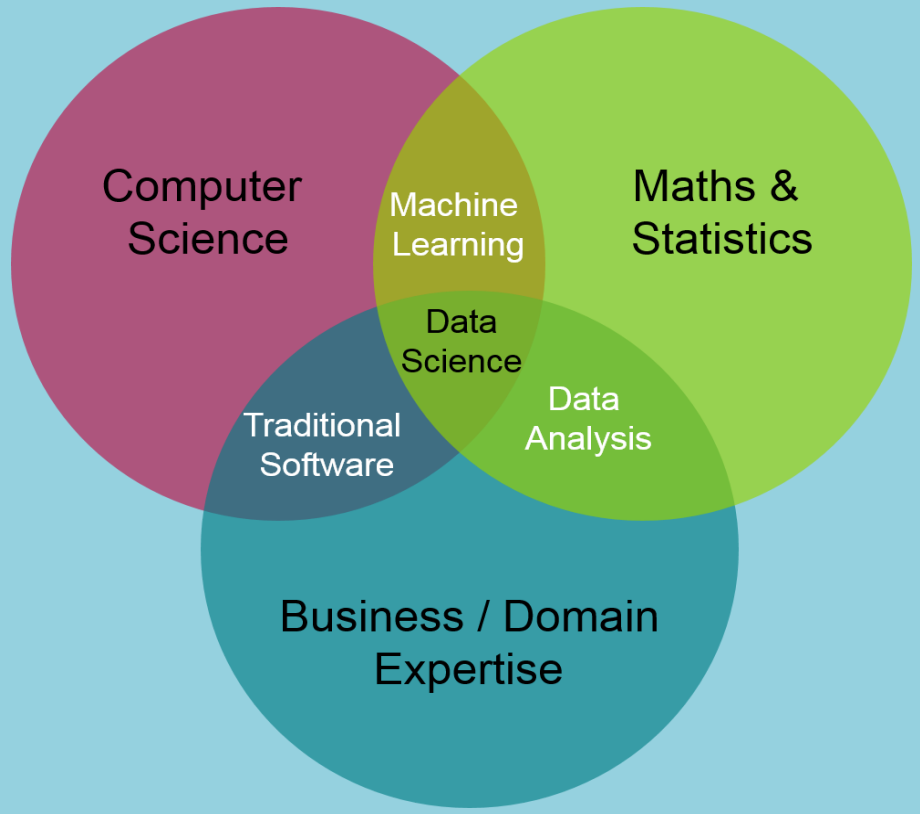
¿Qué necesitamos para resolver problemas analíticos?

Para resolver cualquiera de los ejemplos anteriores, se necesitan conocimientos de:

- Big Data: Volumen masivo de datos (las cinco V del Big data son Volumen, Variedad, Velocidad, Veracidad y Valor).
- Data Mining: Extracción de información a partir de los datos.
- Business Intelligence: Análisis descriptivo, muestra lo que sucede/sucedió, visualización y comunicación adecuada de los datos en el contexto del negocio.
- Machine Learning: Algoritmos que acceden y aprenden de los los datos.
- Deep Learning: Algoritmos que buscan comprender estructuras abstractas y complejas de los datos.
- Business Analytics: Técnicas predictivas, determina la probabilidad de resultados futuros con interpretación en el negocio.

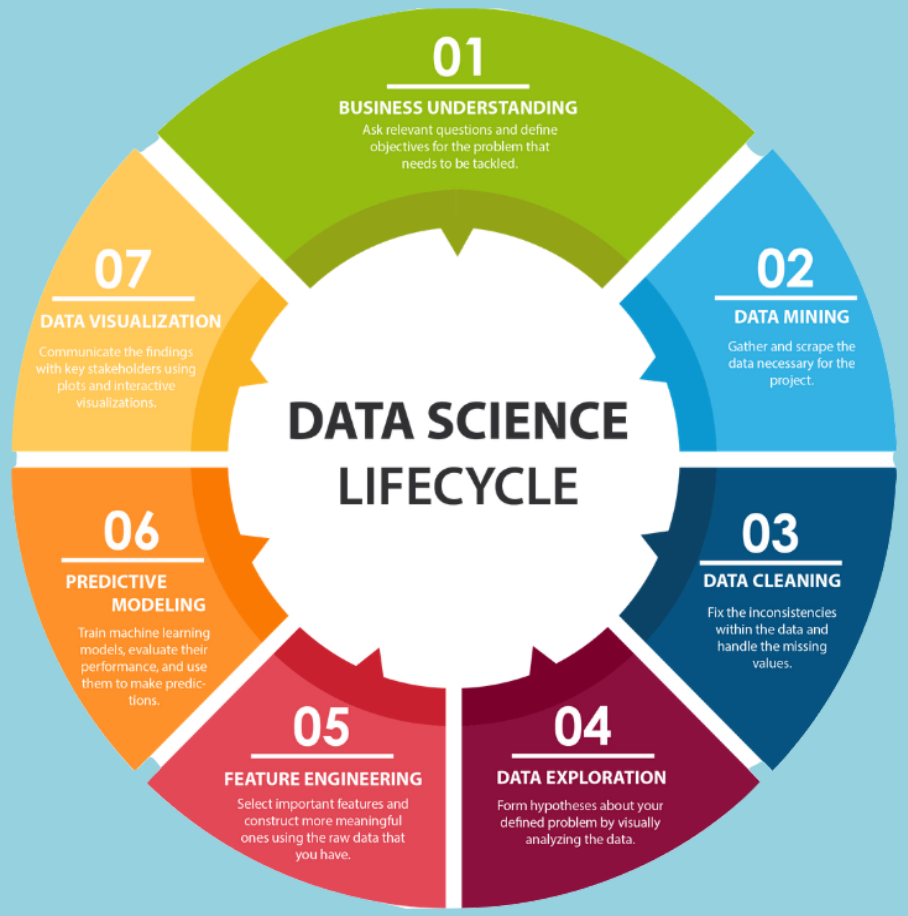
La Ciencia de datos es interdisciplinar

La ciencia de datos es una combinación de varias herramientas, algoritmos y principios, conformando un estudio interdisciplinar de los datos con el fin de generar conocimiento.



El ciclo de un Proyecto de Data Science

- Hacer las preguntas correctas, determinar un dolor, una necesidad del negocio.
- Obtener y recopilar datos, estructurar los datos limpios en un formato apropiado y realizando reducción de la dimensionalidad, estandarización, normalización, entre otras.
- Realizar gráficos, análisis exploratorio, generar intuiciones de negocio. Determinar los features a utilizar, crear o transformar variables.
- Modelar el algoritmo de Machine Learning, se construye con los datos de entrenamiento. Evaluar el modelo obtenido y ajustarlo para maximizar su rendimiento.



- A partir del modelo creado, obtener inferencias e insights que agreguen valor de negocio.

Herramientas útiles para un proyecto de Data Science

Big Data

- *Hadoop* es un framework open source para almacenar datos y ejecutar aplicaciones en clusters de computadores. Proporciona un almacenamiento masivo para cualquier tipo de datos, un enorme poder de procesamiento y la capacidad de manejar tareas o trabajos prácticamente ilimitados.

- *Amazon Web Services* es un proveedor de servicios de almacenamiento, recursos de computación, aplicaciones móviles, bases de datos y otros, en la nube.

- *Apache Spark* se considera el primer software open source que hace la programación distribuida (se distribuye el trabajo en clusteres que trabajan como uno). Se pueden programar aplicaciones usando diferentes lenguajes como Java, Scala, Python o R.

Herramientas útiles para un proyecto de Data Science

Data Mining

- *R* es un lenguaje de programación vastamente utilizado a la hora de realizar minería de datos, análisis descriptivo, limpieza, incluso creación de gráficas, reportes automáticos y dashboards.

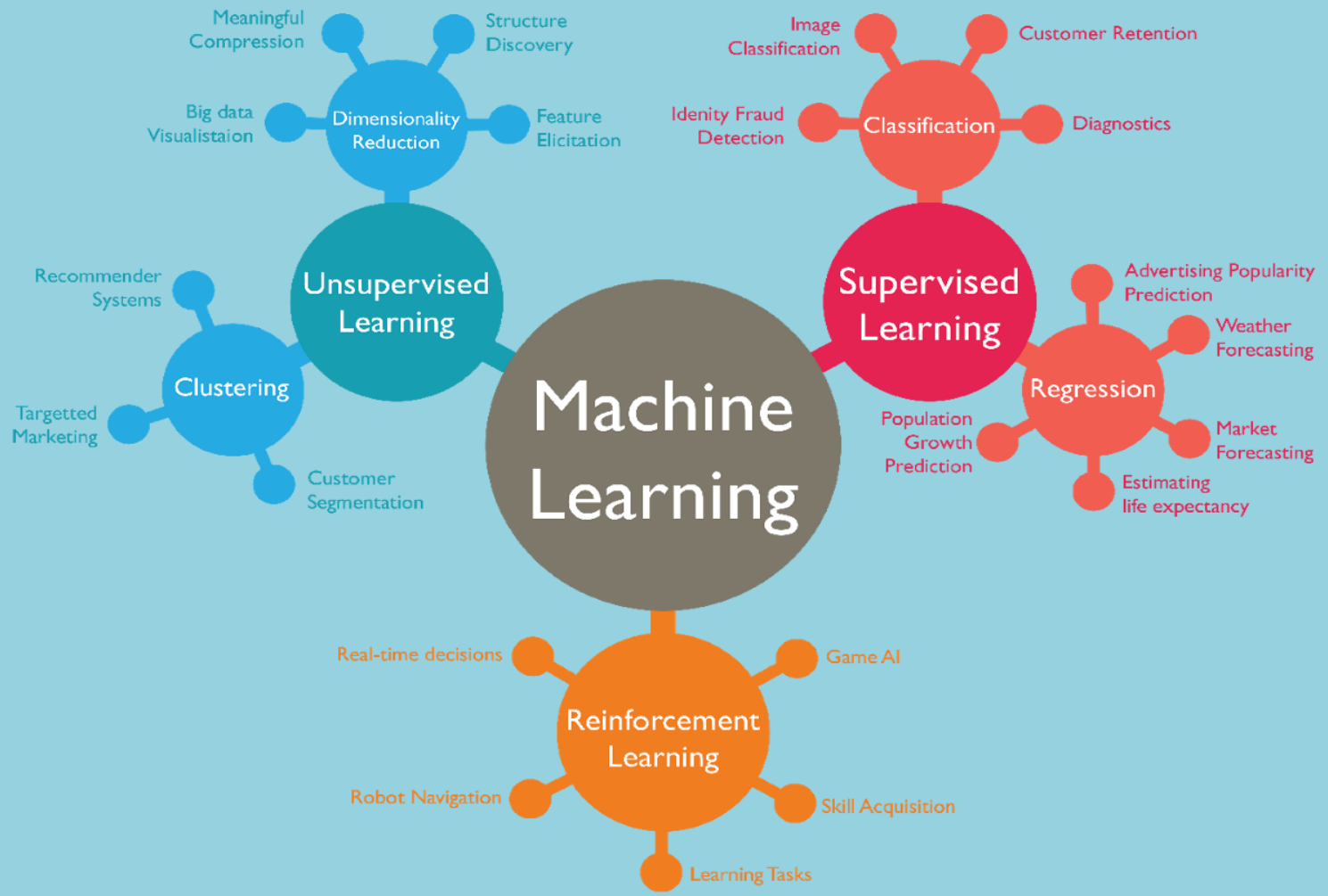
- *Python* es un lenguaje de programación que destaca por su versatilidad y fácil integración con otras aplicaciones y plataformas.

Visualización: Business Intelligence

- *Power BI* y *Tableau* herramientas que permiten crear informes con múltiples visualizaciones interactivas mediante una interfaz sencilla de utilizar.

Los modelos son algoritmos aplicados a un conjunto de datos determinado. Un algoritmo se puede clasificar según la manera en la que aprende de los datos.

Tipos de Aprendizaje



Aprendizaje No Supervisado

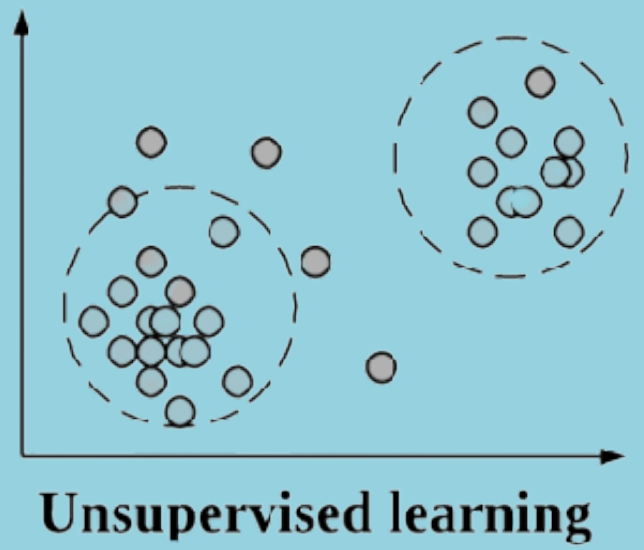
- Un algoritmo tiene un Aprendizaje No Supervisado cuando se busca *identificar grupos o patrones a partir de la similitud de sus características*.
- El algoritmo aprende a partir de la variable explicativa sin ninguna variable respuesta asociada, lo que le permite determinar los patrones de datos por sí mismo. Los algoritmos se dejan guiar por sus propios mecanismos para descubrir la estructura de datos.
- Algunos algoritmos de Aprendizaje No Supervisado:
  - K - Medias
  - K - Medians
  - Sistemas de Recomendación

Aprendizaje Supervisado

- Un algoritmo tiene un Aprendizaje Supervisado cuando se busca predecir casos futuros a partir de datos cuya respuesta *ya se conoce.*
- El algoritmo aprende a partir de las variables explicativas asociadas a la variable respuesta (numérica o categórica). Luego, predice el valor de la variable respuesta cuando se presentan nuevas variables explicativas.
- Algunos algoritmos de Aprendizaje Supervisado:
    - Regresión Lineal
    - Vecino más cercano
    - Árbol de Decisión

Aprendizaje Supervisado y No Supervisado

Supervised learning



Unsupervised learning

Aprendizaje de Refuerzo

- Un algoritmo tiene un aprendizaje de refuerzo cuando tomar decisiones usando un sistema de *recompensa y castigo*.
- El algoritmo aprende interactuando con su entorno, donde recibe recompensas por tomar la decisión adhoc y castigos por errar. Aprende maximizando su recompensa y minimizando su penalización.
- Un ejemplo muy sencillo de este método de aprendizaje es entrenar un algoritmo para que juegue ajedrez, el hecho de que el algoritmo gane una partida y reciba una recompensa positiva no indica que los movimientos de piezas realizados sean los correctos, simplemente indica que éstos fueron adecuados en el entorno específico en el que se produjeron.

¿Qué aprenderemos en las próximas clases?

Nos enfocaremos en los algoritmos de Aprendizaje Supervisado y No Supervisado.

Algunos algoritmos que aprenderemos:

- Decision Tree
- Random Forest
- K-Mode
- K-prototype
- Naive Bayes
- Support Vector Machine
- LDA
- QDA
- Neural network

¡Gracias!