**Ciencia de datos**

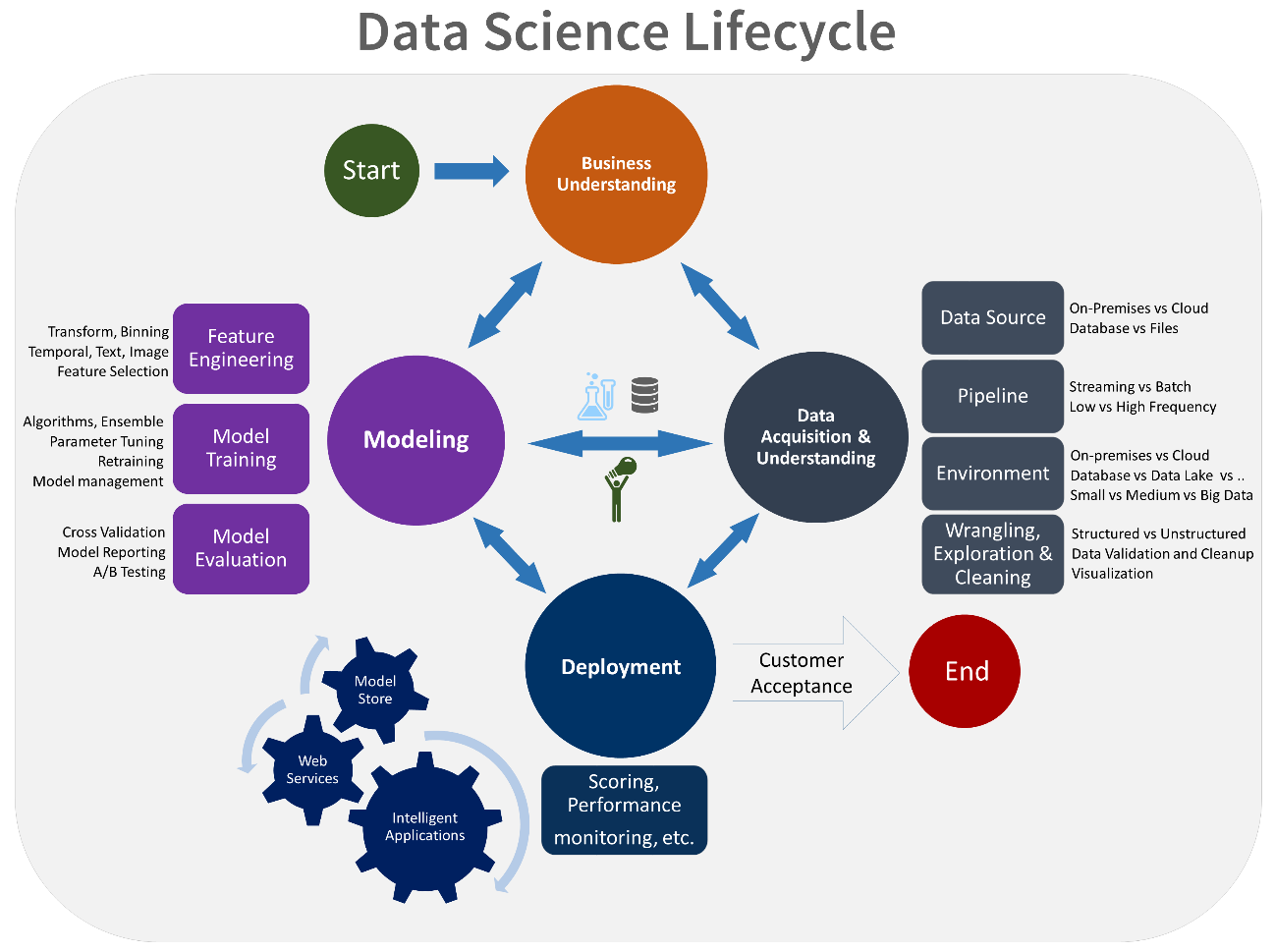
es un campo interdisciplinario que, con métodos científicos, **extrae conocimiento** de datos estructurados o no estructurados​. Es una **combinación** de campos de análisis de datos como la estadística, la minería de datos, el aprendizaje automático, y la analítica predictiva.

El concepto ciencia de datos **reúne** estadísticas, análisis de datos, aprendizaje automático, y sus métodos relacionados, para comprender y analizar los fenómenos reales. Emplea técnicas y teorías extraídas de muchos campos dentro del contexto de las matemáticas, la estadística, la ciencia de la información, y la informática.

Esta centrada en el **análisis** de grandes fuentes de datos para extraer información, comprender la realidad y descubrir patrones con los que tomar decisiones.

**Ciclo de vida** data science

* Bussiness problem
* Adquirir los datos: api, databases
* Preparar los datos: limpiar y transformar
* Análisis exploratorio: ver que variables son importantes
* Modelar los datos: knn, decisión tree. Ver que modelo encaja mejor. python
* Visualizar y comunicar: reportes power bi
* Poner en producción y mantener



**Aprendizaje automático**

es el subcampo de las ciencias de la computación y una rama de la inteligencia artificial, cuyo objetivo es **desarrollar técnicas** que permitan que las computadoras aprendan. Se dice que un agente aprende cuando su desempeño mejora con la experiencia y mediante el uso de datos; es decir, cuando la habilidad no estaba presente en su genotipo o rasgos de nacimiento. "En el aprendizaje de máquinas un computador observa datos, **construye un modelo** basado en esos datos y utiliza ese modelo a la vez como una hipótesis acerca del mundo y una pieza de software que puede resolver problemas"

A través de algoritmos, dota a los ordenadores de la **capacidad** de identificar patrones en datos masivos y elaborar predicciones

**Aprendizaje supervisado**

es una técnica para deducir una función a partir de **datos de entrenamiento**. Los datos de entrenamiento consisten en **pares de objetos** (normalmente vectores): una componente del par son los datos de entrada y el otro, los resultados deseados.

La **salida de la función** puede ser un valor numérico (como en los problemas de regresión) o una etiqueta de clase (como en los de clasificación).

El objetivo del aprendizaje supervisado es el de crear una función capaz de **predecir** el valor correspondiente a cualquier objeto de entrada válida después de haber visto una serie de ejemplos, los datos de entrenamiento. Para ello, tiene que **generalizar** a partir de los datos presentados a las situaciones no vistas previamente

Por ejemplo, un proceso de aprendizaje supervisado podría consistir en **clasificar vehículos** de dos y cuatro ruedas a partir de sus imágenes. Los datos de entrenamiento tendrían que estar correctamente etiquetados para identificar si un vehículo es de dos o cuatro ruedas. El aprendizaje supervisado permite que los algoritmos 'aprendan' de datos históricos/de entrenamiento y los apliquen a entradas desconocidas para obtener la salida correcta.

regresión lineal y logística, los árboles de decisión, knn

**Aprendizaje no supervisado**

es un método de Aprendizaje Automático donde un modelo se ajusta a las **observaciones**. Se distingue del Aprendizaje supervisado por el hecho de que **no hay un conocimiento** a priori. En el aprendizaje no supervisado, un conjunto de datos de objetos de entrada es **tratado**. Así, el aprendizaje no supervisado típicamente trata los objetos de entrada como un conjunto de variables aleatorias, siendo construido un **modelo de densidad** para el conjunto de datos. Tiene datos sin etiquetar que el algoritmo tiene que intentar entender por sí mismo

los modelos no aprenden a partir de los llamados datos de entrenamiento. Son los propios modelos sin supervisión los que **encuentran los patrones** subyacentes en los datos a analizar.

Agrupamiento y kmeans

**Aprendizaje por refuerzo**

es un área del aprendizaje automático inspirada en la psicología conductista, cuya ocupación es determinar qué acciones debe escoger un **agente de software** en un entorno dado con el fin de maximizar alguna noción de "recompensa" o premio acumulado

Su principal particularidad es que es capaz de funcionar sin grandes cantidades de datos de entrenamiento. Tan “sólo” necesita una **serie de indicaciones** para ir aprendiendo a través de prueba y error. A diferencia del aprendizaje supervisado basado en un conjunto de datos que le indica a la máquina qué debe hacer, aquí se utilizan **recompensas** para reforzar el comportamiento deseado.

En un sistema de aprendizaje por refuerzo, un **agente explora** un entorno desconocido y determinar las acciones a llevar a cabo mediante prueba y error. Será capaz de aprender por sí sólo obteniendo recompensas o penalizaciones, no sólo de manera inmediata sino buscando maximizar la recompensa a la larga.

El aprendizaje por refuerzo puede ser usado en **robots**, por ejemplo en brazos mecánicos en donde en vez de enseñar instrucción por instrucción a moverse, podemos dejar que haga intentos “a ciegas” e ir recompensando cuando lo hace bien.

Otro caso de uso que está ganando terreno es el de usar RL para crear **“webs personalizadas”** para cada internauta. Y si lo piensas… tiene algo de sentido tomar el concepto de “premiar” al algoritmo si acierta con las sugerencias que hace al usuario si hace clic ó penalizar al modelo si sus recomendaciones no le son de utilidad.

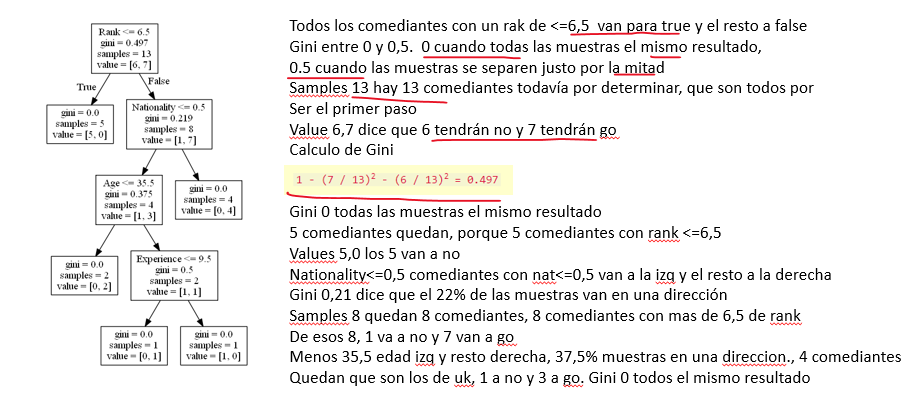
**Ejemplos supervisado/no supervidado**

Si una computadora estuviera aprendiendo a identificar **frutas** en un entorno de aprendizaje **supervisado**, se le darían imágenes de ejemplo de frutas etiquetadas, a esto se le llama datos de entrada. Por ejemplo, las **etiquetas** dirían que los plátanos son largos, curvos y amarillos, que las manzanas son redondas y rojas, mientras que una naranja es esférica, de aspecto ceroso y anaranjada. Después de un tiempo conveniente, la máquina debería poder identificar con seguridad qué fruta es cuál, basándose en esos descriptores. Si se le presenta una manzana, por ejemplo, podría decir con seguridad que no es de color naranja, por lo tanto, no es una naranja, pero también que no es amarilla y larga, por lo tanto, no es un plátano. Entonces, será una manzana debido a que es redonda y roja.

Por el contrario, el aprendizaje **no supervisado** es cuando no existe ninguna categorización o etiquetado de los datos. La máquina no tendrá idea del concepto de fruta, por lo que no podrá etiquetar los objetos. Sin embargo, podrá **agruparlos** según sus colores, tamaños, formas y diferencias. La máquina agrupará las cosas de acuerdo con las similitudes, encontrando estructuras y patrones ocultos en datos sin etiquetar. No existe un camino correcto o incorrecto, ni tampoco un maestro. No existen resultados, solo un análisis puro de los datos.

Un ejemplo de aprendizaje supervisado sería la construcción de un **modelo de reingresos** en hospitalización partiendo de un **conjunto de datos** previo de los que conocemos si el paciente reingresó o no (el atributo que nos indique la condición de reingreso en el conjunto de datos original sería la etiqueta). Un ejemplo de aprendizaje **no supervisado** sería la de segmentar los pacientes que han sido atendidos en urgencias en **grupos homogéneos** pero sin un conocimiento previo de los grupos que queremos obtener; lo haríamos a partir de estructuras no evidentes subyacentes en los datos.

**Árbol de decisión**



**Índice de Gini**

medir cualquier forma de **distribución desigual**. El coeficiente de Gini es un número entre 0 y 1, donde 0 se corresponde con la **perfecta igualdad** (todos tienen los mismos ingresos) y donde el valor 1 se corresponde con la perfecta desigualdad (una persona tiene todos los ingresos y los demás ninguno)

**Entropía e información ganada**

La Entropia mide la **incertidumbre** de una fuente de información. Mide la impureza

Para la Información ganada, cuando partimos el conjunto en partes mas pequeñas la entropía cambia. Mide este **cambio en entropía**

1 bola roja, 3 bolas moradas, 4 bolas amarillas

Texto

Descripción generada automáticamente con confianza media

Ahora 5 bolas moradas, 5 bolas amarillas



Imagen que contiene Diagrama

Descripción generada automáticamente

Una rama con 4= 4 moradas y otra con 6=1 morada y 5 amarillas



Imagen de la pantalla de un celular con letras

Descripción generada automáticamente con confianza baja

Por haber 4 en una rama y 6 en otra

Gráfico

Descripción generada automáticamente con confianza baja

Diagrama

Descripción generada automáticamente con confianza baja

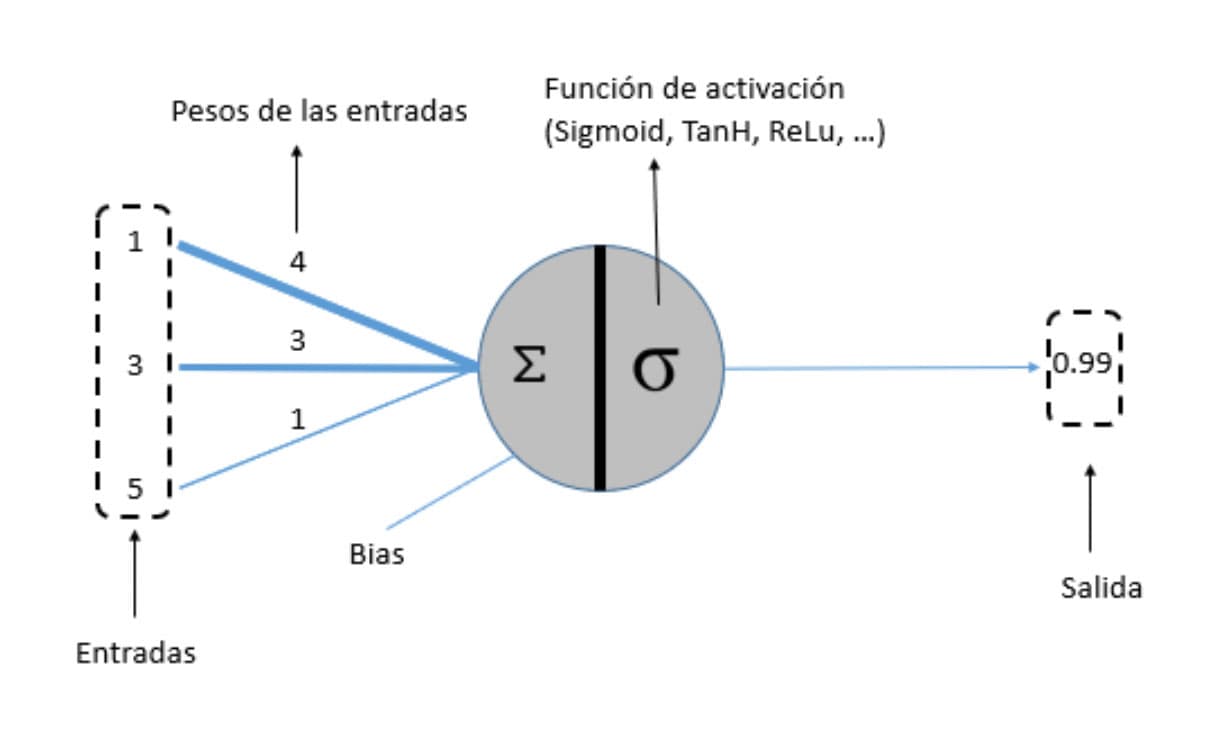
**Redes neuronales artificiales**

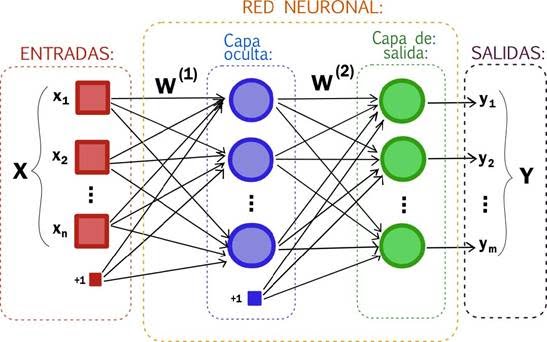
Consiste en un conjunto de unidades, llamadas **neuronas artificiales**, conectadas entre sí para transmitirse señales. La **información de entrada** atraviesa la red neuronal (donde se somete a diversas operaciones) produciendo unos valores de salida.

Cada neurona está **conectada** con otras a través de unos enlaces. En estos enlaces el valor de salida de la neurona anterior es multiplicado por un valor de **peso**. Estos pesos en los enlaces pueden incrementar o inhibir el estado de activación de las neuronas adyacentes. Del mismo modo, a la salida de la neurona, puede existir una **función limitadora** o umbral, que modifica el valor resultado o impone un límite que no se debe sobrepasar antes de propagarse a otra neurona. Esta función se conoce como función de activación.(0 o 1)

En el caso de las neuronas artificiales, la **suma** de las entradas multiplicadas por sus pesos asociados determina el “impulso nervioso” que recibe la neurona. Este valor, se procesa en el interior de la célula mediante una **función de activación** que devuelve un valor que se envía como salida de la neurona.

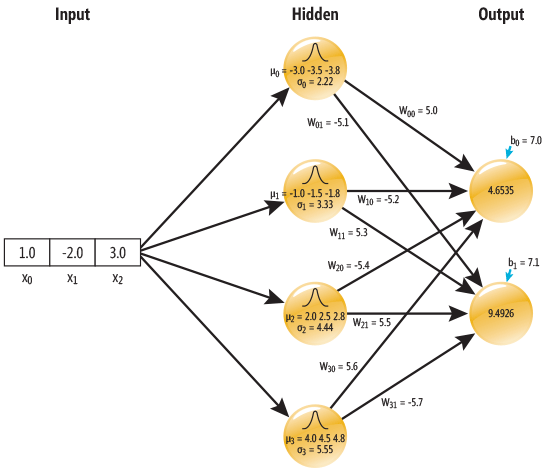
Si queremos conseguir que la red neuronal sea capaz de generalizar e **identificar gatos** en cualquier imagen, es importante utilizar un elevado número de imágenes para realizar el entrenamiento, tanto de imágenes que son gatos (etiquetadas como 1) como de imágenes que no son gatos (etiquetadas como 0), incluyendo la mayor **variabilidad posible**. Con esto, la red será capaz de ajustar sus **parámetros** para satisfacer en la medida de lo posible todas las imágenes, por lo que será capaz de extraer de manera precisa las características que identifican la presencia de un gato en una imagen.





**Rbf**

Una red RBF acepta uno o más **datos numéricos** valores, tales como (-2,0, 1.0, 3.0) y genera uno o más valores de salida numérica, tales como (4.6535, 9.4926). Redes RBF (a veces denominadas redes radiales) pueden utilizarse para clasificar los datos y realizar **predicciones**. Por ejemplo, una red RBF podría utilizarse para predecir las **puntuaciones** de los dos equipos de fútbol que están programados para jugar entre sí, basado en datos históricos como el porcentaje de victorias actual de cada equipo, ventaja (-1,0 o + 1,0) y así sucesivamente. O una red RBF podría ser utilizada para clasificar el **riesgo del paciente** de un hospital de cáncer (bajo = 0, alto = 1) basado en los valores de resultados de las pruebas médicas y otros factores como la edad y sexo.



**Adaline**

Generalmente se compone de **una sola capa** de n neuronas ( por tanto n valores de salida ) con m entradas con las siguientes características:

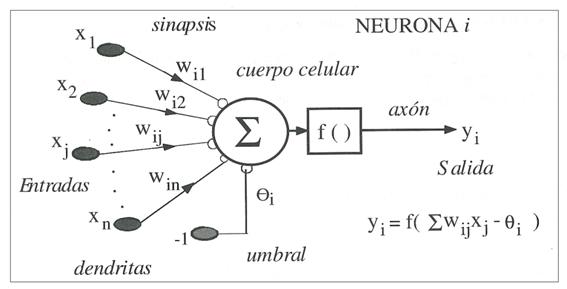
Las m **entradas** representan un vector x

Por cada neurona, existe un **vector w de pesos** sinápticos que indican la fuerza de conexión entre los valores de entrada y la neurona. En la práctica representan la ponderación de cada entrada sobre la neurona.

Una constante **theta**.

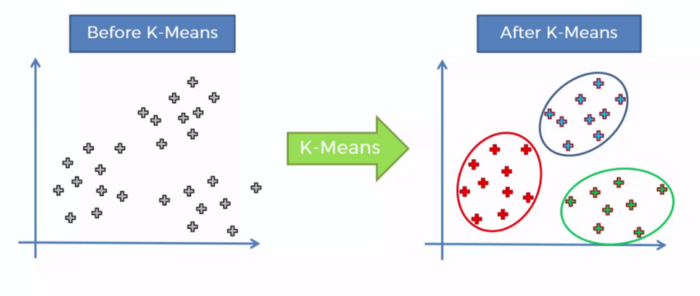
La **salida** y de la neurona se representa por la función de activación, que se define como





**Kmedias**

es un método de **agrupamiento**, que tiene como objetivo la partición de un conjunto de n observaciones en k grupos en el que cada observación pertenece al grupo cuyo valor medio es más cercano. Es un método utilizado en minería de datos.



genera una partición de un conjunto de n observaciones en k grupos. Cada grupo está representado por el **promedio** de los puntos que lo componen. El representante de cada grupo se denomina **centroide**. La cantidad de grupos a descubrir, k, es un parámetro que se debe fijar a priori. El método de clustering comienza con k centroides ubicados de forma **aleatoria**, y asigna cada observación al centroide más cercano. Después de asignarlos, los centroides se mueven a la ubicación promedio de todos los datos asignados a él, y se vuelven a reasignar los puntos de acuerdo a las nuevas posiciones de los centroides.

El objetivo es **agrupar** a las observaciones de forma tal que todas las que se encuentren en el mismo grupo sean lo más semejantes entre sí y que las pertenecientes a grupos distintos sean lo más desemejantes entre sí. Las **medidas de distancia**, como la euclídea, son utilizadas para medir la semejanza y desemejanza

**Perceptrón**

Un perceptrón es una **neurona artificial**, y, por tanto, una unidad de red neuronal. El perceptrón efectúa **cálculos** para detectar características o tendencias en los datos de entrada.

El perceptrón recibe múltiples **señales de entrada**. Si la suma de las señales supera un **umbral** determinado, se produce una señal o, por el contrario, no se emite ningún resultado.

En el marco del método de aprendizaje supervisado de Machine Learning, es lo que permite predecir la **categoría** de una muestra de datos. Por tanto, se trata de un elemento esencial.

Diagrama

Descripción generada automáticamente

En realidad el perceptrón es una función matemática. Los datos de entrada (x) se multiplican por los **coeficientes** de peso (w). El resultado es un valor. Ese valor puede ser positivo o negativo. La neurona artificial se activa si el valor es positivo. Solo se activa si el peso calculado de los datos de entrada supera un **umbral** determinado.

El resultado **predicho se compara** con el resultado conocido. En caso de diferencia, el error se retropropaga para permitir ajustar los pesos.

En resumidas cuentas, una red neuronal es un conjunto de **perceptrones interconectados**. Su funcionamiento se basa en operaciones de multiplicación entre dos componentes importantes: las entradas de datos (input) y el peso.

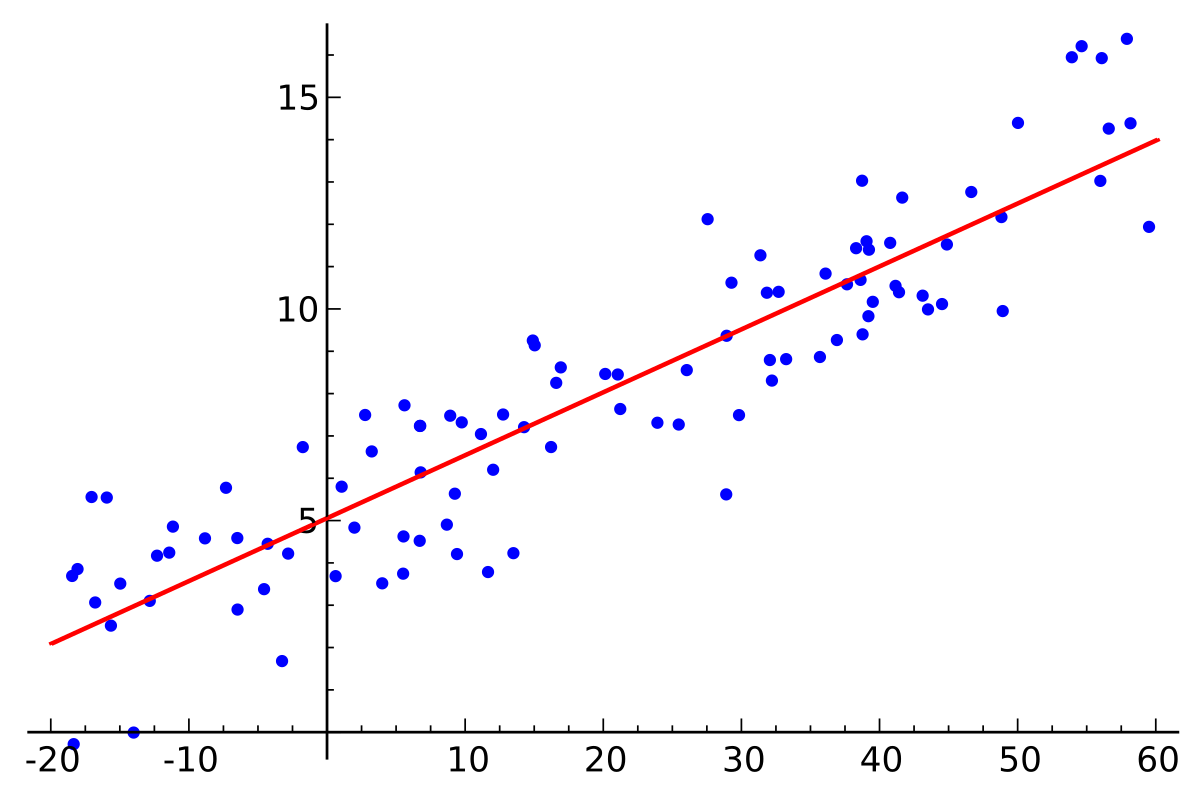
La suma de esa multiplicación se transmite a una **función de activación**, que determina un valor binario de 0 o 1. Lo que permite clasificar los datos.

**Regresión lineal**

trata de explicar la **relación** que existe entre una variable dependiente (variable respuesta) Y un conjunto de variables independientes (variables explicativas) X

Escala de tiempo

Descripción generada automáticamente



Tabla

Descripción generada automáticamente

**Cinco niños** de 2, 3, 5, 7 y 8 años de edad pesan, respectivamente, 14, 20, 32, 42 y 44 kilos



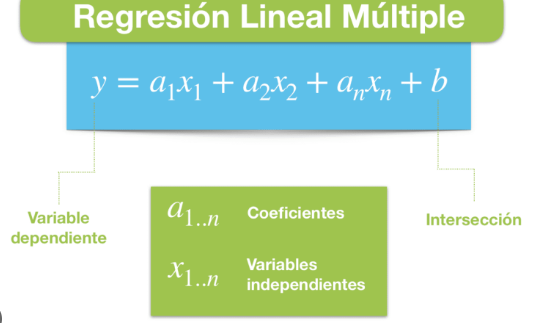
De 6 años

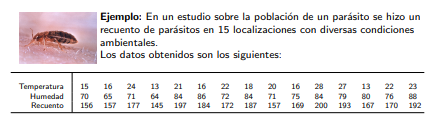
Logotipo

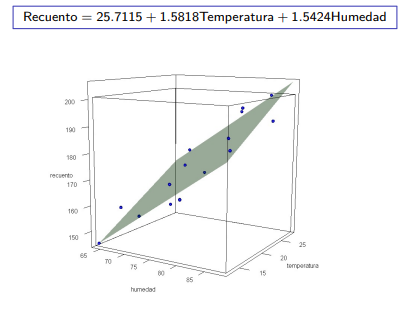
Descripción generada automáticamente

**Regresión lineal multiple**

La regresión lineal múltiple permite generar un modelo lineal en el que el valor de la variable dependiente o respuesta (Y) **se determina a partir** de un conjunto de variables independientes llamadas predictores (X1, X2, X3…).

****

****

****

**Sgd**

Cada neurona de la red (exceptuando la capa de entrada) es en realidad un sumatorio de todas sus entradas; que no son más que las salidas de las capas anteriores multiplicadas por unos pesos. A esta suma se le añade un término adicional llamado sesgo o bias. Y al resultado se le aplica una función no lineal conocida como función de activación. Pues bien; los **parámetros** de la red (pesos, bias) son precisamente los valores numéricos que trataremos de **ajustar** mediante entrenamiento usando un conjunto de muestras ya etiquetadas. lo primero que debemos hacer es inicializar sus parámetros. A partir de aquí procederemos iterativamente siguiendo un algoritmo de optimización, que tratará de **minimizar** la diferencia entre la salida real y la estimada por la red. el gradiente es un cálculo que nos permite saber cómo ajustar los parámetros de la red de tal forma que se **minimice su desviación** a la salida.

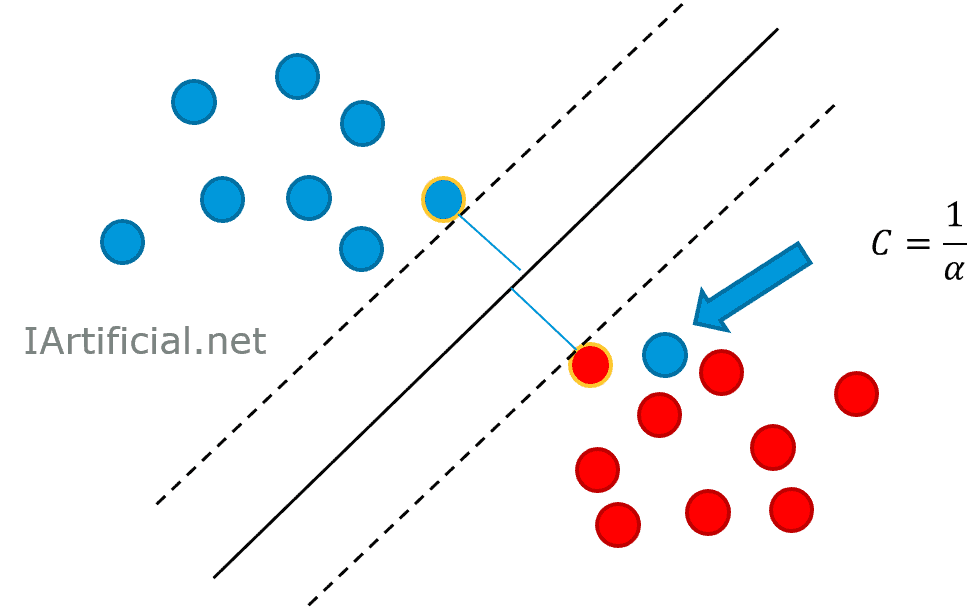
En el Descenso del gradiente estocástico se introduce **una única muestra aleatoria** en cada iteración. El gradiente se calculará para esa muestra concreta, lo que supone la introducción de la deseada aleatoriedad, dificultando así el estancamiento. El problema de esta versión es su lentitud, ya que necesita de muchas más iteraciones, y además no aprovecha los recursos disponibles.

El descenso del grandiente en un algoritmo iterativo que empieza en un punto aleatorio de una función y viaja a través de la pendiente hacia abajo para alcanzar el **punto mas bajo** de la función

El descenso de gradiente estocástico es un **método iterativo** para optimizar una función objetivo con propiedades de suavidad adecuadas

**Maquina de vectores soporte**

Dado un conjunto de ejemplos de entrenamiento (de muestras) podemos etiquetar las clases y entrenar una SVM para construir un modelo que prediga la clase de una nueva muestra. Intuitivamente, una SVM es un modelo que representa a los puntos de muestra en el espacio, **separando las clases** a 2 espacios lo más amplios posibles mediante un hiperplano de separación definido como el vector entre los 2 puntos, de las 2 clases, más cercanos al que se llama vector soporte. Cuando las nuevas muestras se ponen en correspondencia con dicho modelo, en **función de los espacios** a los que pertenezcan, pueden ser clasificadas a una o la otra clase.



La línea que mejor distingue las zona de los puntos azules de la zona de los puntos rojos es la línea que **maximiza el margen** entre ambos. Las máquinas de vectores de soporte son una técnica de machine learning que encuentra la **mejor separación** posible entre clases. Con dos dimensiones es fácil entender lo que está haciendo. Normalmente, los problemas de aprendizaje automático tienen muchísimas dimensiones. Así que en vez de encontrar la línea óptima, el SVM encuentra el hiperplano que maximiza el margen de separación entre clases.

**Discretizar, normalizar**

la discretización es el proceso de transferir **funciones continuas**, modelos, variables y ecuaciones a contrapartes discretas. Es el proceso de transformar una **variable numérica** en una categórica. Un ejemplo muy común es el de convertir una variable como “Edad” en categóricas tales como “20-30”, “60-79” etc

la normalización de índices significa ajustar los valores medidos en diferentes escalas respecto a una **escala común**, a menudo previo a un proceso de realizar promedios.

**variables continuas, descretas y categoricas**

Una variable **discreta** es una variable que no puede tomar algunos valores dentro de un **mínimo conjunto numerable**, quiere decir, no acepta cualquier valor, únicamente aquellos que pertenecen al conjunto. el número de hijos en una familia **(1; 2; 3; 4; …).**

Una variable **continua** puede tomar un valor fijo dentro de un **intervalo** determinado. Una variable continua toma valores a lo largo de un continuo, esto es, en todo un intervalo de valores. estatura de una persona **(1,72 m; 1,719 m; 1,718 6 m....)**

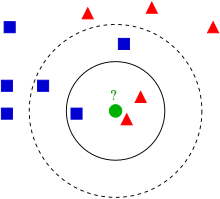
Una variable **categórica** es una variable que puede tomar uno de un número **limitado**, y por lo general fijo, de posibles valores. **1 a soltero, 2 a casado**, 3 a divorciado, 4 a en pareja y 5 a otros. número finito de categorías o grupos distintos.

**Kvecinos**

En la clasificación por k-NN, la salida es la **pertenencia a una clase**. Un objeto es clasificado como perteneciente a una clase si la **mayoría de sus k vecinos** pertenecen a esa clase: Si se quiere clasificar un objeto, con k= 5 vecinos, con 3 vecinos de la clase A y 2 de la clase B, por mayoría el objeto se clasifica como clase A. Para evitar empates, se prefiere que el número de vecinos k seleccionado sea un número impar. En la regresión por k-NN, la salida es un valor de una propiedad del objeto; ese valor se puede calcular usando el promedio de los valores de los k vecinos. La palabra “cercano” implica una **métrica de la distancia**. Típicamente se puede usar la distancia Euclidiana:

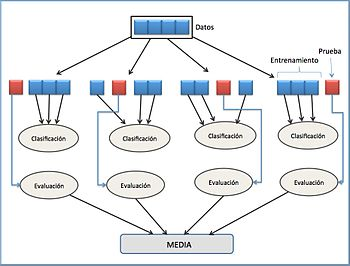
Gráfico

Descripción generada automáticamente con confianza baja



**Validación cruzada**

es una técnica utilizada para evaluar los resultados de un análisis estadístico y garantizar que son **independientes** de la partición entre datos de entrenamiento y prueba. Consiste en **repetir y calcular la media aritmética** obtenida de las medidas de evaluación sobre diferentes particiones



**Holdout**

Es el método mas simple para evaluar un clasificador. El conjunto de datos es separado en conjunto de **entrenamiento y de test**

**Email spam** o no. 20 items: 12 entrenamiento, 8 test. Creamos el modelo y con los datos de test evaluamos la precisión

Diagrama, Escala de tiempo

Descripción generada automáticamente

**Regresión logisitca**

es un tipo de análisis de regresión utilizado para predecir el resultado de una variable **categórica** (una variable que puede adoptar un número limitado de categorías) en función de las variables independientes o predictoras. Es útil para modelar la **probabilidad** de un evento ocurriendo en función de otros factores.

**enfermedad** es realmente grave o cuando no.

Gráfico, Gráfico de líneas

Descripción generada automáticamente

**Naive bayes**

un clasificador de Naive Bayes asume que la presencia o ausencia de una característica particular **no está relacionada con la presencia** o ausencia de cualquier otra característica, dada la clase variable. Por ejemplo, una fruta puede ser considerada como una manzana si es roja, redonda y de alrededor de 7 cm de diámetro. Un clasificador de Naive Bayes considera que cada una de estas características contribuye de **manera independiente** a la probabilidad de que esta fruta sea una manzana, independientemente de la presencia o ausencia de las otras características.

Interfaz de usuario gráfica, Texto

Descripción generada automáticamente

Alicia viene a **la oficina 3 días** a la semana.Bruno viene a la oficina 1 día a la semana.Esta sería nuestra información “anterior”.

Estamos en la oficina y vemos pasar delante de nosotros a alguien muy rápido, tan rápido que no sabemos si es Alicia o Bruno.

Dada la información que tenemos hasta ahora y asumiendo que solo trabajan 4 días a la semana, las probabilidades de que la persona vista sea Alicia o Bruno, son:

P(Alicia) = 3/4 = 0.75

P(Bruno) = 1/4 = 0.25

Cuando vimos a la persona pasar, vimos que él o ella llevaba una chaqueta roja. También sabemos lo siguiente:Alicia viste de rojo 2 veces a la semana.Bruno viste de rojo 3 veces a la semana.

Así que, para cada semana de trabajo, que tiene cinco días, podemos inferir lo siguiente:

La probabilidad de que Alicia vista de rojo es → P(Rojo|Alicia) = 2/5 = 0.4

La probabilidad de que Bruno vista de rojo → P(Rojo|Bruno) = 3/5 = 0.6

