Tipos de algoritmos en relación con el aprendizaje automático.

Caso práctico

@casfatesvano (CC BY-SA)

Max acaba de terminar unas prácticas como estudiante de Inteligencia Artificial en una empresa de logística. Y está a punto de terminar también los estudios correspondientes.

Quiere seguir aprendiendo y, en la medida de lo posible, ganar experiencia profesional en el sector. Así que ha solicitado acceder como becaria en una empresa de servicios hospitalarios, en el departamento de I+D. Justo estaban buscando un perfil como el suyo, con conocimientos de aplicaciones prácticas de Inteligencia Artificial.

En los primero días en esta nueva empresa ha descubierto que tienen en marcha diversos proyectos de Inteligencia Artificial. En todos ellos están aplicando *Machine Learning* (Aprendizaje Automático), pero en cada caso lo hacen con diferentes técnicas y "algoritmos". Resulta que la clasificación entre "supervisado", "no supervisado" y "por refuerzo" era solo "la punta del iceberg". Tiene mucho que preguntar y observar de sus compañeros para entender por qué se usa cada una de estas técnicas en cada caso.

A lo largo de esta unidad vas a profundizar en técnicas concretas que se utilizan en el Aprendizaje Automático Supervisado y No Supervisado, y también irás comprendiendo cuáles de ellas son las óptimas según el tipo de aplicación práctica se quiera desarrollar. En concreto vas a conocer (hay algunos más, pero no tan utilizados como estos):

- Regresión Lineal.
- Regresión Logística.
- Árbol de Decisión.
- Máquinas de Vector Soporte.
- Clustering.

Materiales formativos de FP Online propiedad del Ministerio de Educación y Formación Profesional.

Aviso Legal

1.- Regresión Lineal.

Caso práctico

El primer proyecto en el que va a colaborar Max en su nueva beca es una aplicación que calcula el precio de venta de instalaciones sanitarias (consultas privadas, consultorios médicos, etc). Para ello se basan en una base de datos con los casos de este tipo de instalaciones que se han vendido en los últimos 19 años en todo el territorio nacional.

Max sabe que si se tiene claro cuál es el dato objetivo que se quiere encontrar hay que aplicar Aprendizaje Automático Supervisado... ¿Pero con qué técnica de todas?

Sus compañeros de departamento llevan trabajando sobre la base de datos una temporada y han decidido que van a utilizar campos como el número de habitaciones de cada instalación, si está a pie de calle o en altura, la extensión en metros cuadrados, el número de ventanas al exterior y el número de ventanas a patios interiores, el año de construcción, el año de la última reforma (si ha tenido alguna reforma), la ciudad y barrio en el que se encuentra.

Muchos de estos datos son valores reales con distribución continua (por ejemplo, los metros cuadrados de extensión). Así que han decidido que van a aplicar un algoritmo de Regresión Lineal, pues sospechan que la relación de la mayoría de estos campos con el precio de venta tienen una relación directa (a más habitaciones, a más extensión, a más ventanas, a mayor año de construcción... mayor precio de venta).

Introducción

Ya hemos visto que existen, principalmente, tres tipos de Aprendizaje Automático: supervisado, no supervisado, y por refuerzo. Pero ahora vamos a fijarnos en los algoritmos y técnicas que se utilizan para programar los modelos que serán entrenados con los datos y

que después podrán utilizarse para predecir, encontrar patrones o ejecutar acciones similares a las de la inteligencia humana.

Dentro de los principales algoritmos más conocidos (según el grupo correspondiente a las matemáticas en las que están basados) vamos a empezar por los algoritmos de Regresión Lineal.

En general, cualquier regresión funciona generando un modelo que relaciona las variables del problema, de forma iterativa, a través de la media del error con la función que gobierna la predicción.

Esto quiere decir que si, por ejemplo, queremos predecir a través de una radiografía si un paciente tiene una enfermedad, la regresión lo que hace en el proceso de "entrenamiento" con miles de radiografías "etiquetadas" (sabiendo cuáles se corresponden con pacientes enfermos y cuáles no) es buscar relaciones entre la información que logra sacar de cada radiografía con el objetivo del estudio (saber si hay enfermedad o no). Y esto lo hace construyendo una "función matemática" a base de "probar" sus predicciones con la realidad de los casos etiquetados.

Podríamos decir que la regresión en general es más un proceso de ajuste que un algoritmo como tal.

Regresión Lineal

La Regresión Lineal **se usa para estimar valores reales de variables con distribución continua**. Por ejemplo el precio de viviendas, el número de llamadas que hay que hacer para conseguir un cliente nuevo, las ventas totales que habrá el trimestre que viene...

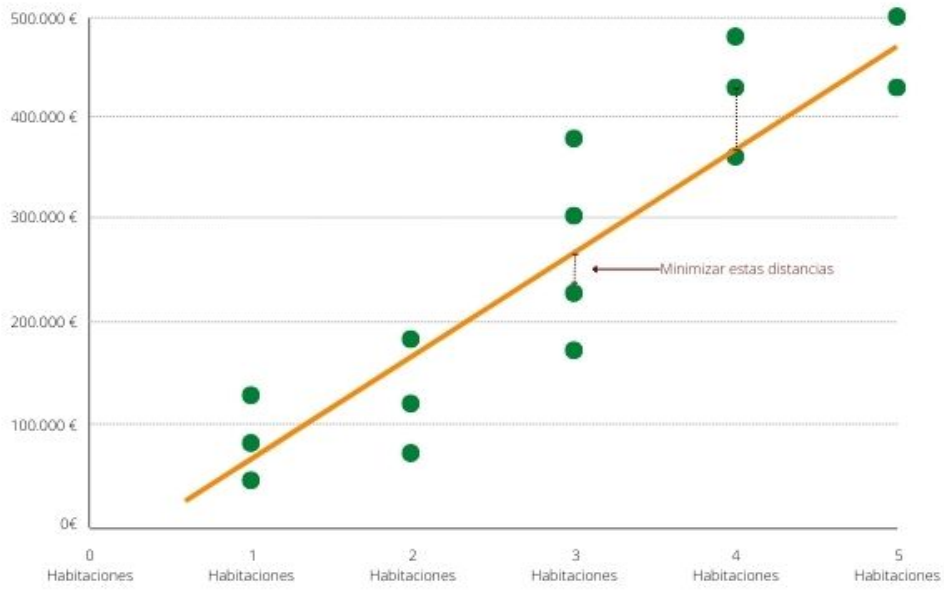
Como hemos dicho hay unas **variables de entrada**, que en un problema concreto de Aprendizaje Automático son las características de cada instancia que conocemos. Por ejemplo, el número de habitaciones que tiene una vivienda, el número de planta que ocupa en el edificio, el número de ventanas al exterior, el número de ventanas a patios interiores, cuántos ascensores tiene el edificio, cuántos metros cuadrados tiene la vivienda, etc. Y nosotros, al ser un problema de Aprendizaje Automático Supervisado, definimos el campo objetivo, por ejemplo el precio por el que se vende dicha vivienda. A este campo objetivo, desde el punto de vista matemático de las regresiones lo llamamos **valor de salida**.

Pues en el caso de la regresión lineal, la relación entre las variables de entrada y el valor de salida (o la predicción) se produce a través de la **generación de una línea que se ajuste lo mejor posible a la distribución de resultados**.

En los casos más sencillos esta relación se representa con la ecuación de una recta: $y = a \cdot x + b$, donde la x representa la variable independiente (datos de entrada), mientras que la y representa la variable dependiente (valor de salida).

Y los coeficientes a y b se obtienen al **encontrar la distancia mínima entre cada caso particular (instancia) con la recta que se está intentando encontrar**. Bueno, matemáticamente se dice que dichos coeficientes se obtienen al hallar el valor mínimo de la suma de los cuadrados de las distancias de los puntos a la recta.

Minimizar distancias - Regresión Lineal



Fran Bartolomé - Elaboración propia (CC BY-SA)

Para entendernos, lo que hace el algoritmo de regresión lineal es ir inclinando más o menos la línea para que al final pase lo más cerca posible de cada punto. Y cuando logra dar con la inclinación que mejor se ajusta a este requerimiento, mira a ver qué fórmula le corresponde. Y eso es lo que nos da como resultado el entrenamiento: una fórmula. A partir de ahí, cuando metamos en dicha fórmula los datos de una nueva instancia, nos dará como resultado la predicción. En el ejemplo que estamos viendo, al decirle el número de habitaciones que tiene la vivienda, nos dará una predicción del precio por el que se venderá dicha vivienda.

Cuando tenemos una relación simple entre variables (regresión simple o unidimensional) es fácil representar gráficamente esta relación, pues es una línea recta. Pero lo normal es que nos encontremos problemas en los que haya numerosas variables. Entonces es más complicado "dibujar" esa línea. Pero en cualquier caso el algoritmo, aunque no podamos "representarlo" sigue funcionando igual de bien. En el ejemplo de la vivienda, son muchas más variables las que influyen en el precio de venta que simplemente el número de habitaciones (la extensión, el barrio, si es un sótano o está en pisos altos, el número de cuartos de baño, el año de construcción...).

Aunque en realidad esta técnica pertenece todavía al campo de la estadística, y solo sirve para casos muy concretos y sencillos en los que la relación de las variables es proporcional y además, con una proporcionalidad lineal. Pero es un buen caso de partida para entender cómo funcionan los algoritmos de aprendizaje automático.

2.- Regresión Logística.

Caso práctico

@casfatesvano (CC BY-SA)

Max se enfrenta a su segundo proyecto en la beca que ha conseguido en una empresa de gestión hospitalaria. Con el historial médico de pacientes como base de datos, sus jefes quieren entrenar un modelo de Inteligencia Artificial que pronostique qué pacientes tienen más posibilidades de padecer un infarto o ataque cardiaco.

Los compañeros de Max le explican que el primer prototipo que van a entrenar va a utilizar el algoritmo de regresión logística, que es el más adecuado para cuando quieres clasificar casos en valores discretos (sí o no; blanco o negro; verdadero o falso). De este modo el campo objetivo en el entrenamiento será si el paciente ha sufrido infarto o no, e irán estudiando qué otros datos del historial médico tienen más influencia en los casos positivos.

No descartan, más adelante, hacer otros desarrollos de Inteligencia Artificial con diferentes algoritmos, para comparar. Pero de momento, como primer paso, será este algoritmo de regresión logística el que van a probar.

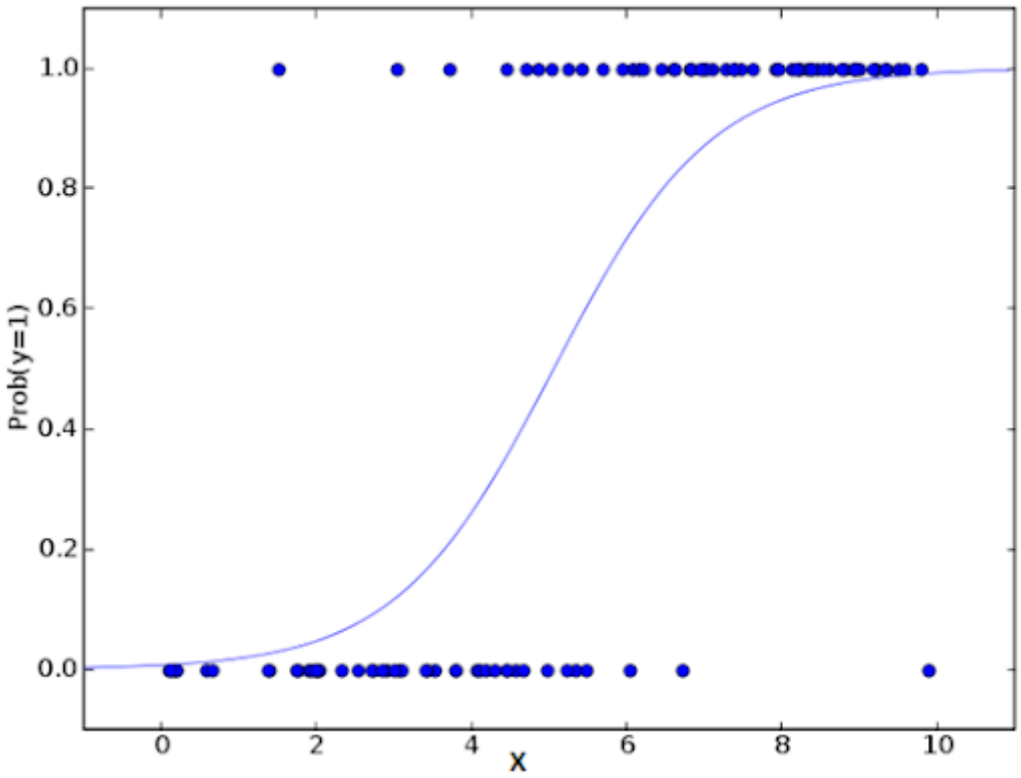
El algoritmo de regresión logística es más bien un algoritmo de clasificación que de regresión, pues va a estimar valores discretos (SI/NO, 1 ó 0, verdadero o falso) en función de las variables de entrada (y no una distribución continua de valores como en el caso de la Regresión Lineal).

Es decir, que este tipo de algoritmo nos va a permitir predecir la probabilidad de que se produzca un evento.

La regresión logística resulta útil para los casos en los que se desea predecir la presencia o ausencia de una característica o resultado según los valores de un conjunto de predictores

Matemáticamente la función logística se representa por una curva en forma de S (conocida como *función sigmoide*), que va, en el eje Y, desde el valor cero (0% probabilidad de que se produzca el evento) al valor 1 (100% probabilidad de que se produzca el evento).

Función Sigmoide, propia de las regresiones logísticas



Fran Bartolomé (Dominio público)

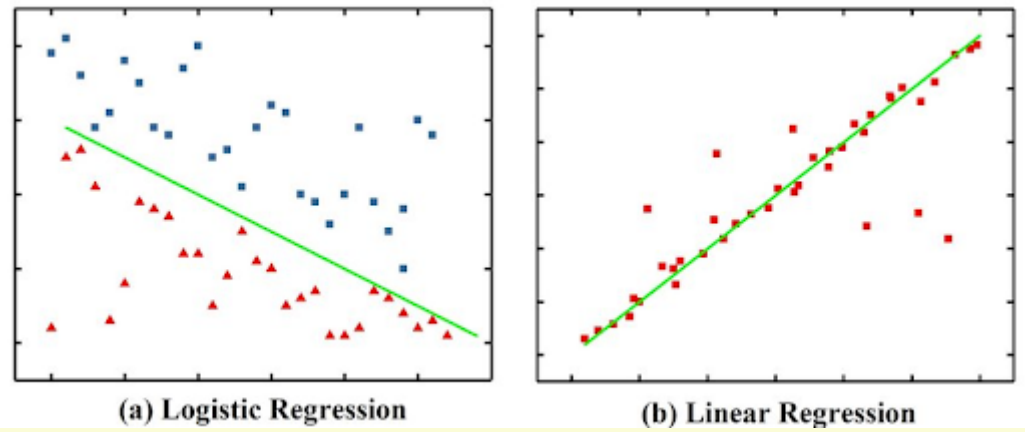
En los casos sencillos, en los que podemos hacer una representación gráfica realista, la curva nos separa a un lado y a otro las instancias según cumplan o no la condición objetivo.

La regresión logística es una técnica muy empleada por los **científicos de datos** debido a su **eficacia y simplicidad**. No es necesario disponer de grandes recursos computacionales, tanto en el entrenamiento como en la ejecución. Además, los resultados son fácilmente interpretables. Siendo esta una de sus principales ventajas respecto a otras técnicas. El resultado del entrenamiento también nos permite conocer la influencia de cada una de las características o datos de entrada en el resultado final. Por lo tanto, se puede afirmar que el modelo ha tomado una decisión u otra en base a la existencia de una u otra característica en el registro. Lo que en muchas aplicaciones es altamente deseado además del modelo en sí.

En este tipo de entrenamientos es importante hacer un buen trabajo de cribado de datos antes de hacer el entrenamiento. Es especialmente importante eliminar datos que no tengan una relación directa con el campo objetivo. También es importante eliminar las características que muestran una gran multicolinealidad entre sí (por ejemplo en un diagnóstico de diabetes, el azúcar en sangre y la glucosa en sangre son a efectos prácticos iguales, podemos prescindir de uno de ellos al hacer el entrenamiento). En conclusión: **la selección de las características previa al entrenamiento del modelo es clave**.

Hay muchos casos y ejemplos en los que se utiliza la regresión logística en Aprendizaje Automático Supervisado. Por ejemplo, el clasificador de Spam del correo electrónico (observando características de un mail ¿es spam o no lo es?), o para predecir enfermedades (con tal historial clínico, ¿tendrá tal enfermedad o no la tendrá?).

Para saber más



Fran Bartolomé (Dominio público)

Como puedes ver en el gráfico, la regresión logística lo que hace es "buscar" la línea que separa en dos partes las instancias (las que sí cumplen la condición objetivo y las que no). En el caso de la regresión lineal lo que se "busca" es dibujar la línea que pasa lo más cerca posible de todas las instancias.

3.- Árbol de Decisión.

Caso práctico

Max se va a enfrentar a un nuevo proyecto dentro de la empresa en el transcurso de su beca. En esta ocasión tienen que desarrollar un modelo de Inteligencia Artificial que pronostique qué pacientes tienen diabetes.

Para ello cuentan con una base de datos de casos desde hace veinte años. En ellos se recogen datos de análisis de sangre de cada paciente en el momento de diagnosticarles dicha enfermedad, así como de los análisis de sangre de pacientes para los que el diagnóstico fue negativo (o sea, que no tenían diabetes).

@Casfatesvano (CC BY-SA)

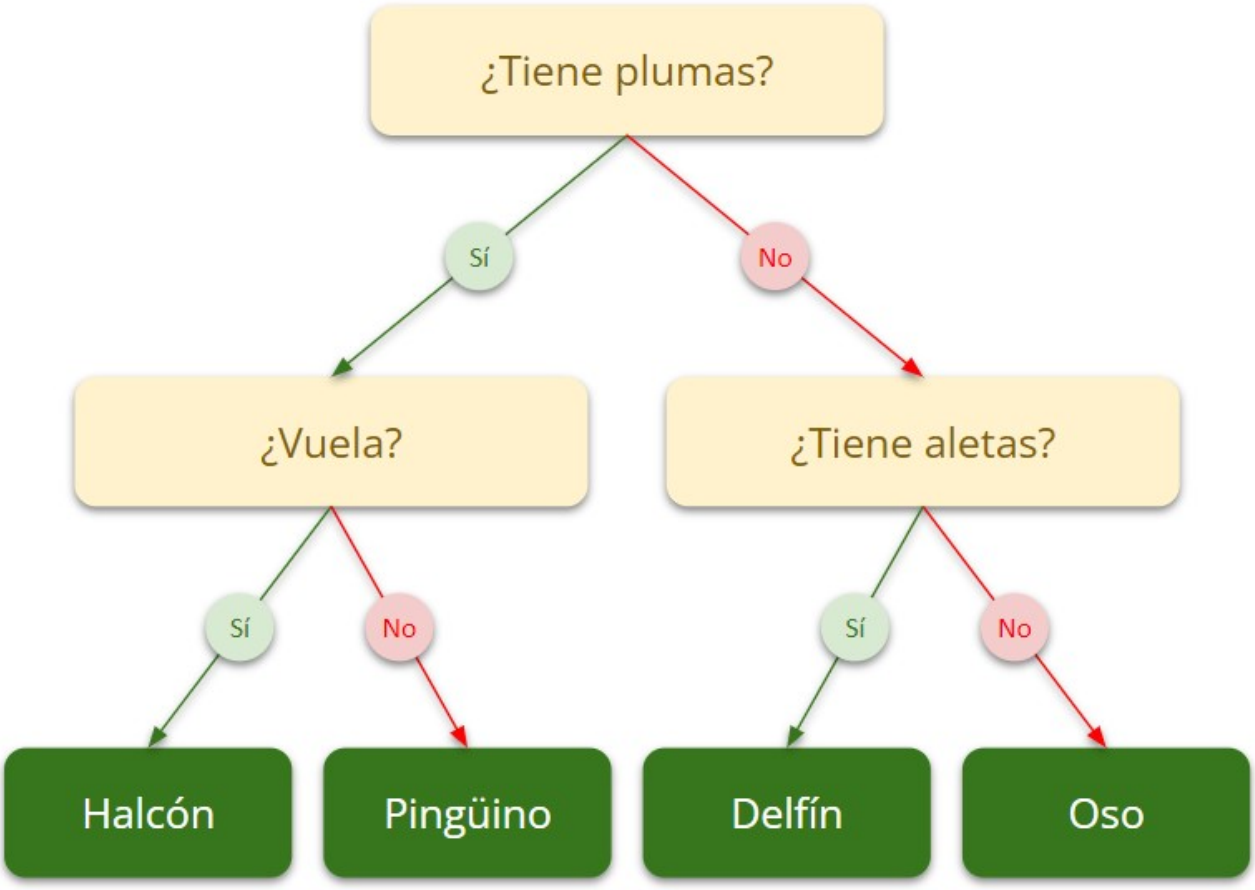
Como siempre, en primer lugar tienen que revisar los datos, para decidir qué parámetros van a tener en cuenta en el entrenamiento, cuáles no son significativos (por experiencia o bien porque no tengan suficientes en todo el histórico, etcétera).

Para este tipo de casos, en los que intervienen un buen número de parámetros tanto numéricos como categóricos, han decidido utilizar un algoritmo del tipo Árbol de Decisión, que es el más apropiado.

Los algoritmos de **árbol de decisión** construyen un modelo de decisiones basadas en los atributos que presentan los datos que entran en el modelo.

Se crean bifurcaciones hasta que se llega a una decisión concluyente en base a una predicción buscada. Estos algoritmos trabajan muy bien tanto con datos del tipo de regresión como de clasificación.

Ejemplo sencillo de Árbol de Decisión



Son los grandes favoritos en aprendizaje automático, porque son rápidos y precisos. Así que cuando no se tiene claro qué tipo de algoritmo usar para algún caso concreto, lo mejor es empezar aplicando este tipo de algoritmo.

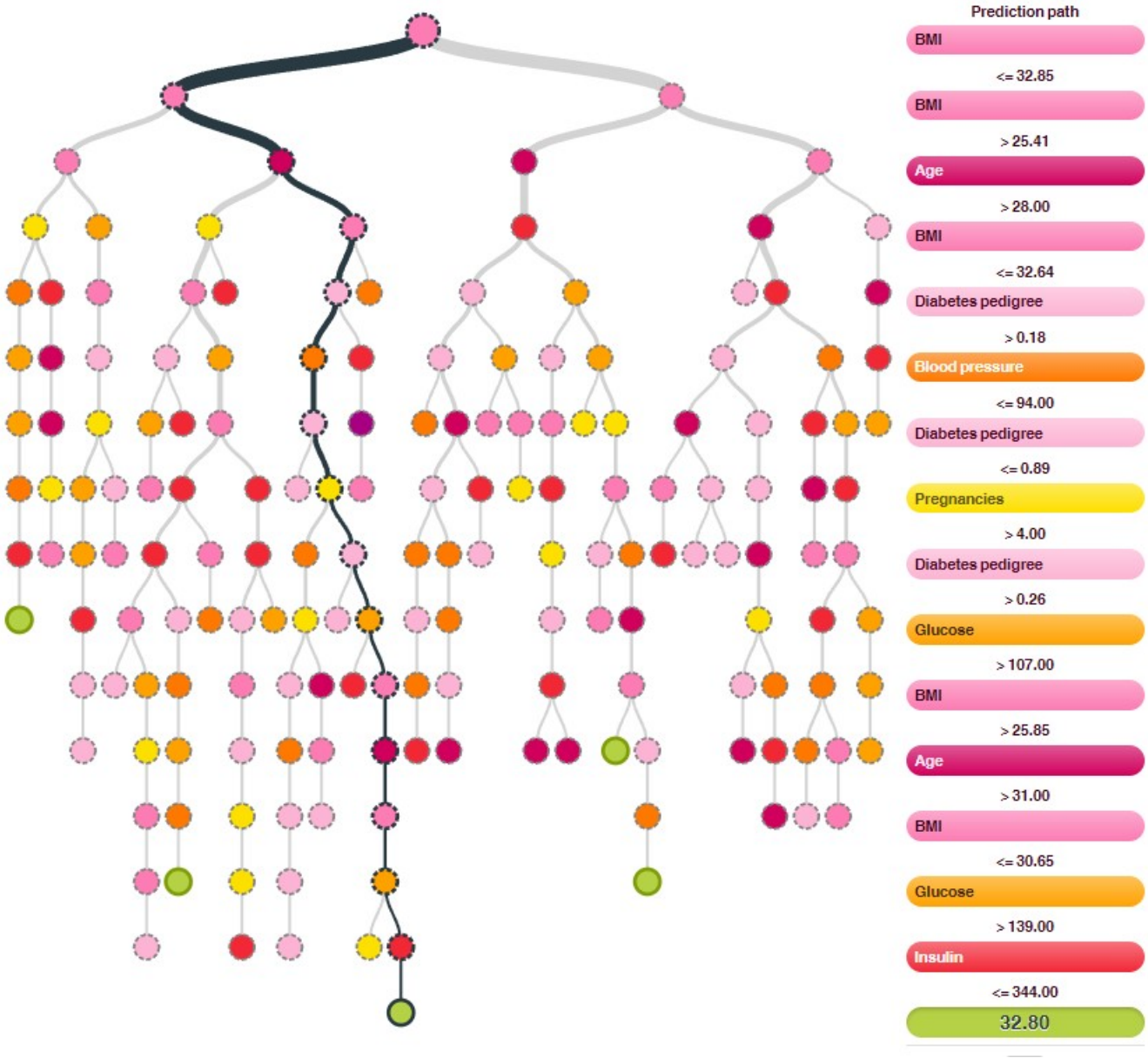
Los más utilizados son:

- Classification and Regression Tree (CART).
- Iterative Dichotomiser 3 (ID3).
- C4.5 and C5.0 (different versions of a powerful approach).
- Chi-squared Automatic Interaction Detection (CHAID).
- Decision Stump.
- M5.
- Conditional Decision Trees.

El más popular es el CART, que se conoce directamente como algoritmo de árbol de decisión, ya que los otros son mucho más específicos. Se usa bastante en problemas de clasificación, y se basa en distinguir grupos lo más relevantes posible en función de los valores que pueden tomar diferentes atributos.

Esta técnica consiste en dividir una muestra o población en dos grupos homogéneos (en cada bifurcación) basándonos en la variable de entrada más significativa o diferenciadora en ese momento.

Ejemplo de Árbol de Decisión



Fran Bartolomé - BigML (Dominio público)

En este ejemplo que muestra la ilustración, obtenida de un caso concreto en la herramienta BigML que estudia la probabilidad de tener diabetes según una serie de parámetros como el índice BMI, la edad, la glucosa en sangre, la insulina, entre otros, vemos cómo se va dividiendo de dos en dos en función de estos parámetros, hasta que llega a resultados abajo del todo. En este caso, la herramienta BigML nos muestra en color verde aquellos casos que tienen un índice de fiabilidad suficientemente significativo como para considerarlos válidos.

Autoevaluación

Completa la frase con las palabras que faltan.

La Regresión Lineal se usa para estimar valores reales de variables con distribución _____

Enviar

Autoevaluación

La regresión [redacted] resulta útil para los casos en los que se desea predecir la presencia o ausencia de una característica o resultado según los valores de un conjunto de predictores

Enviar

Autoevaluación

Los algoritmos de [redacted] de [redacted] construyen un modelo de decisiones basadas en los atributos que presentan los datos que entran en el modelo

Enviar

4.- Máquinas de Vector Soporte (SVM).

Caso práctico

@Casfatesvano (CC BY-SA)

Max se encuentra ya en su cuarto mes de prácticas en una empresa de servicios hospitalarios. En este nuevo mes le han dicho que va a colaborar en el desarrollo de una aplicación móvil capaz de reconocer la escritura de los médicos cuando hacen recetas. A veces los médicos van tan rápido al final de las consultas que escriben muy deprisa y con poca precisión, por lo que cuando los pacientes llegan a la farmacia para comprar el medicamento a veces tienen dificultades para saber exactamente qué tienen que pedir. Por no hablar de las indicaciones sobre la frecuencia con la que tomar cada medicina.

Así que lo que se proponen con este proyecto es crear un lector de recetas escritas a mano y, a través de Inteligencia Artificial, "traducir" lo que ha sido escrito.

En una segunda fase, dependiendo de lo bien preparada que esté la muestra de recetas con las que se entrene el modelo, se están planteando lograr identificar también qué doctor o doctora ha escrito dicha receta.

Max ya sabe que lo que le va a tocar va a ser trabajar con la base de datos para organizar convenientemente los datos. Van a trabajar en colaboración con veinte farmacias que han ido archivando copias de recetas desde hace años.

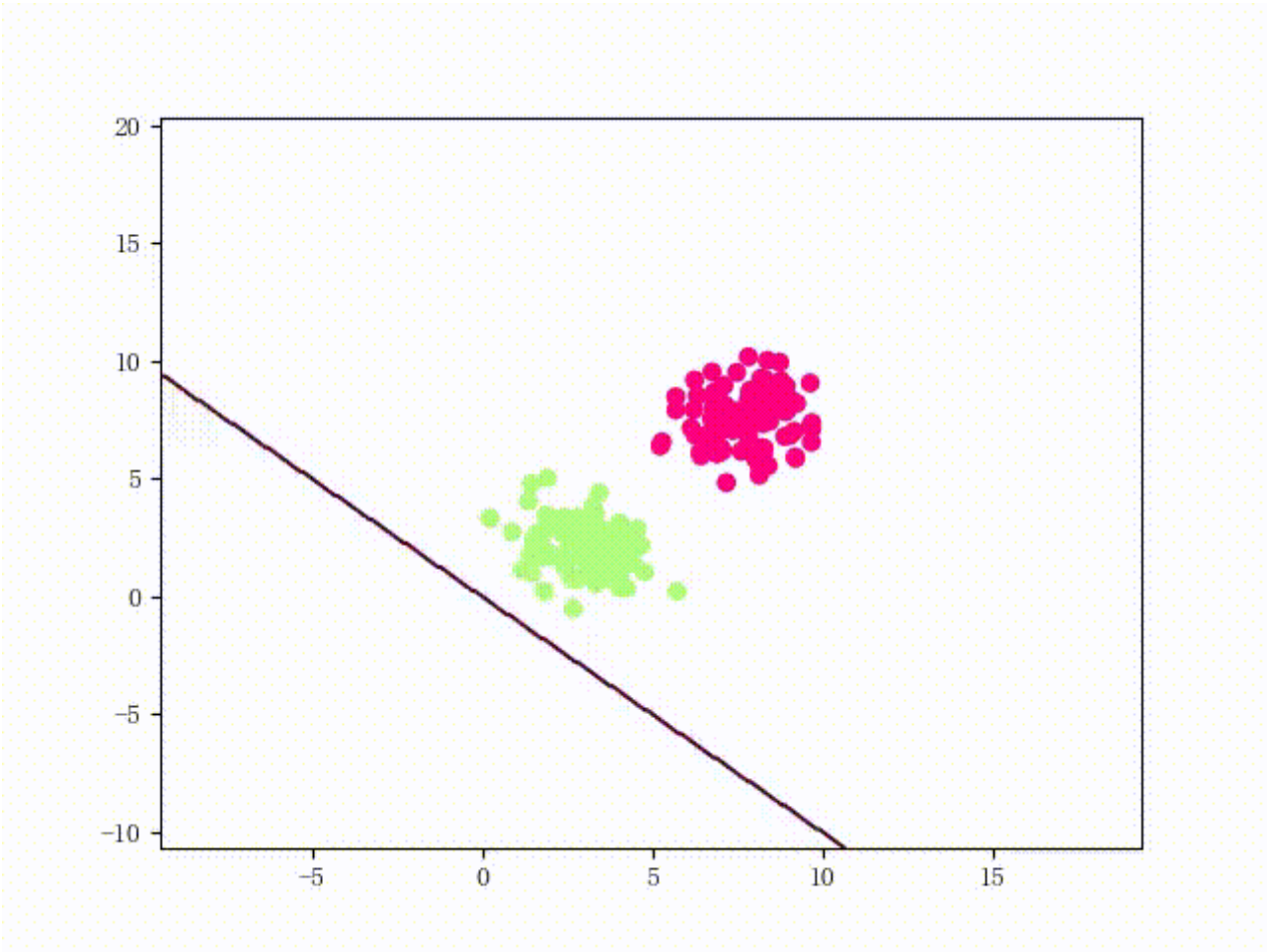
En este caso van a utilizar el método de Máquina de Vector Soporte, que se ha demostrado muy útil para casos de reconocimiento de escritura.

Las **Máquinas de Vector Soporte** (en inglés *Support Vector Machines* - SVM), es otro método de Aprendizaje Automático Supervisado que se aplica a **problemas de clasificación**. Podemos decir que este método "dibuja cada instancia" o caso como un punto en un espacio n-dimensional (donde n es el número de atributos que tomaremos como coordenadas para cada punto).

Por tanto 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.

Por ejemplo, si tuviésemos casos con solo dos atributos, el algoritmo de tipo SVM buscaría definir la función que separara los dos atributos de esta manera:



La técnica consiste en calcular la recta que separa ambos grupos de forma que esté equidistante de los dos puntos más cercanos, tal y como se puede ver en la imagen anterior. A la distancia total se le llama margen y los puntos que definen la distancia del margen se denominan vectores soporte.

La línea oscura será el clasificador. Al evaluar datos nuevos, dependiendo de en qué lado de la línea estén, serán clasificados como el resto de los puntos de esa región.

Claro, cuando estudiamos casos en los que participan más de dos atributos, la representación gráfica se complica. Pero nuestro objetivo no es saber dibujar... sino comprender en qué consiste este método.

Dentro de las técnicas de Machine Learning (Aprendizaje Automático) el uso de las máquinas de vector soporte como clasificador se ha visto incrementado en los últimos años debido a que sirven para resolver problemas de clasificación y regresión y que su rendimiento en los diferentes campos en los que se utilizan suele ser bastante alto, resultando ser uno de las técnicas más precisas.

Algunos campos de aplicación exitosos de este tipo de algoritmos han sido:

- Reconocimiento óptico de caracteres.

- Detección de caras para que las cámaras digitales enfoquen correctamente.
- Filtros de spam para correo electrónico.
- Reconocimiento de imágenes a bordo de satélites (saber qué partes de una imagen tienen nubes, tierra, agua, hielo, etc.).

5.- Clustering.

Caso práctico

@Casfatesvano (CC BY-SA)

¡Último mes de prácticas de Max en la beca con la empresa de servicios hospitalarios!

En esta ocasión el equipo de Inteligencia Artificial con el que va a colaborar Max está iniciando una aplicación que sea capaz de detectar melanomas (tipo de cáncer de piel) y diferenciarlos de lunares normales y corrientes. El proyecto se va a basar en reconocimiento de imágenes, y va a utilizar redes neuronales profundas, con las que Max aún no ha trabajado nunca.

Pero antes de iniciar el entrenamiento de esta Inteligencia Artificial, necesitan procesar y organizar bien la cantidad de datos. Según le han dicho a Max sus compañeros, es fundamental hacer una correcta clasificación previa de los tipos de melanomas, en función de los tipos de piel, tipos de pacientes, sintomatologías, enfermedades previas, y un largo etcétera.

De manera que van a empezar por hacer unos cuantos ejercicios de clasificación con la búsqueda de clústeres o agrupamientos (Aprendizaje Automático No Supervisado) para ver de qué manera están relacionados unos parámetros con otros. ¡A ver qué encuentran! En un primer momento iban a utilizar el algoritmo K-Means, pero como en realidad no tienen ninguna intuición de cuántos clústeres son los óptimos para este caso, han decidido recurrir al algoritmo G-Means.

Los algoritmos de tipo clustering, también llamados "de agrupación", son típicos de Aprendizaje Automático No Supervisado. Es decir, cuando en el entrenamiento de la Inteligencia Artificial no determinamos ningún campo objetivo. Se utilizan para **agrupar datos existentes de los que**, a priori, **no intuimos sus características en común**.

Funcionan creando unos puntos centrales o “centroides” y jerarquías para diferenciar los grupos y descubrir características comunes por cercanía.

Las agrupaciones o clusters que generan este tipo de algoritmos son útiles para explorar los datos en una primera aproximación, identificar anomalías en ellos y, finalmente, para realizar predicciones. Los modelos de agrupación en clústeres también pueden ayudar a identificar relaciones en un conjunto de datos que podrían no deducirse lógicamente mediante una simple observación o examen. Por estos motivos, la agrupación en clústeres se suele usar

en las primeras fases de las tareas de aprendizaje automático, a fin de explorar los datos y **detectar correlaciones inesperadas**.

Se utilizan muy frecuentemente en herramientas de negocios que te recomiendan productos o servicios que "intuyen" te pueden interesar, pues tienes características y comportamientos similares a los de clientes que ya han comprado o consumido dicho producto o servicio. Por ejemplo las recomendaciones de series o películas que te hacen en plataformas de contenidos audivisuales, o los productos que te recomiendan en portales de venta online.

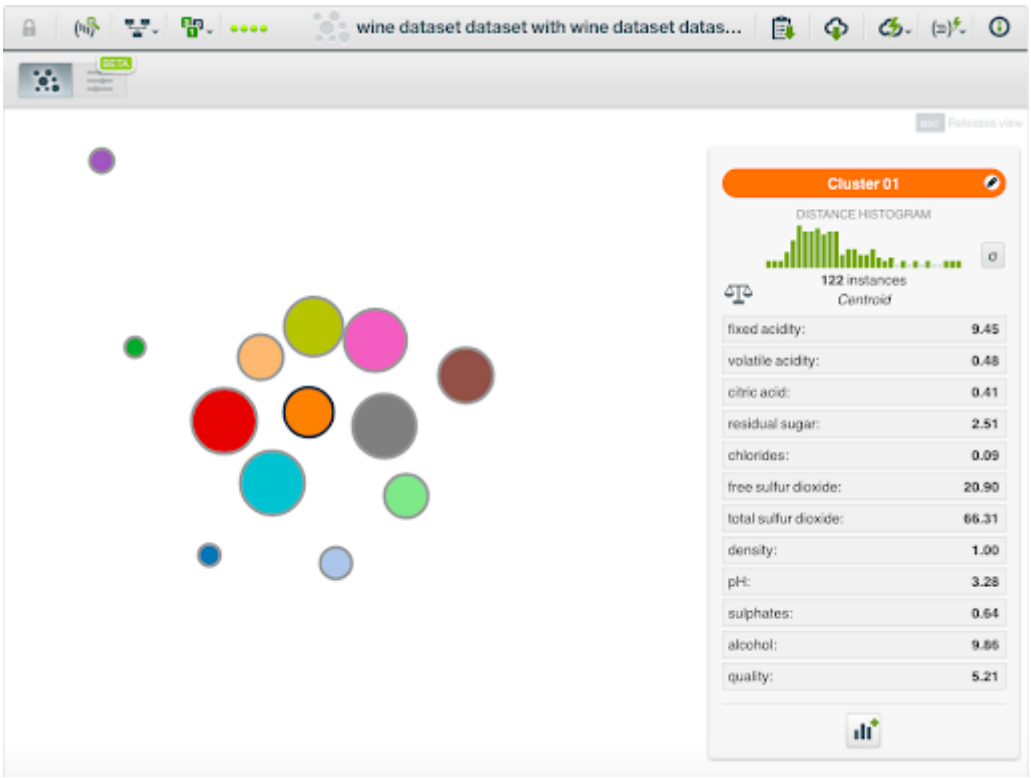
5.1.- K-Means.

Como todos algoritmos de aprendizaje no supervisado, el algoritmo k-Means se utiliza cuando tenemos una buena cantidad de datos sin etiquetar con el objetivo de encontrar “k” grupos entre los datos con características similares. El valor de "k" lo decidimos nosotros.

El algoritmo trabaja iterativamente para asignar a cada punto uno de los “k” grupos en función de sus características. Serán agrupados en función de similitudes en sus atributos.

Este método da como resultado unos puntos calculados que hacen las veces de centros o “centroides” de estos grupos, y etiquetas que se asignarán a cada uno de los k grupos formados.

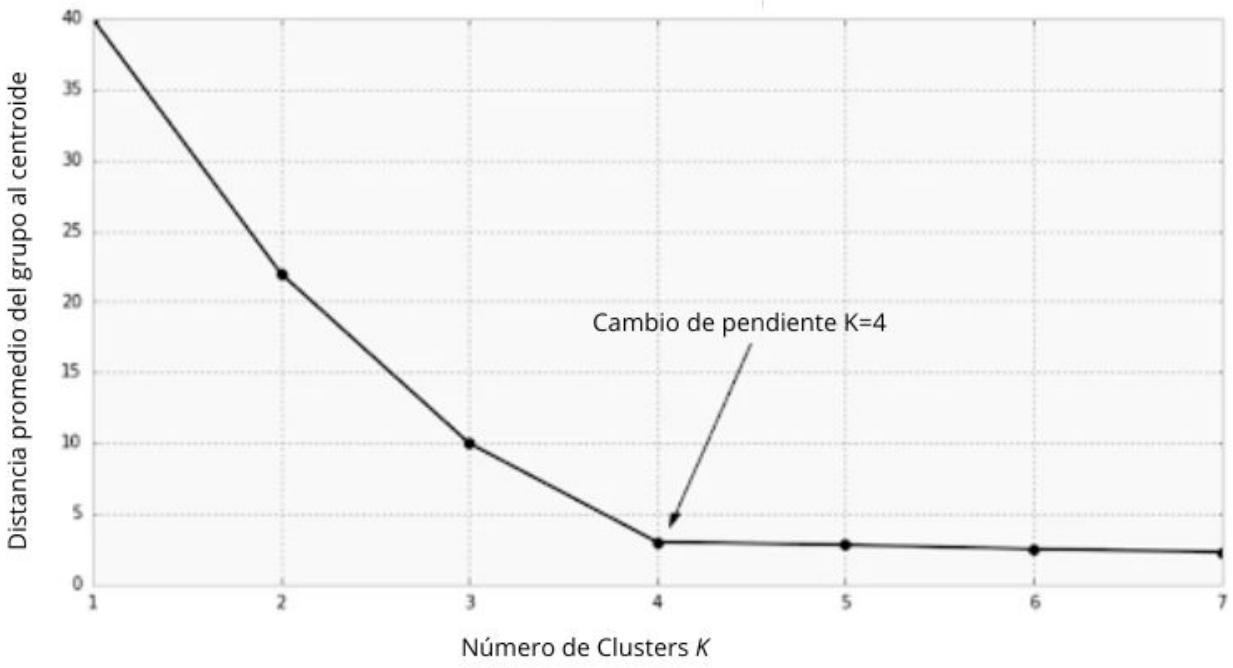
Los grupos se van definiendo de manera “orgánica”. Se va ajustando su posición en cada iteración del proceso, hasta que converge el algoritmo.



BigML - Fran Bartolomé (Dominio público)

Una cosa a tener en cuenta en este método es que, a priori, no podemos saber cuál es el valor más adecuado de k, es decir, cuántos grupos deberíamos tener para representar de forma útil los diferentes grupos de interés de nuestro problema. Como ya hemos mencionado, cada *cluster* tiene su propio centroide, que representa una media de los valores de los puntos que contiene el *cluster*.

A medida que aumentamos el número de clusters, k, el valor de la distancia media entre el centroide y los puntos, decrece. Pero hay un valor de k que marca una diferencia significativa en la pendiente de la curva que representa esta distancia media. Ese valor de k es el valor óptimo de grupos que deberíamos utilizar en el modelo.



Fran Bartolomé - Elaboración propia (CC BY-SA)

Algunos de sus usos más comunes son:

- Detección de datos anormales.
- Agrupación en clústeres de documentos de texto.
- Análisis de conjuntos de datos antes de utilizar otros métodos de clasificación o regresión.

Autoevaluación

Indica si las siguientes afirmaciones son verdaderas o falsas.

El uso de las **máquinas de vector soporte** como clasificador se ha visto incrementado en los últimos años debido a que sirven para resolver **problemas de clasificación y regresión** y que su rendimiento en los diferentes campos en los que se utilizan suele ser bastante alto.

Sugerencia

○ Verdadero ○ Falso

Verdadero

Efectivamente, las máquinas de vector soporte están resultando ser una de las técnicas más precisas en problemas de clasificación y regresión.

Los algoritmos de Clustering son los más utilizados para hacer predicciones en el caso de valores numéricos reales.

Sugerencia

○ Verdadero ○ Falso

Falso

Los algoritmos de Clustering los usamos en realidad en aprendizaje no supervisado, para encontrar grupos de características comunes entre las instancias.

Para hacer predicciones con instancias cuyo datos sean valores reales usamos la regresión lineal, en casos de aprendizaje supervisado.

Uno de los casos de uso más comunes de los algoritmos K-Means es en la detección de anomalías en un conjunto de datos.

Sugerencia

○ Verdadero ○ Falso

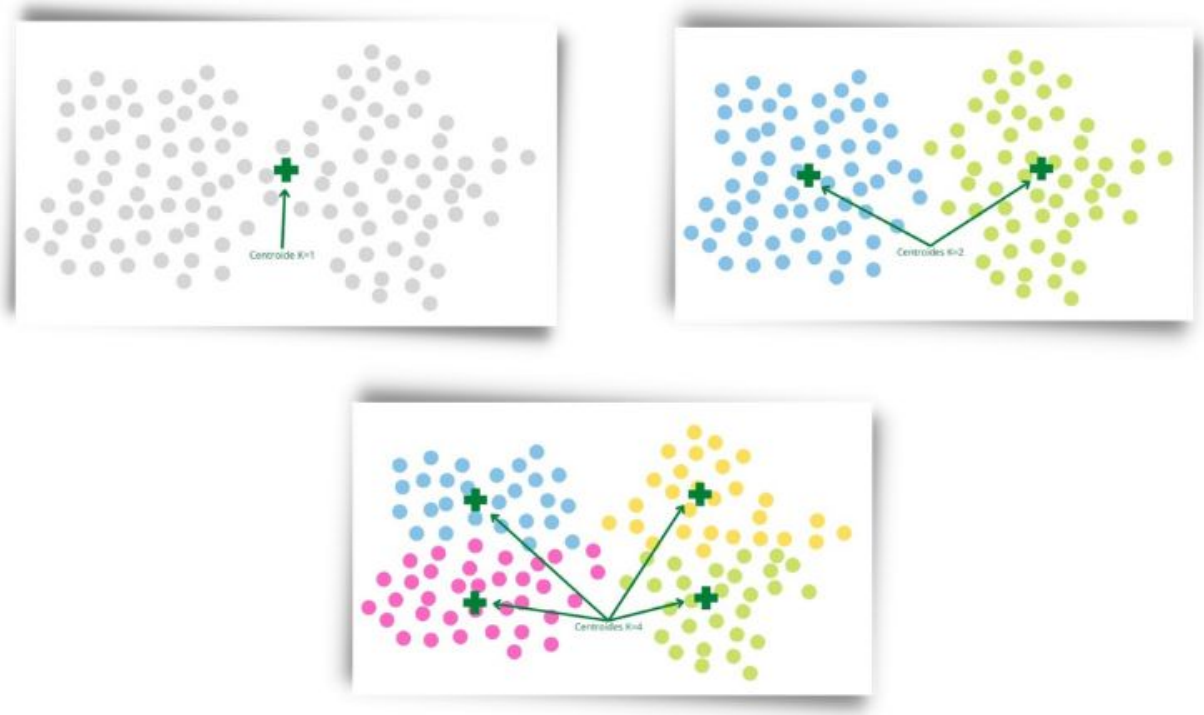
Verdadero

Así es, los algoritmos de tipo K-Means se están empleando habitualmente cuando se desea entrenar un modelo que detecte casos anómalos.

5.2.- G-Means.

El algoritmo G-Means recibe su nombre de Gaussian-Means (que vendría a traducirse por medias gausianas). Su objetivo es descubrir por sí mismo la cantidad de agrupaciones o clústeres óptimos en un problema de Aprendizaje Automático No Supervisado. Realiza sucesivas iteraciones, partiendo de un único grupo o clúster, dividiendo en cada una de ellas los datos en dos nuevos grupos mientras que no detecte que sus datos adquieren una distribución gausiana.

Es decir, que va buscando nuevos centroides de nuevos cluster dividendo en dos el o los clústeres obtenidos en la iteración anterior hasta que logra crear todas las divisiones posibles de tal manera que los elementos de cada uno de esos clústeres conserva una distribución más o menos homogénea (matemáticamente se dice que dichos elementos sigue una distribución gausiana, de ahí el nombre del algoritmo).



Fran Bartolomé - Elaboración propia (CC BY-SA)

Este tipo de algoritmo es muy útil cuando en los casos de Aprendizaje Automático No Supervisado no tenemos ni idea de las relaciones que pueden llegar a existir entre los datos de nuestras instancias.