

The background of the image is a dark grey or black color, overlaid with a large, semi-transparent red diagonal shape that runs from the top-left towards the bottom-right. In the background, there is faint, light-colored Python code. Visible snippets include: 'self.logger = logging...', 'path:', 'self.file = open(os.path...', 'self.file.seek(0)', 'self.fingerprints...', 'ssmethod', 'from...', 'debug', 'return', 'request = se...', 'fp = s...', 'if fp...', 'return True', 'self.fingerprints.add(fp)', 'if self.file:', 'self.file.write(fp + os.l...', 'def request_fingerprint(self, request)', and 'return request_fingerprint...'.

MACHINE LEARNING

www.postparaprogramadores.com

Contenido

Machine Learning - Inicio

Machine Learning - Introducción

¿Qué puede hacer la IA de hoy?

Aprendizaje automático: IA tradicional

¿Qué es el aprendizaje automático?

Aprendizaje automático - Categorías

Aprendizaje automático: supervisado

**Aprendizaje automático: algoritmo de
aprendizaje de Scikit**

Aprendizaje automático: sin supervisión

Redes neuronales artificiales

Aprendizaje automático: aprendizaje profundo

Aprendizaje automático: habilidades

Aprendizaje automático: implementación

Aprendizaje automático: conclusión

Machine Learning - Introducción

La Inteligencia Artificial (IA) de hoy ha superado con creces la exageración de blockchain y la computación cuántica. Esto se debe al hecho de que los recursos informáticos enormes están fácilmente disponibles para el hombre común. Los desarrolladores ahora aprovechan esto para crear nuevos modelos de Machine Learning y volver a entrenar los modelos existentes para un mejor rendimiento y resultados. La fácil disponibilidad de High Performance Computing (HPC) ha resultado en un aumento repentino de la demanda de profesionales de TI con habilidades de Machine Learning.

En este tutorial, aprenderá en detalle sobre:

¿Cuál es el quid del aprendizaje automático?

- ¿Cuáles son los diferentes tipos de aprendizaje automático?
- ¿Cuáles son los diferentes algoritmos disponibles para desarrollar modelos de aprendizaje automático?
- ¿Qué herramientas están disponibles para desarrollar estos modelos?
- ¿Cuáles son las opciones de lenguaje de programación?
- ¿Qué plataformas admiten el desarrollo y la implementación de aplicaciones de Machine Learning?
- ¿Qué IDE (Entorno de desarrollo integrado) están disponibles?
- ¿Cómo mejorar rápidamente tus habilidades en esta importante área?

Descarga más libros de programación GRATIS [click aquí](#)



Síguenos en Instagram para que estés al tanto de los nuevos libros de programación. [Click aqui](#)

Aprendizaje automático: ¿qué puede hacer la IA de hoy?

Cuando etiqueta una cara en una foto de Facebook, es la IA la que se ejecuta detrás de escena e identifica las caras en una imagen. El etiquetado de rostros ahora está omnipresente en varias aplicaciones que muestran imágenes con rostros humanos. ¿Por qué solo rostros humanos? Existen varias aplicaciones que detectan objetos como gatos, perros, botellas, automóviles, etc. Tenemos automóviles autónomos que circulan por nuestras carreteras que detectan objetos en tiempo real para conducir el automóvil. Cuando viaja, utiliza las **Direcciones de Google** para conocer las situaciones de tráfico en tiempo real y seguir el mejor camino sugerido por Google en ese momento. Esta es otra implementación de la técnica de detección de objetos en tiempo real.

Consideremos el ejemplo de la aplicación Google **Translate** que usualmente usamos cuando visitamos países extranjeros. La aplicación de traductor en línea de Google en su dispositivo móvil lo ayuda a comunicarse con la gente local que habla un idioma que le es extraño.

Hay varias aplicaciones de IA que utilizamos prácticamente hoy. De hecho, cada uno de nosotros usamos IA en muchas partes de nuestras vidas, incluso sin nuestro conocimiento. La IA de hoy puede realizar trabajos extremadamente complejos con una gran precisión y velocidad. Discutamos un ejemplo de tarea compleja para comprender qué capacidades se esperan en una aplicación de inteligencia artificial que desarrollaría hoy para sus clientes.

Ejemplo

Todos usamos Google **Directions** durante nuestro viaje a cualquier lugar de la ciudad para un viaje diario o incluso para viajes interurbanos. La aplicación Google Directions sugiere la ruta más rápida a nuestro destino en ese momento. Cuando seguimos este camino, hemos observado que Google tiene casi el 100% de razón en sus sugerencias y ahorramos nuestro valioso tiempo en el viaje.

Puede imaginar la complejidad involucrada en el desarrollo de este tipo de aplicación considerando que hay múltiples rutas hacia su destino y la aplicación tiene que juzgar la situación del tráfico en cada ruta posible para darle una estimación del tiempo de viaje para cada ruta. Además, considere el hecho de que Google Directions cubre todo el mundo. Sin lugar a dudas, muchas de las técnicas de inteligencia artificial y aprendizaje automático están en uso bajo el capó de tales aplicaciones.

Teniendo en cuenta la continua demanda para el desarrollo de tales aplicaciones, ahora apreciará por qué hay una repentina demanda de profesionales de TI con habilidades de inteligencia artificial.

En nuestro próximo capítulo, aprenderemos lo que se necesita para desarrollar programas de IA.

Aprendizaje automático: IA tradicional

El viaje de la IA comenzó en la década de 1950 cuando la potencia informática era una fracción de lo que es hoy. Al comenzó con las predicciones hechas por la máquina de una manera que un estadístico hace predicciones usando su calculadora. Por lo tanto, todo el desarrollo inicial de IA se basó principalmente en técnicas estadísticas.

En este capítulo, analicemos en detalle cuáles son estas técnicas estadísticas.

Técnicas estadísticas

El desarrollo de las aplicaciones actuales de IA comenzó con el uso de las técnicas estadísticas tradicionales de antaño. Debe haber utilizado la interpolación en línea recta en las escuelas para predecir un valor futuro. Existen varias otras técnicas estadísticas que se aplican con éxito en el desarrollo de los llamados programas de IA. Decimos "supuestos" porque los programas de IA que tenemos hoy son mucho más complejos y utilizan técnicas mucho más allá de las técnicas estadísticas utilizadas por los primeros programas de IA.

Aquí se enumeran algunos de los ejemplos de técnicas estadísticas que se utilizan para desarrollar aplicaciones de IA en esos días y que todavía están en práctica:

- Regresión
- Clasificación
- Agrupación
- Teorías de probabilidad
- Árboles de decisión

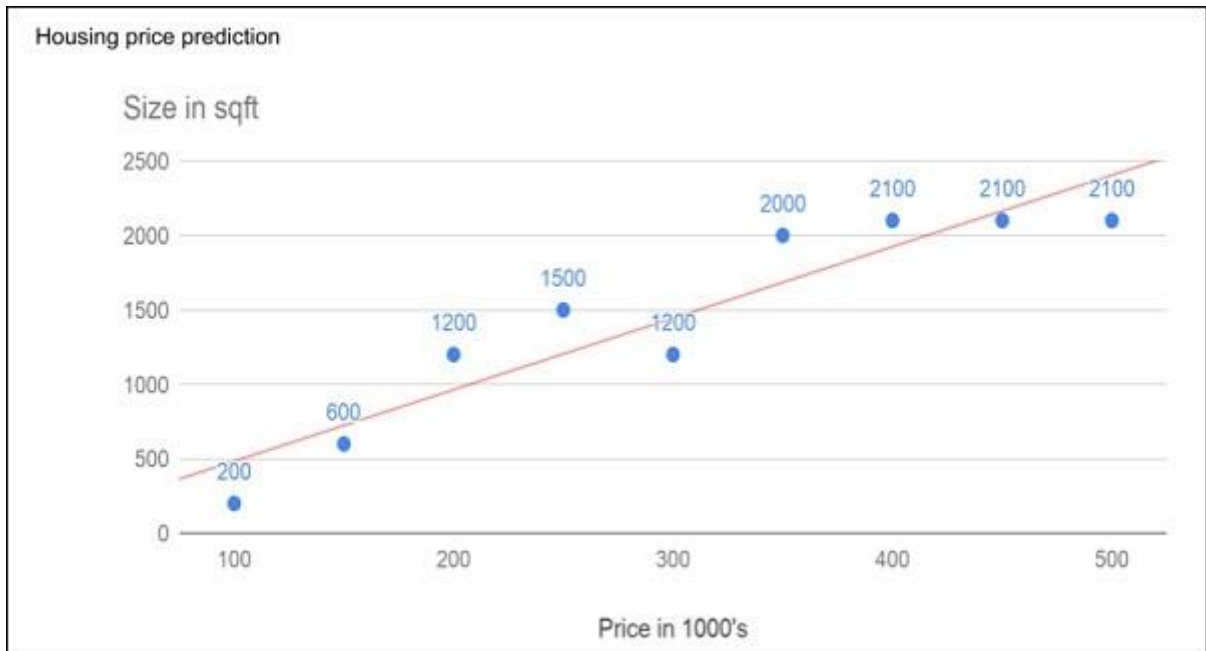
Aquí hemos enumerado solo algunas técnicas principales que son suficientes para que comiences con la IA sin asustarte de la inmensidad que exige la IA. Si está desarrollando aplicaciones de IA basadas en datos limitados, estaría utilizando estas técnicas estadísticas.

Sin embargo, hoy los datos son abundantes. Analizar el tipo de gran cantidad de datos que poseemos, las técnicas estadísticas no son de mucha ayuda ya que tienen algunas limitaciones propias. Por lo tanto, se desarrollan métodos más avanzados como el aprendizaje profundo para resolver muchos problemas complejos.

A medida que avancemos en este tutorial, entenderemos qué es Machine Learning y cómo se usa para desarrollar aplicaciones de IA tan complejas.

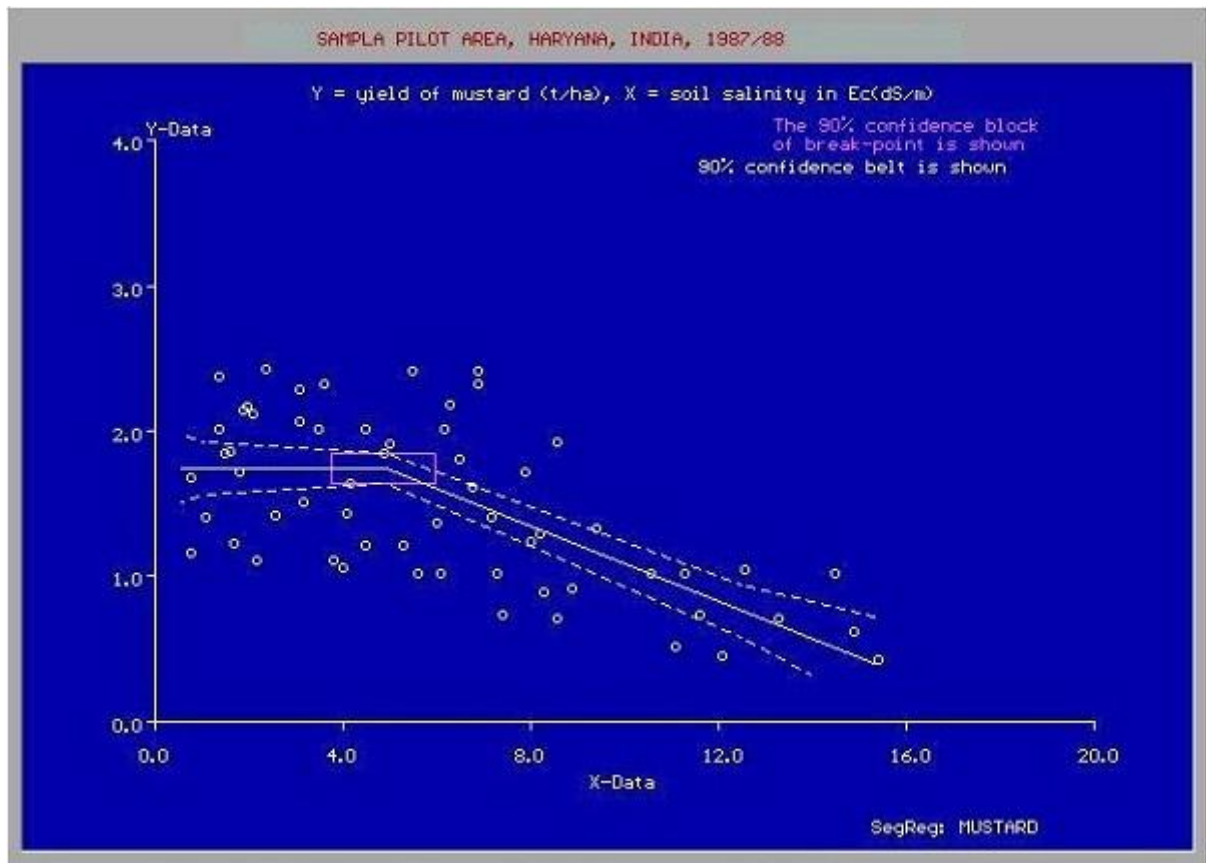
Aprendizaje automático: ¿qué es el aprendizaje automático?

Considere la siguiente figura que muestra una gráfica de los precios de la vivienda versus su tamaño en pies cuadrados.



Después de trazar varios puntos de datos en el diagrama XY, dibujamos una línea de mejor ajuste para hacer nuestras predicciones para cualquier otra casa dado su tamaño. Alimentará los datos conocidos a la máquina y le pedirá que encuentre la mejor línea de ajuste. Una vez que la máquina encuentre la línea de mejor ajuste, probará su idoneidad introduciendo un tamaño de casa conocido, es decir, el valor Y en la curva anterior. La máquina ahora devolverá el valor X estimado, es decir, el precio esperado de la casa. El diagrama se puede extrapolar para averiguar el precio de una casa que es de 3000 pies cuadrados o incluso más grande. Esto se llama regresión en estadística. Particularmente, este tipo de regresión se llama regresión lineal ya que la relación entre los puntos de datos X e Y es lineal.

En muchos casos, la relación entre los puntos de datos X e Y puede no ser una línea recta, y puede ser una curva con una ecuación compleja. Su tarea ahora sería encontrar la curva de mejor ajuste que se pueda extrapolar para predecir los valores futuros. Una de estas parcelas de aplicación se muestra en la figura a continuación.



Fuente:

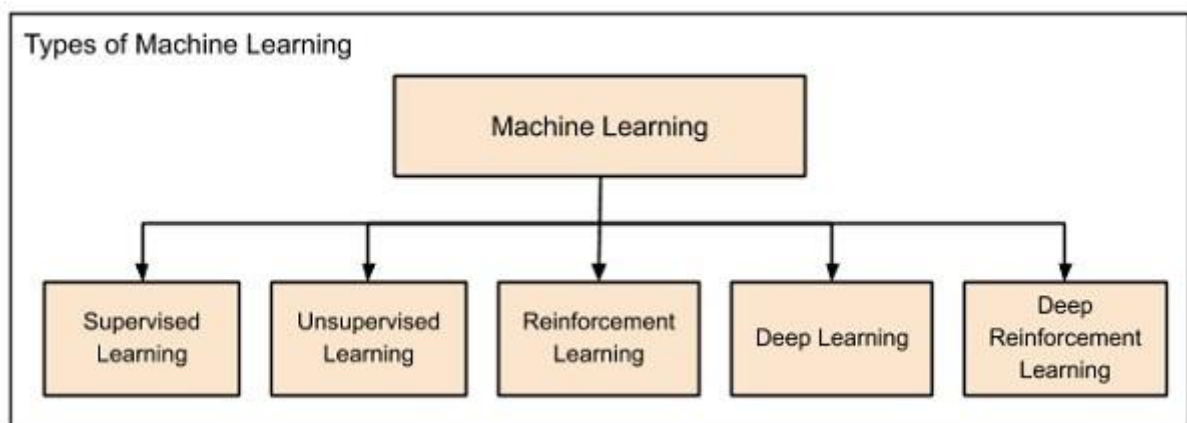
<https://upload.wikimedia.org/wikipedia/commons/c/c9/>

Utilizará las técnicas de optimización estadística para encontrar la ecuación para la curva de mejor ajuste aquí. Y de esto se trata exactamente Machine Learning. Utiliza técnicas de optimización conocidas para encontrar la mejor solución a su problema.

A continuación, veamos las diferentes categorías de Machine Learning.

Aprendizaje automático - Categorías

El aprendizaje automático se clasifica en términos generales en los siguientes títulos:



El aprendizaje automático evolucionó de izquierda a derecha como se muestra en el diagrama anterior.

- Inicialmente, los investigadores comenzaron con el aprendizaje supervisado. Este es el caso de la predicción del precio de la vivienda discutido anteriormente.
- Esto fue seguido por un aprendizaje no supervisado, donde la máquina está hecha para aprender por sí sola sin ninguna supervisión.
- Los científicos descubrieron además que puede ser una buena idea recompensar a la máquina cuando hace el trabajo de la manera esperada y llegó el aprendizaje de refuerzo.
- Muy pronto, los datos que están disponibles en estos días se han vuelto tan enormes que las técnicas convencionales desarrolladas hasta ahora no pudieron analizar los datos grandes y proporcionarnos las predicciones.
- Así, llegó el aprendizaje profundo donde el cerebro humano se simula en las Redes Neuronales Artificiales (ANN) creadas en nuestras computadoras binarias.
- La máquina ahora aprende por sí sola utilizando la alta potencia de cómputo y los enormes recursos de memoria disponibles en la actualidad.
- Ahora se observa que Deep Learning ha resuelto muchos de los problemas que antes no se podían resolver.
- La técnica ahora está más avanzada al dar incentivos a las redes de aprendizaje profundo como premios y finalmente llega el aprendizaje de refuerzo profundo.

Estudiamos ahora cada una de estas categorías con más detalle.

Aprendizaje supervisado

El aprendizaje supervisado es análogo a entrenar a un niño a caminar. Sostendrá la mano del niño, le mostrará cómo adelantar el pie, caminará para una demostración, etc., hasta que el niño aprenda a caminar solo.

Regresión

Del mismo modo, en el caso del aprendizaje supervisado, usted da ejemplos concretos conocidos a la computadora. Usted dice que para el valor de característica dado x_1 la salida es y_1 , para x_2 es y_2 , para x_3 es y_3 , y así sucesivamente. Con base en estos datos, usted deja que la computadora descubra una relación empírica entre x e y .

Una vez que la máquina esté entrenada de esta manera con un número suficiente de puntos de datos, ahora le pediría a la máquina que prediga Y para una X dada. Suponiendo que conozca el valor real de Y para esta X dada, podrá deducir si la predicción de la máquina es correcta.

Por lo tanto, probará si la máquina ha aprendido utilizando los datos de prueba conocidos. Una vez que esté satisfecho de que la máquina puede hacer las predicciones con un nivel deseado de precisión (digamos 80 a 90%), puede dejar de seguir entrenando la máquina.

Ahora, puede usar la máquina de manera segura para hacer las predicciones en puntos de datos desconocidos, o pedirle a la máquina que prediga Y para una X dada para la cual no conoce el valor real de Y . Esta capacitación viene bajo la regresión de la que hablamos más temprano.

Clasificación

También puede usar técnicas de aprendizaje automático para problemas de clasificación. En problemas de clasificación, clasifica objetos de naturaleza similar en un solo grupo. Por ejemplo, en un conjunto de 100 estudiantes, puede agruparlos en tres grupos según su altura: corta, media y larga. Midiendo la altura de cada alumno, los colocará en un grupo adecuado.

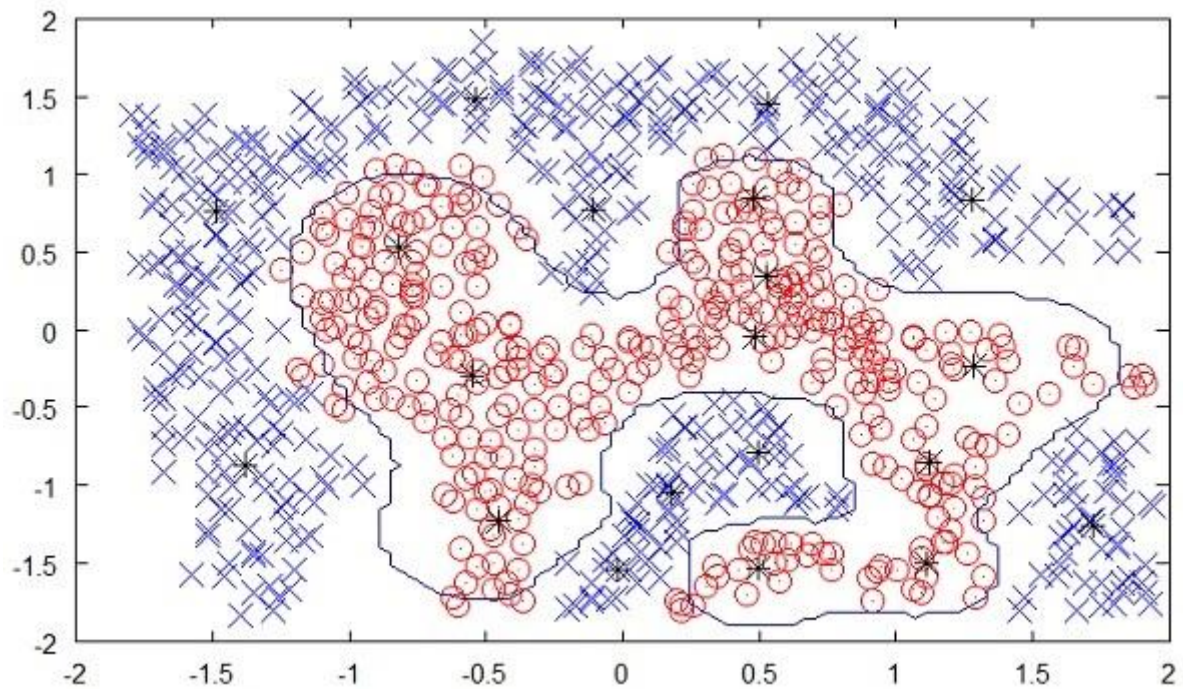
Ahora, cuando entra un nuevo estudiante, lo colocará en un grupo apropiado midiendo su altura. Al seguir los principios del entrenamiento de regresión, entrenará la máquina para clasificar a un alumno según su característica: la altura. Cuando la máquina aprende cómo se forman los grupos, podrá clasificar a cualquier nuevo estudiante desconocido correctamente. Una vez más, usaría los datos de prueba para verificar que la máquina haya aprendido su técnica de clasificación antes de poner en producción el modelo desarrollado.

El aprendizaje supervisado es donde la IA realmente comenzó su viaje. Esta técnica se aplicó con éxito en varios casos. Ha utilizado este modelo mientras realizaba el reconocimiento escrito a mano en su máquina. Se han desarrollado varios algoritmos para el aprendizaje supervisado. Aprenderá sobre ellos en los siguientes capítulos.

Aprendizaje sin supervisión

En el aprendizaje no supervisado, no especificamos una variable objetivo para la máquina, sino que le preguntamos a la máquina "¿Qué me puede decir sobre X ?". Más específicamente, podemos hacer preguntas como, dado un gran conjunto de datos X , "¿Cuáles son los cinco mejores grupos que podemos hacer de X ?" O "¿Qué características ocurren juntas con mayor frecuencia en X ?". Para llegar a las respuestas a tales preguntas, puede comprender que la cantidad de puntos de datos que la máquina necesitaría para deducir una estrategia sería muy grande. En caso de aprendizaje supervisado, la máquina se puede entrenar incluso con unos pocos miles de puntos de datos. Sin embargo, en caso de aprendizaje no supervisado, el número de puntos de datos que se aceptan razonablemente para el aprendizaje comienza en unos pocos millones. En estos días, los datos generalmente están abundantemente disponibles. La información idealmente requiere curación. Sin embargo,

La siguiente figura muestra el límite entre los puntos amarillo y rojo según lo determinado por el aprendizaje automático no supervisado. Puede ver claramente que la máquina podría determinar la clase de cada uno de los puntos negros con una precisión bastante buena.



Fuente:

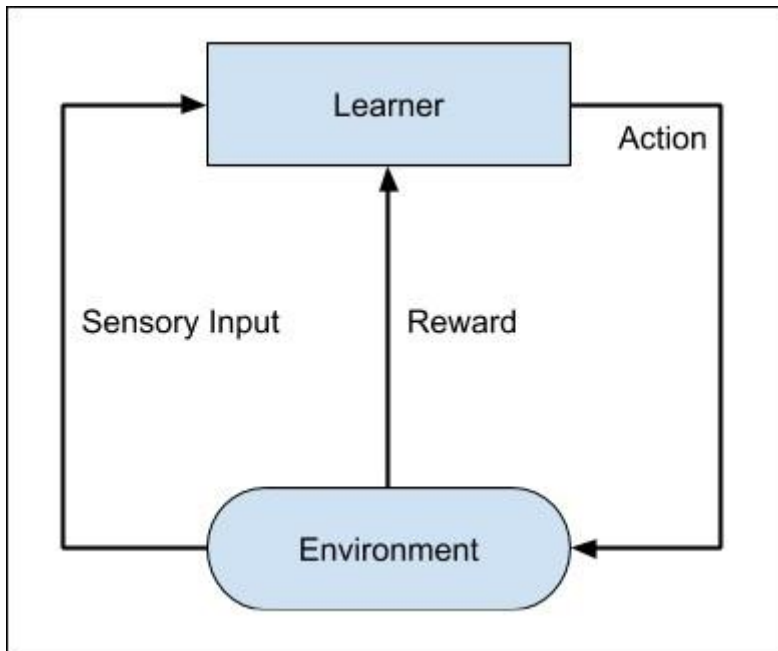
https://chrisjmccormick.files.wordpress.com/2013/08/approx_decision_boundary.png

El aprendizaje no supervisado ha demostrado un gran éxito en muchas aplicaciones modernas de inteligencia artificial, como la detección de rostros, la detección de objetos, etc.

Aprendizaje reforzado

Considere entrenar a un perro mascota, entrenamos a nuestra mascota para que nos traiga una pelota. Lanzamos la pelota a cierta distancia y le pedimos al perro que nos la devuelva. Cada vez que el perro hace esto bien, recompensamos al perro. Poco a poco, el perro aprende que hacer el trabajo correctamente le da una recompensa y luego el perro comienza a hacer el trabajo correctamente cada vez en el futuro. Exactamente, este concepto se aplica en el tipo de aprendizaje de "refuerzo". La técnica se desarrolló inicialmente para máquinas para jugar juegos. La máquina recibe un algoritmo para analizar todos los movimientos posibles en cada etapa del juego. La máquina puede seleccionar uno de los movimientos al azar. Si el movimiento es correcto, la máquina es recompensada, de lo contrario puede ser penalizada. Despacio, la máquina comenzará a diferenciar entre movimientos correctos e incorrectos y después de varias iteraciones aprenderá a resolver el rompecabezas del juego con mayor precisión. La precisión de ganar el juego mejoraría a medida que la máquina juegue más y más juegos.

Todo el proceso se puede representar en el siguiente diagrama:



Esta técnica de aprendizaje automático difiere del aprendizaje supervisado en que no necesita suministrar los pares de entrada / salida etiquetados. El objetivo es encontrar el equilibrio entre explorar las nuevas soluciones y explotar las soluciones aprendidas.

Aprendizaje profundo

El aprendizaje profundo es un modelo basado en Redes Neuronales Artificiales (ANN), más específicamente Redes Neuronales Convolucionales (CNN). Hay varias arquitecturas utilizadas en el aprendizaje profundo, tales como redes neuronales profundas, redes de creencias profundas, redes neuronales recurrentes y redes neuronales convolucionales.

Estas redes se han aplicado con éxito para resolver los problemas de visión por computadora, reconocimiento de voz, procesamiento del lenguaje natural, bioinformática, diseño de medicamentos, análisis de imágenes médicas y juegos. Hay varios otros campos en los que el aprendizaje profundo se aplica de manera proactiva. El aprendizaje profundo requiere una gran potencia de procesamiento y datos enormes, que generalmente están fácilmente disponibles en estos días.

Hablaremos sobre el aprendizaje profundo con más detalle en los próximos capítulos.

Aprendizaje de refuerzo profundo

El aprendizaje de refuerzo profundo (DRL) combina las técnicas de aprendizaje profundo y de refuerzo. Los algoritmos de aprendizaje de refuerzo como Q-learning ahora se combinan con el aprendizaje profundo para crear un poderoso modelo DRL. La técnica ha tenido un gran éxito en los campos de la robótica, los videojuegos, las finanzas y la asistencia sanitaria. Muchos problemas que antes no se podían resolver ahora se resuelven creando

modelos DRL. Hay mucha investigación en curso en esta área y esto es muy activo por las industrias.

Hasta ahora, tiene una breve introducción a varios modelos de aprendizaje automático, ahora permítanos explorar un poco más profundamente en varios algoritmos que están disponibles bajo estos modelos.

Aprendizaje automático: supervisado

El aprendizaje supervisado es uno de los modelos importantes de aprendizaje involucrados en las máquinas de entrenamiento. Este capítulo habla en detalle sobre lo mismo.

Algoritmos para el aprendizaje supervisado

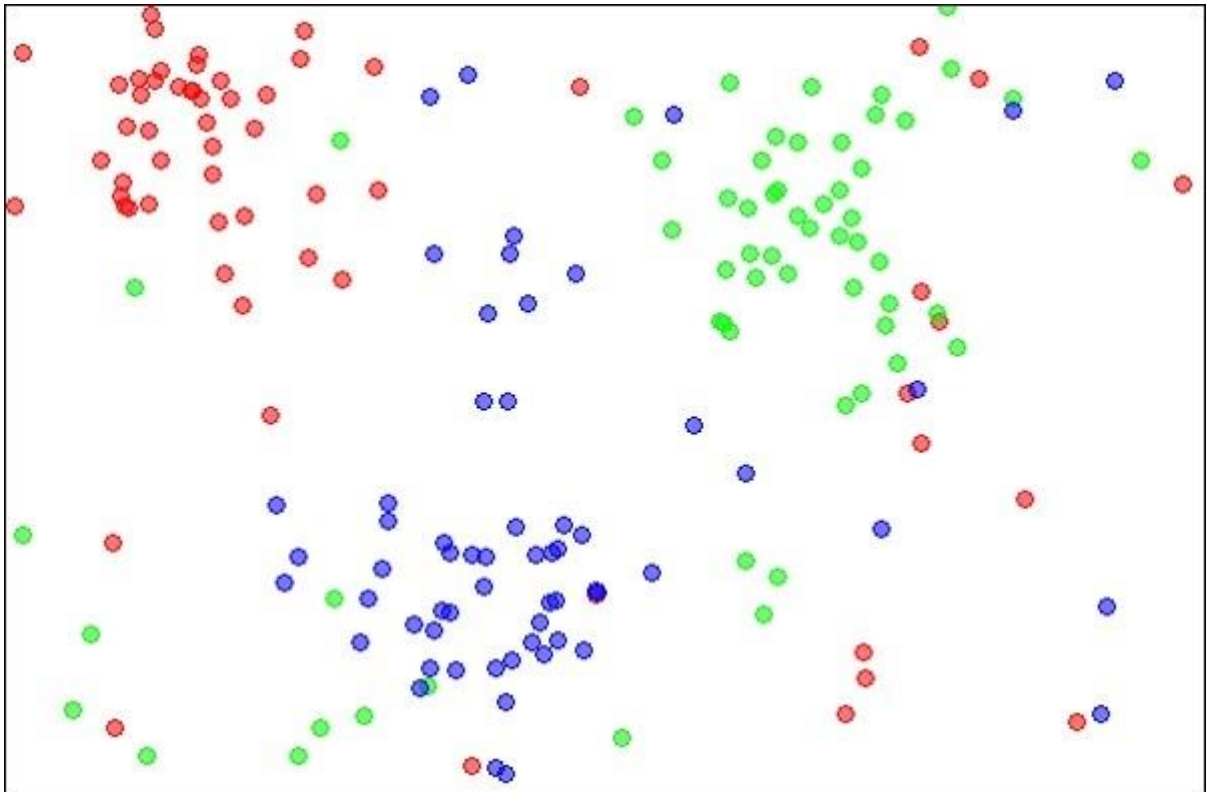
Hay varios algoritmos disponibles para el aprendizaje supervisado. Algunos de los algoritmos de aprendizaje supervisado ampliamente utilizados se muestran a continuación:

- k-Vecinos más cercanos
- Árboles de decisión
- Bayes ingenuos
- Regresión logística
- Máquinas de vectores de soporte

A medida que avanzamos en este capítulo, analicemos en detalle cada uno de los algoritmos.

k-Vecinos más cercanos

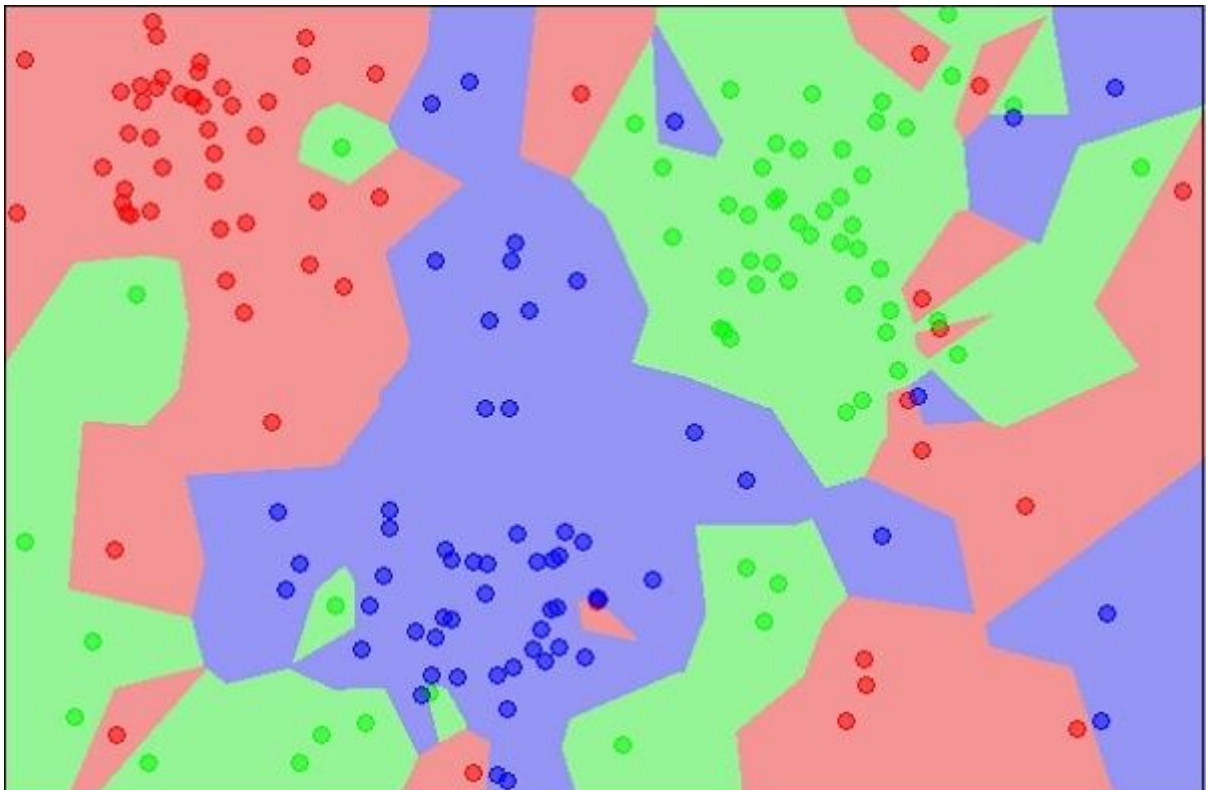
Los vecinos más cercanos a k , que simplemente se llama kNN, es una técnica estadística que se puede utilizar para resolver problemas de clasificación y regresión. Discutamos el caso de clasificar un objeto desconocido usando kNN. Considere la distribución de objetos como se muestra en la imagen a continuación:



Fuente:

https://en.wikipedia.org/wiki/K-nearest_neighbours_algorithm

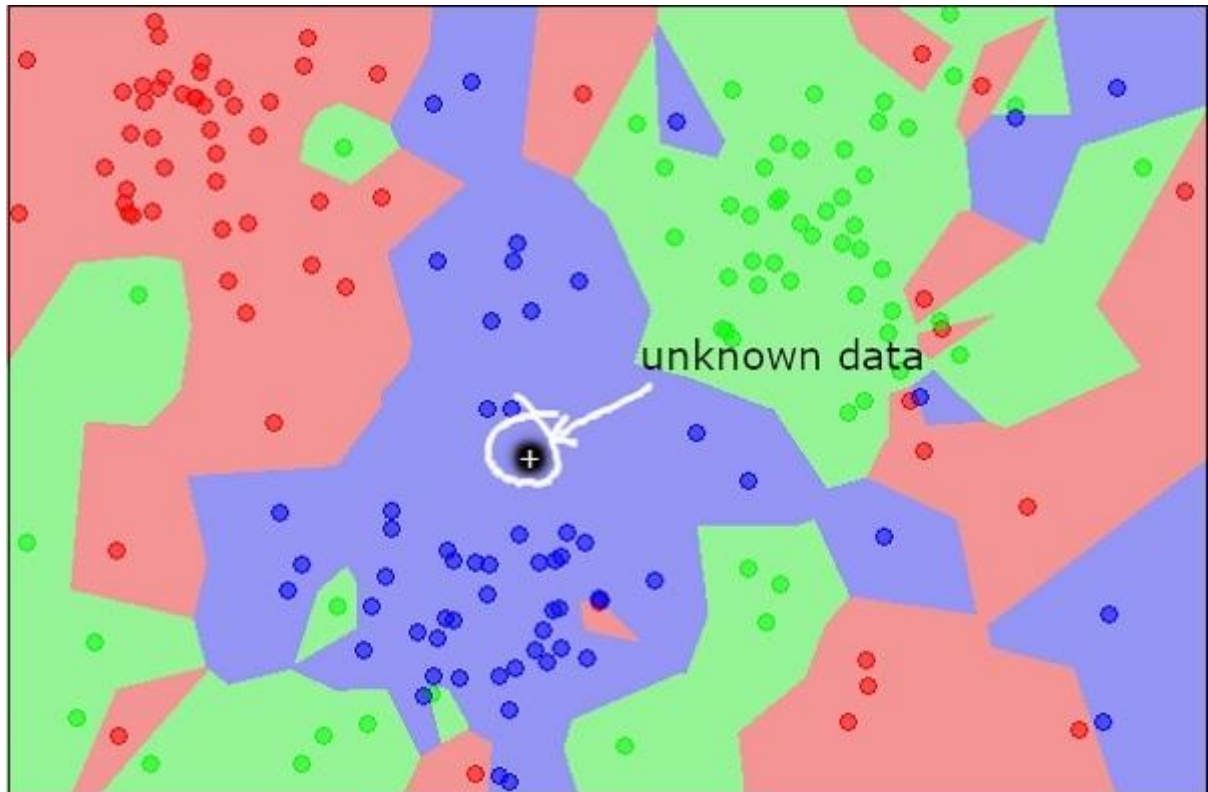
El diagrama muestra tres tipos de objetos, marcados en colores rojo, azul y verde. Cuando ejecuta el clasificador kNN en el conjunto de datos anterior, los límites para cada tipo de objeto se marcarán como se muestra a continuación:



Fuente:

https://en.wikipedia.org/wiki/K-nearest_neighbours_algorithm

Ahora, considere un nuevo objeto desconocido que desea clasificar como rojo, verde o azul. Esto se representa en la figura a continuación.

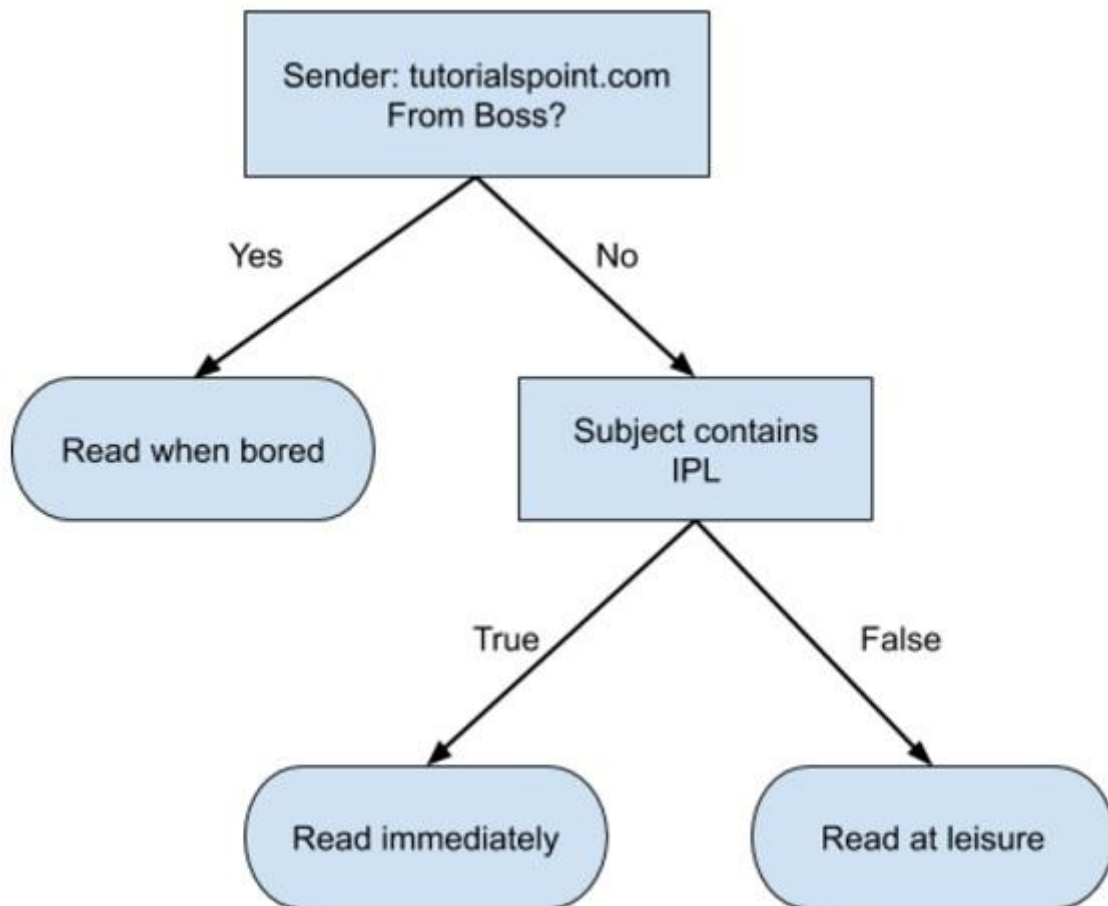


Como lo ve visualmente, el punto de datos desconocido pertenece a una clase de objetos azules. Matemáticamente, esto se puede concluir midiendo la distancia de este punto desconocido con cualquier otro punto en el conjunto de datos. Cuando lo haga, sabrá que la mayoría de sus vecinos son de color azul. La distancia promedio a los objetos rojos y verdes sería definitivamente mayor que la distancia promedio a los objetos azules. Por lo tanto, este objeto desconocido se puede clasificar como perteneciente a la clase azul.

El algoritmo kNN también se puede usar para problemas de regresión. El algoritmo kNN está disponible como listo para usar en la mayoría de las bibliotecas ML.

Árboles de decisión

A continuación se muestra un árbol de decisión simple en formato de diagrama de flujo:



Escribiría un código para clasificar sus datos de entrada en función de este diagrama de flujo. El diagrama de flujo se explica por sí mismo y es trivial. En este escenario, está intentando clasificar un correo electrónico entrante para decidir cuándo leerlo.

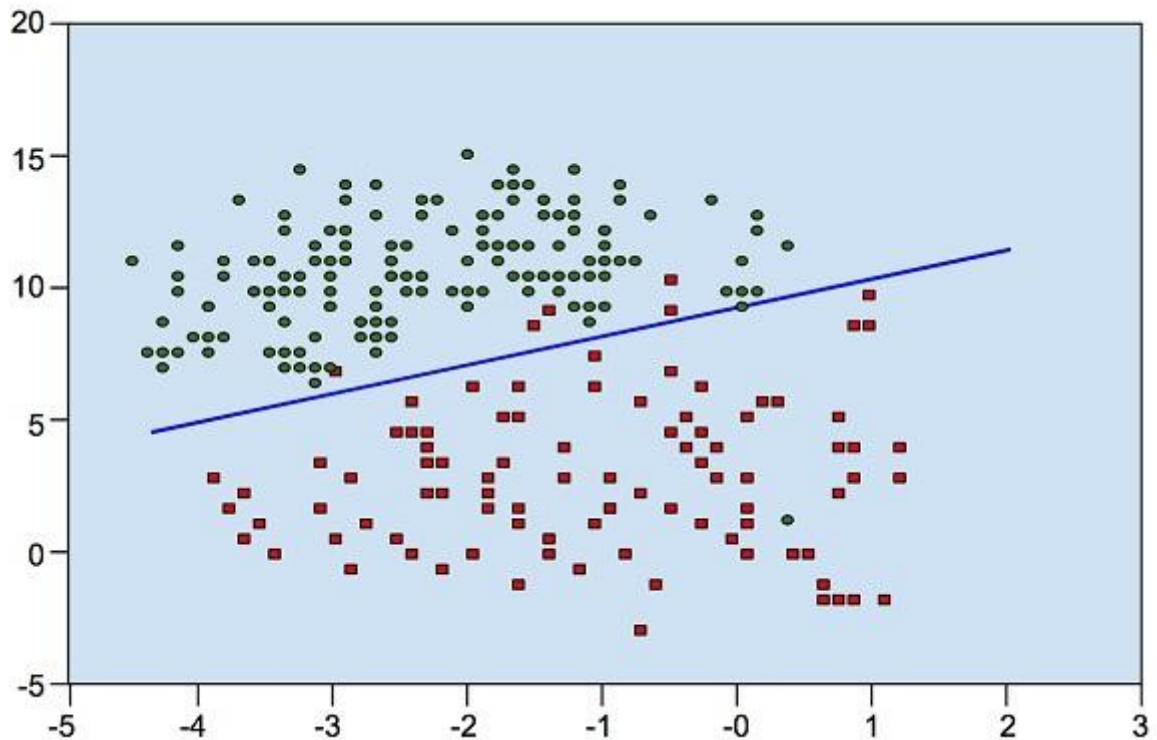
En realidad, los árboles de decisión pueden ser grandes y complejos. Hay varios algoritmos disponibles para crear y atravesar estos árboles. Como entusiasta del aprendizaje automático, debe comprender y dominar estas técnicas para crear y atravesar árboles de decisión.

Bayes ingenuos

Naive Bayes se usa para crear clasificadores. Suponga que desea clasificar (clasificar) frutas de diferentes tipos de una cesta de frutas. Puede usar características como el color, el tamaño y la forma de una fruta. Por ejemplo, cualquier fruta que sea de color rojo, de forma redonda y de unos 10 cm de diámetro puede considerarse Apple. Por lo tanto, para entrenar el modelo, usaría estas características y probaría la probabilidad de que una característica dada coincida con las restricciones deseadas. Las probabilidades de diferentes características se combinan para llegar a una probabilidad de que una fruta dada sea una Apple. Naive Bayes generalmente requiere una pequeña cantidad de datos de entrenamiento para la clasificación.

Regresión logística

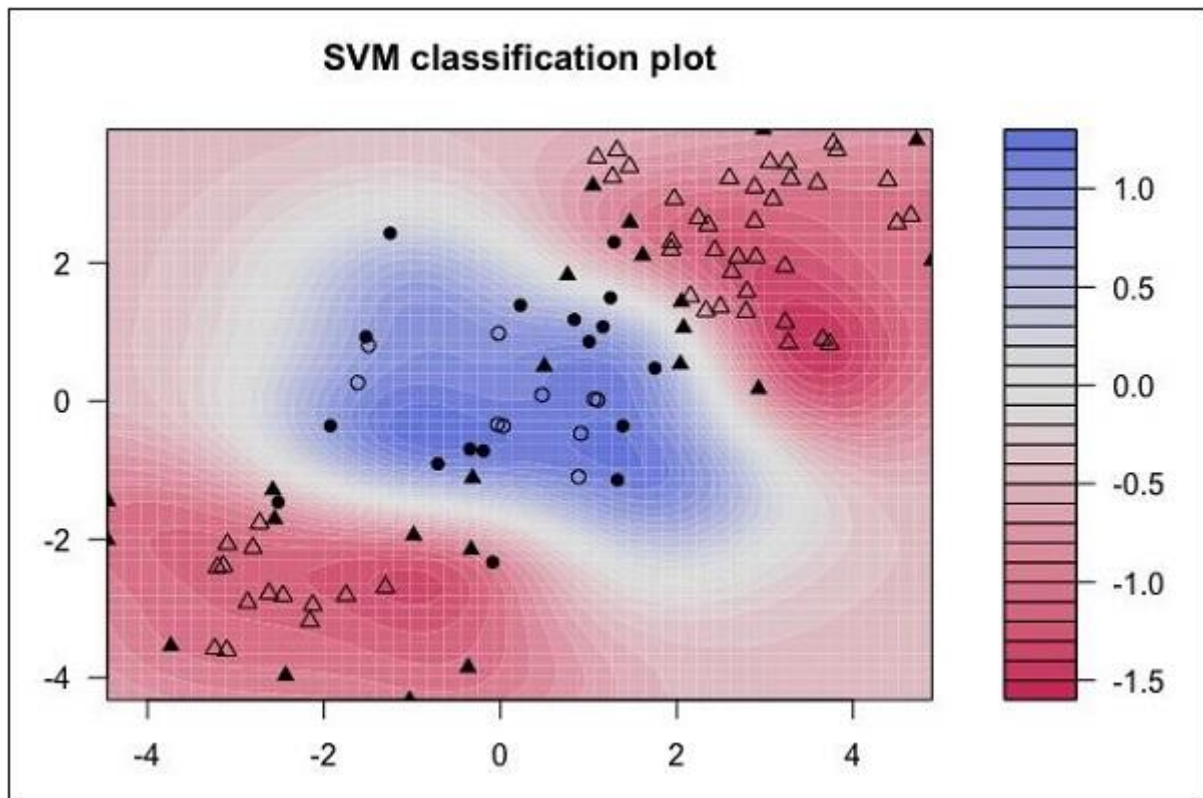
Mira el siguiente diagrama. Muestra la distribución de puntos de datos en el plano XY.



Desde el diagrama, podemos inspeccionar visualmente la separación de los puntos rojos de los puntos verdes. Puede dibujar una línea límite para separar estos puntos. Ahora, para clasificar un nuevo punto de datos, solo tendrá que determinar en qué lado de la línea se encuentra el punto.

Máquinas de vectores de soporte

Mire la siguiente distribución de datos. Aquí las tres clases de datos no se pueden separar linealmente. Las curvas de límite son no lineales. En tal caso, encontrar la ecuación de la curva se convierte en un trabajo complejo.



Fuente: <http://uc-r.github.io/svm>

Support Vector Machines (SVM) es útil para determinar los límites de separación en tales situaciones.

Aprendizaje automático: algoritmo de aprendizaje de Scikit

Afortunadamente, la mayoría de las veces no tiene que codificar los algoritmos mencionados en la lección anterior. Hay muchas bibliotecas estándar que proporcionan la implementación lista para usar de estos algoritmos. Uno de estos juegos de herramientas que se usa popularmente es scikit-learn. La siguiente figura ilustra el tipo de algoritmos que están disponibles para su uso en esta biblioteca.

con X?". O incluso podría ser como "¿Qué tres características ocurren juntas con mayor frecuencia en X?".

Este es exactamente el aprendizaje no supervisado.

Algoritmos para el aprendizaje no supervisado

Analicemos ahora uno de los algoritmos ampliamente utilizados para la clasificación en el aprendizaje automático no supervisado.

k-significa agrupamiento

Las elecciones presidenciales de 2000 y 2004 en los Estados Unidos estuvieron cerca, muy cerca. El mayor porcentaje del voto popular que recibió cualquier candidato fue del 50,7% y el más bajo fue del 47,9%. Si un porcentaje de los votantes hubiera cambiado de bando, el resultado de las elecciones habría sido diferente. Hay pequeños grupos de votantes que, cuando se les solicita adecuadamente, cambiarán de bando. Es posible que estos grupos no sean enormes, pero con carreras tan cercanas, pueden ser lo suficientemente grandes como para cambiar el resultado de las elecciones. ¿Cómo encuentras estos grupos de personas? ¿Cómo les atraes con un presupuesto limitado? La respuesta es agrupación.

Déjanos entender cómo se hace.

- Primero, usted recopila información sobre las personas con o sin su consentimiento: cualquier tipo de información que pueda dar alguna pista sobre lo que es importante para ellos y lo que influirá en su voto.
- Luego pones esta información en algún tipo de algoritmo de agrupamiento.
- A continuación, para cada grupo (sería inteligente elegir el más grande primero) elabore un mensaje que atraerá a estos votantes.
- Finalmente, entrega la campaña y mide para ver si está funcionando.

El agrupamiento es un tipo de aprendizaje no supervisado que forma automáticamente grupos de cosas similares. Es como la clasificación automática. Puede agrupar casi cualquier cosa, y cuanto más similares sean los elementos en el clúster, mejores serán los clústeres. En este capítulo, vamos a estudiar un tipo de algoritmo de agrupamiento llamado k-means. Se llama k-means porque encuentra 'k' grupos únicos, y el centro de cada grupo es la media de los valores en ese grupo.

Identificación de clúster

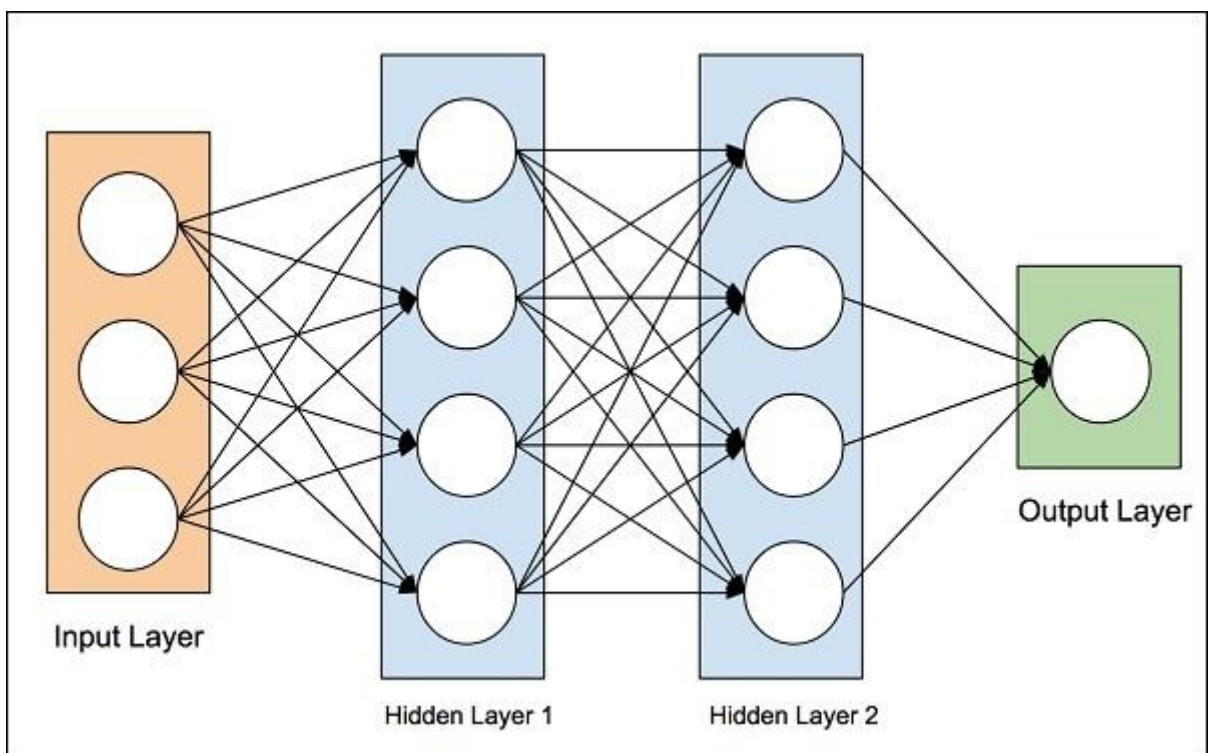
La identificación del clúster le dice a un algoritmo: "Aquí hay algunos datos. Ahora agrupe cosas similares y cuénteme sobre esos grupos ". La diferencia clave de la clasificación es que en la clasificación usted sabe lo que está buscando. Si bien ese no es el caso en la agrupación.

La agrupación a veces se denomina clasificación no supervisada porque produce el mismo resultado que la clasificación pero sin tener clases predefinidas.

Ahora, nos sentimos cómodos con el aprendizaje supervisado y no supervisado. Para comprender el resto de las categorías de aprendizaje automático, primero debemos comprender las Redes Neuronales Artificiales (ANN), que aprenderemos en el próximo capítulo.

Aprendizaje automático: redes neuronales artificiales

La idea de redes neuronales artificiales se derivó de las redes neuronales en el cerebro humano. El cerebro humano es realmente complejo. Estudiando cuidadosamente el cerebro, los científicos e ingenieros idearon una arquitectura que podría encajar en nuestro mundo digital de computadoras binarias. Una de estas arquitecturas típicas se muestra en el siguiente diagrama:



Hay una capa de entrada que tiene muchos sensores para recopilar datos del mundo exterior. En el lado derecho, tenemos una capa de salida que nos da el resultado predicho por la red. Entre estos dos, varias capas están ocultas. Cada capa adicional agrega más complejidad en el entrenamiento de la red, pero proporcionaría mejores resultados en la mayoría de las situaciones. Hay varios tipos de arquitecturas diseñadas que discutiremos ahora.

ANN Architectures

El diagrama a continuación muestra varias arquitecturas ANN desarrolladas durante un período de tiempo y que están en práctica hoy.

A mostly complete chart of Neural Networks

©2016 Fjodor van Veen - asimovinstitute.org

Backed Input Cell

Input Cell

Noisy Input Cell

Hidden Cell

Probabilistic Hidden Cell

Spiking Hidden Cell

Output Cell

Match Input Output Cell

Recurrent Cell

Memory Cell

Different Memory Cell

Kernel

Convolution or Pool

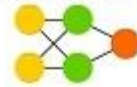
Perceptron (P)



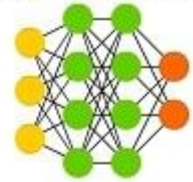
Feed Forward (FF)



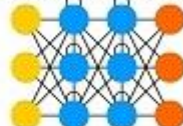
Radial Basis Network (RBF)



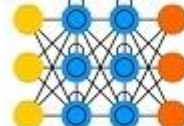
Deep Feed Forward (DFF)



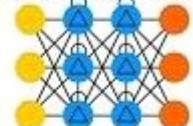
Recurrent Neural Network (RNN)



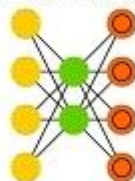
Long / Short Term Memory (LSTM)



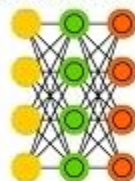
Gated Recurrent Unit (GRU)



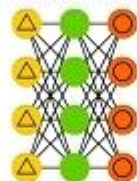
Auto Encoder (AE)



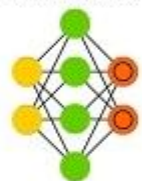
Variational AE (VAE)



Denoising AE (DAE)



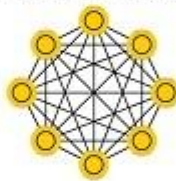
Sparse AE (SAE)



Markov Chain (MC)



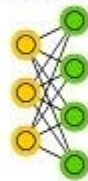
Hopfield Network (HN)



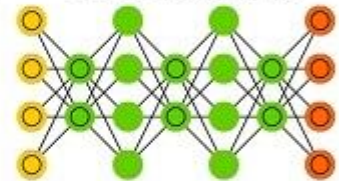
Boltzmann Machine (BM)



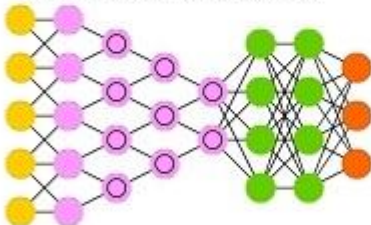
Restricted BM (RBM)



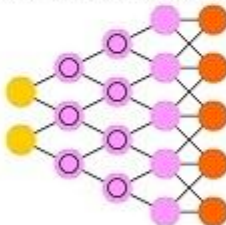
Deep Belief Network (DBN)



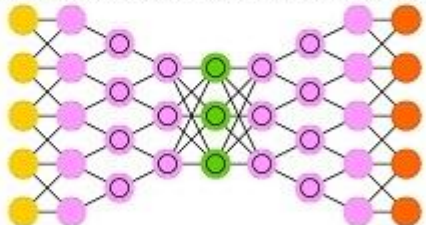
Deep Convolutional Network (DCN)



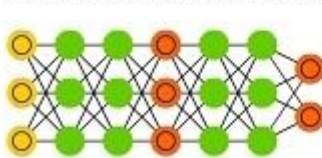
Deconvolutional Network (DN)



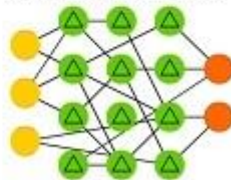
Deep Convolutional Inverse Graphics Network (DCIGN)



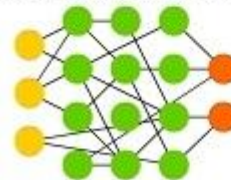
Generative Adversarial Network (GAN)



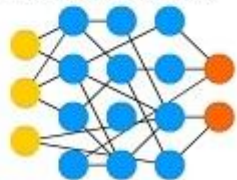
Liquid State Machine (LSM)



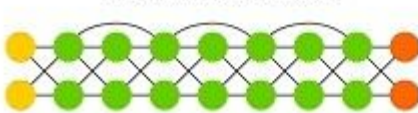
Extreme Learning Machine (ELM)



Echo State Network (ESN)



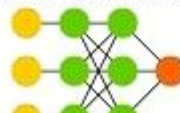
Deep Residual Network (DRN)



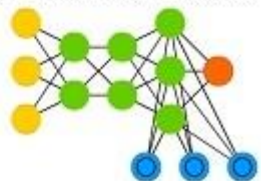
Kohonen Network (KN)



Support Vector Machine (SVM)



Neural Turing Machine (NTM)



Fuente:

<https://towardsdatascience.com/the-mostly-complete-chart-of-neural-networks-explained-3fb6f2367464>

Cada arquitectura está desarrollada para un tipo específico de aplicación. Por lo tanto, cuando use una red neuronal para su aplicación de aprendizaje automático, tendrá que usar una de las arquitecturas existentes o diseñar la suya propia. El tipo de aplicación que finalmente decide depende de las necesidades de su aplicación. No existe una guía única que le indique que use una arquitectura de red específica.

Aprendizaje automático: aprendizaje profundo

Deep Learning utiliza ANN. Primero, veremos algunas aplicaciones de aprendizaje profundo que le darán una idea de su poder.

Aplicaciones

Deep Learning ha demostrado mucho éxito en varias áreas de las aplicaciones de aprendizaje automático.

Automóviles autónomos: los automóviles autónomos utilizan técnicas de aprendizaje profundo. Por lo general, se adaptan a las situaciones de tráfico en constante cambio y mejoran cada vez más en la conducción durante un período de tiempo.

Reconocimiento de voz : otra aplicación interesante de Deep Learning es el reconocimiento de voz. Todos nosotros usamos varias aplicaciones móviles hoy en día que son capaces de reconocer nuestro discurso. Siri de Apple, Alexa de Amazon, Cortana de Microsoft y Asistente de Google: todos estos utilizan técnicas de aprendizaje profundo.

Aplicaciones móviles : utilizamos varias aplicaciones móviles y basadas en la web para organizar nuestras fotos. Detección de rostros, identificación de rostros, etiquetado de rostros, identificación de objetos en una imagen: todo esto utiliza el aprendizaje profundo.

Oportunidades desaprovechadas de aprendizaje profundo

Después de observar el gran éxito que las aplicaciones de aprendizaje profundo han logrado en muchos dominios, las personas comenzaron a explorar otros dominios donde el aprendizaje automático no se aplicaba hasta ahora. Hay varios dominios en los que las técnicas de aprendizaje profundo se aplican con éxito y hay muchos otros dominios que pueden explotarse. Algunos de estos se discuten aquí.

- La agricultura es una de esas industrias donde las personas pueden aplicar técnicas de aprendizaje profundo para mejorar el rendimiento de los cultivos.

- La financiación al consumo es otra área donde el aprendizaje automático puede ser de gran ayuda para proporcionar detección temprana de fraudes y analizar la capacidad de pago del cliente.
- Las técnicas de aprendizaje profundo también se aplican al campo de la medicina para crear nuevos medicamentos y proporcionar una receta personalizada a un paciente.

Las posibilidades son infinitas y hay que seguir observando a medida que aparecen nuevas ideas y desarrollos con frecuencia.

Lo que se requiere para lograr más con el aprendizaje profundo

Para utilizar el aprendizaje profundo, el poder de supercomputación es un requisito obligatorio. Necesita tanto la memoria como la CPU para desarrollar modelos de aprendizaje profundo. Afortunadamente, hoy tenemos una disponibilidad fácil de HPC: computación de alto rendimiento. Debido a esto, el desarrollo de las aplicaciones de aprendizaje profundo que mencionamos anteriormente se convirtió en una realidad hoy y en el futuro también podemos ver las aplicaciones en las áreas sin explotar que discutimos anteriormente.

Ahora, veremos algunas de las limitaciones del aprendizaje profundo que debemos considerar antes de usarlo en nuestra aplicación de aprendizaje automático.

Desventajas de aprendizaje profundo

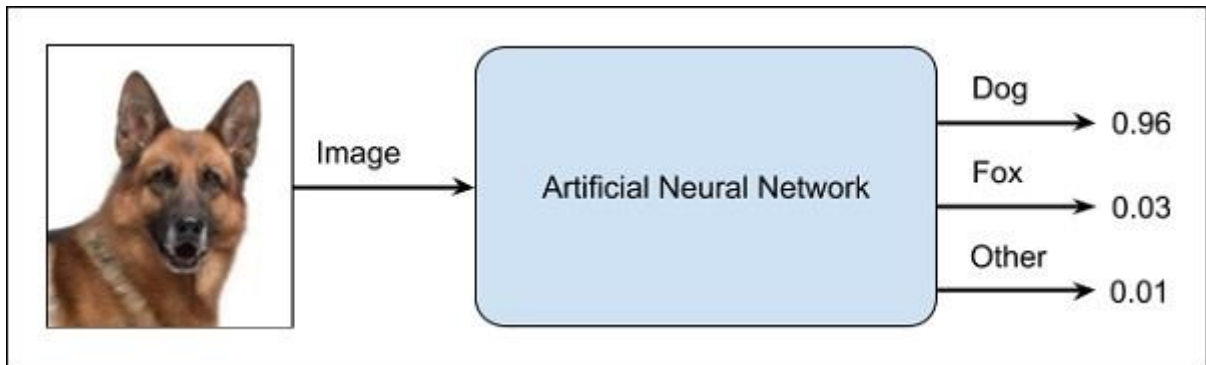
A continuación se enumeran algunos de los puntos importantes que debe tener en cuenta antes de utilizar el aprendizaje profundo:

- Enfoque de caja negra
- Duración del desarrollo
- La cantidad de datos
- Costoso computacionalmente

Ahora estudiaremos cada una de estas limitaciones en detalle.

Enfoque de caja negra

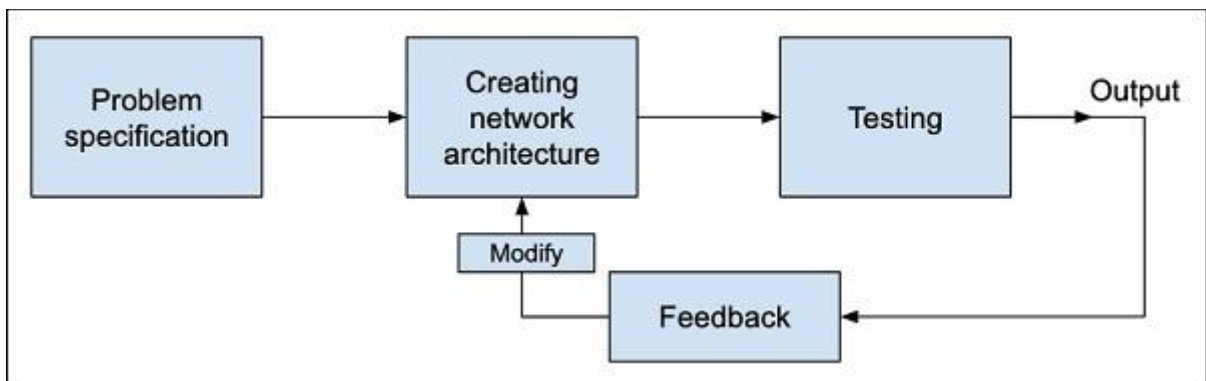
Un ANN es como una caja negra. Le das una cierta entrada y te proporcionará una salida específica. El siguiente diagrama le muestra una de esas aplicaciones donde alimenta una imagen de un animal a una red neuronal y le dice que la imagen es de un perro.



Por qué esto se llama un enfoque de recuadro negro es que no sabes por qué la red obtuvo un resultado determinado. ¿No sabes cómo la red concluyó que es un perro? Ahora considere una aplicación bancaria donde el banco quiera decidir la solvencia de un cliente. La red definitivamente le proporcionará una respuesta a esta pregunta. Sin embargo, ¿podrá justificarlo ante un cliente? Los bancos deben explicar a sus clientes por qué no se sanciona el préstamo.

Duración del desarrollo

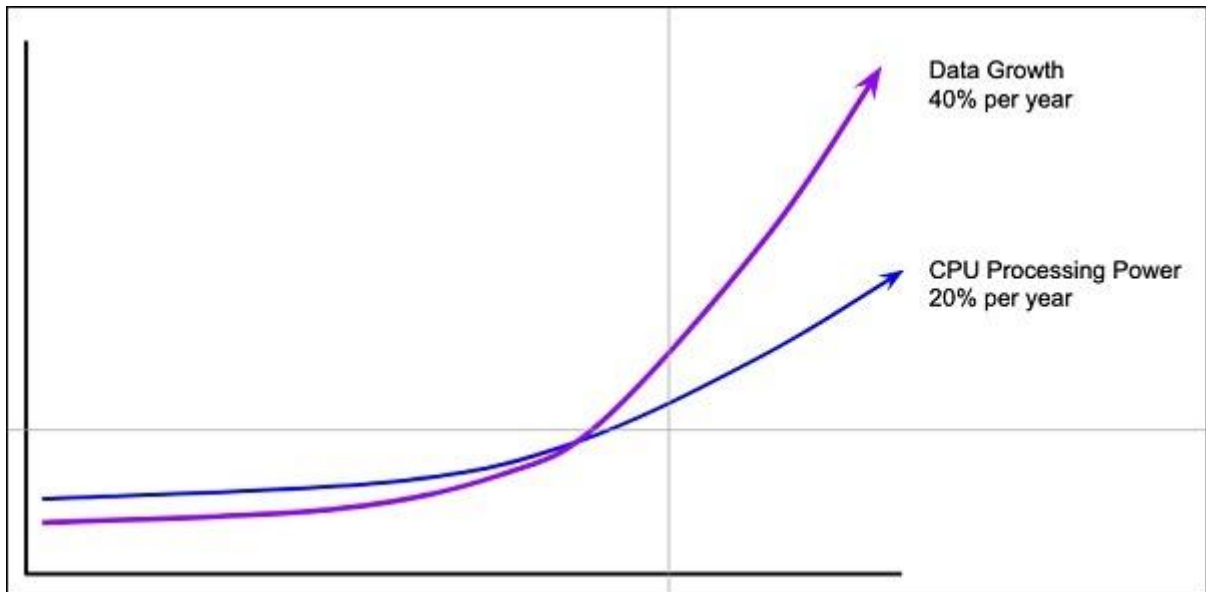
El proceso de entrenamiento de una red neuronal se muestra en el siguiente diagrama:



Primero define el problema que desea resolver, crea una especificación para él, decide las características de entrada, diseña una red, la implementa y prueba la salida. Si el resultado no es el esperado, tome esto como un comentario para reestructurar su red. Este es un proceso iterativo y puede requerir varias iteraciones hasta que la red de tiempo esté completamente entrenada para producir los resultados deseados.

La cantidad de datos

Las redes de aprendizaje profundo generalmente requieren una gran cantidad de datos para el entrenamiento, mientras que los algoritmos tradicionales de aprendizaje automático pueden usarse con gran éxito incluso con solo unos pocos miles de puntos de datos. Afortunadamente, la abundancia de datos está creciendo al 40% por año y la potencia de procesamiento de la CPU está creciendo al 20% por año, como se ve en el diagrama a continuación:



Costoso computacionalmente

Entrenar una red neuronal requiere varias veces más potencia computacional que la requerida para ejecutar algoritmos tradicionales. El entrenamiento exitoso de las redes neuronales profundas puede requerir varias semanas de tiempo de entrenamiento.

En contraste con esto, los algoritmos tradicionales de aprendizaje automático requieren solo unos minutos / horas para entrenar. Además, la cantidad de potencia computacional necesaria para entrenar redes neuronales profundas depende en gran medida del tamaño de sus datos y de la profundidad y complejidad de la red.

Después de tener una visión general de lo que es Machine Learning, sus capacidades, limitaciones y aplicaciones, pasemos ahora al aprendizaje "Machine Learning".

Aprendizaje automático: habilidades

El aprendizaje automático tiene un ancho muy grande y requiere habilidades en varios dominios. A continuación se enumeran las habilidades que necesita adquirir para convertirse en un experto en Machine Learning:

- Estadísticas
- Teorías de probabilidad
- Cálculo
- Técnicas de optimización
- Visualización

Necesidad de varias habilidades de aprendizaje automático

Para darle una breve idea de las habilidades que necesita adquirir, permítanos discutir algunos ejemplos:

Notación Matemática

La mayoría de los algoritmos de aprendizaje automático se basan en gran medida en las matemáticas. El nivel de matemáticas que necesita saber es probablemente solo un nivel de principiante. Lo importante es que puedas leer la notación que los matemáticos usan en sus ecuaciones. Por ejemplo, si puede leer la notación y comprender lo que significa, está listo para aprender el aprendizaje automático. De lo contrario, es posible que necesite repasar sus conocimientos de matemáticas.

$$F_{\text{Una N}(\text{net} - \theta)} = \int \bigcup \gamma \text{net} - \theta - \gamma \text{yo fn} \text{net} - \theta \geq \epsilon \text{yo f} - \epsilon < \text{net} - \theta < \epsilon \text{yo fn} \text{net} - \theta \leq -\epsilon \text{fAN}(\text{net} - \theta) = \{\gamma \text{if net} - \theta \geq \epsilon \text{net} - \theta \text{if} -\epsilon < \text{net} - \theta < \epsilon - \gamma \text{if net} - \theta \leq -\epsilon$$

$$\max_{\alpha} [\sum_{i=1}^m \alpha_i - 12 \sum_{i,j=1}^m \alpha_i \alpha_j \ell(i,j) \cdot \ell(j) \cdot \text{un}_{\text{yo}} \cdot \text{un}_j \langle x(i), x(j) \rangle] \max_{\alpha} [\sum_{i=1}^m \alpha_i - 12 \sum_{i,j=1}^m \alpha_i \alpha_j \ell(i,j) \cdot \text{label}(j) \cdot \text{ai} \cdot \text{aj} \langle x(i), x(j) \rangle]$$

$$F_{\text{Una N}(\text{net} - \theta)} = (e^{\lambda(\text{net} - \theta)} - e^{-\lambda(\text{net} - \theta)}) \min_{\lambda} (\text{net} - \theta) + e^{-\lambda(\text{net} - \theta)} \text{fAN}(\text{net} - \theta) = (e^{\lambda(\text{net} - \theta)} - e^{-\lambda(\text{net} - \theta)}) e^{\lambda(\text{net} - \theta)} + e^{-\lambda(\text{net} - \theta)}$$

Teoría de probabilidad

Aquí hay un ejemplo para probar su conocimiento actual de la teoría de probabilidad: Clasificación con probabilidades condicionales.

$$p(c_{\text{yo}} | x, y) = p(x, y | C_{\text{yo}}) p(C_{\text{yo}}) / p(x, y) p(c_i | x, y) = p(x, y | c_i) p(c_i) / p(x, y)$$

Con estas definiciones, podemos definir la regla de clasificación bayesiana:

- Si $P(c_1 | x, y) > P(c_2 | x, y)$, la clase es c_1 .
- Si $P(c_1 | x, y) < P(c_2 | x, y)$, la clase es c_2 .

Problema de optimización

Aquí hay una función de optimización

$$\max_{\alpha} [\sum_{i=1}^m \alpha_i - 12 \sum_{i,j=1}^m \alpha_i \alpha_j \ell(i,j) \cdot \ell(j) \cdot \text{un}_{\text{yo}} \cdot \text{un}_j \langle x(i), x(j) \rangle] \max_{\alpha} [\sum_{i=1}^m \alpha_i - 12 \sum_{i,j=1}^m \alpha_i \alpha_j \ell(i,j) \cdot \text{label}(j) \cdot \text{ai} \cdot \text{aj} \langle x(i), x(j) \rangle]$$

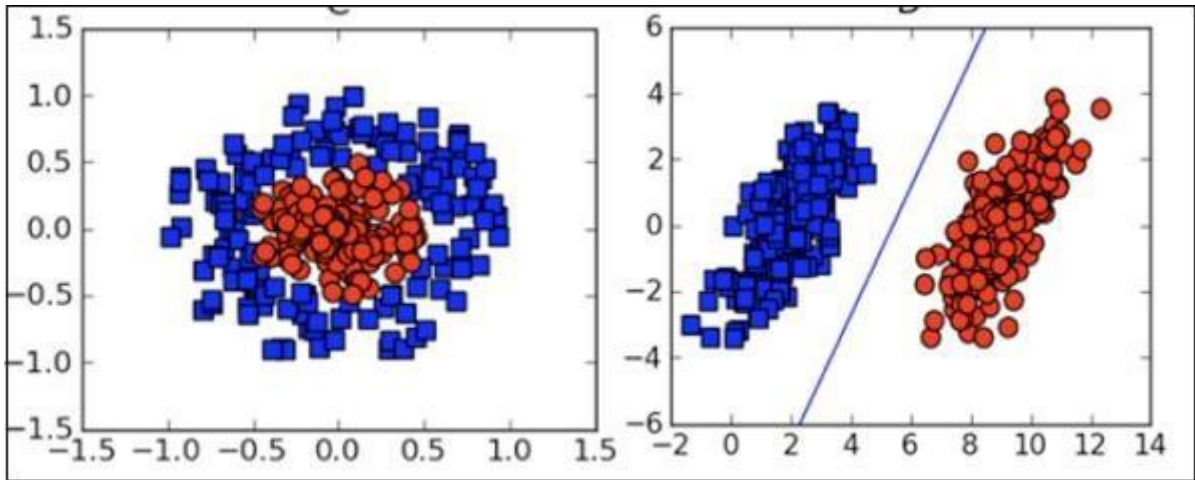
Sujeto a las siguientes restricciones:

$$\alpha_i \geq 0, \text{ and } \sum_{i=1}^m \alpha_i \cdot \text{label}(i) = 0 \quad \alpha_i \geq 0, \text{ and } \sum_{i=1}^m \alpha_i \cdot \text{label}(i) = 0$$

Si puede leer y comprender lo anterior, ya está todo listo.

Visualización

En muchos casos, deberá comprender los diversos tipos de gráficos de visualización para comprender su distribución de datos e interpretar los resultados de la salida del algoritmo.



Además de los aspectos teóricos anteriores del aprendizaje automático, necesita buenas habilidades de programación para codificar esos algoritmos.

Entonces, ¿qué se necesita para implementar ML? Veamos esto en el próximo capítulo.

Aprendizaje automático: implementación

Para desarrollar aplicaciones ML, deberá decidir sobre la plataforma, el IDE y el lenguaje para el desarrollo. Hay varias opciones disponibles. La mayoría de estos cumpliría con sus requisitos fácilmente, ya que todos ellos proporcionan la implementación de algoritmos de IA discutidos hasta ahora.

Si está desarrollando el algoritmo ML por su cuenta, los siguientes aspectos deben entenderse cuidadosamente:

El idioma de su elección: este es esencialmente su dominio en uno de los idiomas admitidos en el desarrollo de ML.

El IDE que utiliza: esto dependerá de su familiaridad con los IDE existentes y su nivel de comodidad.

Plataforma de desarrollo: hay varias plataformas disponibles para el desarrollo y la implementación. La mayoría de estos son de uso gratuito. En algunos casos, es posible que tenga que incurrir en una tarifa de licencia más allá de una cierta cantidad de uso. Aquí hay una breve lista de elección de idiomas, IDEs y plataformas para su referencia.

Elección de idioma

Aquí hay una lista de idiomas que admiten el desarrollo de ML:

- Pitón
- R
- Matlab

- Octava
- Julia
- C ++
- C

Esta lista no es esencialmente completa; sin embargo, cubre muchos lenguajes populares utilizados en el desarrollo del aprendizaje automático. Dependiendo de su nivel de comodidad, seleccione un idioma para el desarrollo, desarrolle sus modelos y pruebe.

IDEs

Aquí hay una lista de IDE que apoyan el desarrollo de ML:

- R Studio
- Pycharm
- Cuaderno iPython / Jupyter
- Julia
- Spyder
- Anaconda
- Rodeo
- Google –Colab

La lista anterior no es esencialmente completa. Cada uno tiene sus propios méritos y deméritos. Se alienta al lector a probar estos IDE diferentes antes de limitarse a uno solo.

Plataformas

Aquí hay una lista de plataformas en las que se pueden implementar aplicaciones ML:

- IBM
- Microsoft Azure
- Google Cloud
- Amazonas
- Mlflow

Una vez más, esta lista no es exhaustiva. Se recomienda al lector que se registre en los servicios mencionados anteriormente y los pruebe por sí mismo.

Aprendizaje automático: conclusión

Este tutorial te ha presentado al aprendizaje automático. Ahora, usted sabe que el aprendizaje automático es una técnica de entrenamiento de máquinas para realizar las actividades que un cerebro humano puede hacer, aunque un poco más rápido y mejor que un ser humano promedio. Hoy hemos visto que las máquinas pueden vencer a los campeones humanos en juegos como el

Ajedrez, AlphaGO, que se consideran muy complejos. Usted ha visto que las máquinas pueden ser entrenadas para realizar actividades humanas en varias áreas y pueden ayudar a los humanos a vivir una vida mejor.

El aprendizaje automático puede ser supervisado o no supervisado. Si tiene una cantidad menor de datos y datos claramente etiquetados para la capacitación, opte por el aprendizaje supervisado. El aprendizaje no supervisado generalmente daría un mejor rendimiento y resultados para grandes conjuntos de datos. Si tiene un gran conjunto de datos fácilmente disponible, busque técnicas de aprendizaje profundo. También has aprendido el aprendizaje por refuerzo y el aprendizaje por refuerzo profundo. Ahora sabe qué son las redes neuronales, sus aplicaciones y limitaciones.

Finalmente, cuando se trata del desarrollo de modelos de aprendizaje automático propios, analizó las opciones de varios lenguajes de desarrollo, IDE y plataformas. Lo siguiente que debe hacer es comenzar a aprender y practicar cada técnica de aprendizaje automático. El tema es amplio, significa que hay ancho, pero si considera la profundidad, cada tema se puede aprender en unas pocas horas. Cada tema es independiente el uno del otro. Debe tener en cuenta un tema a la vez, aprenderlo, practicarlo e implementar el / los algoritmo (s) utilizando el idioma que prefiera. Esta es la mejor manera de comenzar a estudiar Machine Learning. Practicando un tema a la vez, muy pronto adquiriría el ancho que eventualmente se requiere de un experto en Machine Learning.

¡Buena suerte!