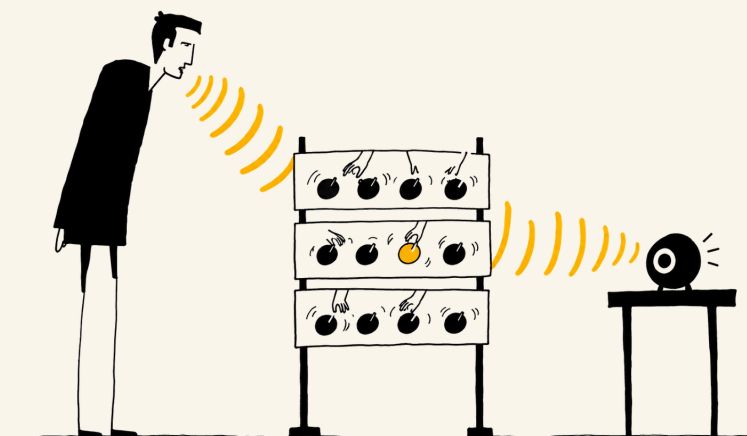


Sumérgete en el aprendizaje profundo



ton Zhang, Zachary C. Lipton, Mu Li, Alexander J. Smola · 20

Audiolecciones ilustradas

Lecturized

Sumérgete en el aprendizaje profundo

Audiolecciones ilustradas

Aston Zhang, Zachary C. Lipton, Mu Li, Alexander J. Smola · 2023

Contado como lecciones ilustradas por Lecturized
lecturized.com

Del original: Dive into Deep Learning (Cambridge University Press, 2023), d2l.ai

Índice

Programas que aprenden de ejemplos

De qué está hecho todo sistema que aprende

Puntuar el modelo, y el examen que debe aprobar

¿Cuánto, o cuál?

Muchas etiquetas, clasificaciones por orden y secuencias

Aprender sin jefe, y un mundo que cambia

Aprender actuando, y dónde empezó todo

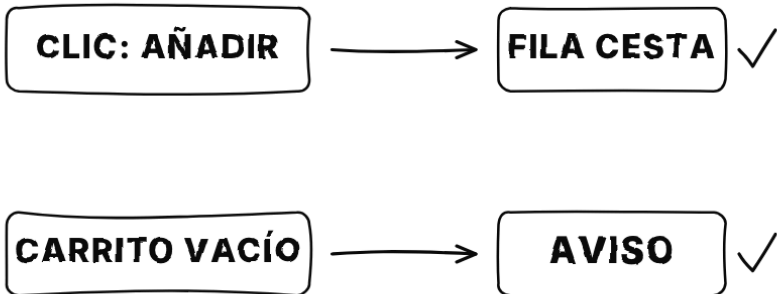
Qué cambió: datos, chips y herramientas

Dónde ya funciona, y qué lo hace profundo

Programas que aprenden de ejemplos

Sostenga el teléfono a poca distancia de la boca y diga las palabras que despiertan a su asistente, «Oye Siri» u «OK Google». Después diga algo parecido pero equivocado, como «Oye Siria» u «OK Gurgle», y observe lo que ocurre. Casi siempre el teléfono despierta con las palabras de verdad y sigue dormido con las que casi aciertan, incluso en una cocina ruidosa e incluso cuando habla otra persona. Ese pequeño truco esconde una pregunta que todo este libro quiere responder: ¿cómo escribió alguien ese programa?

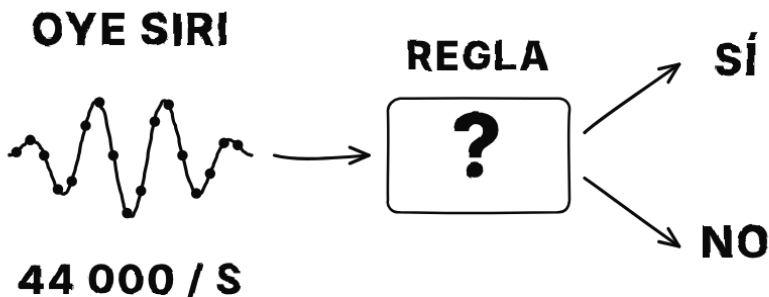
REGLAS



Una tienda funciona con reglas

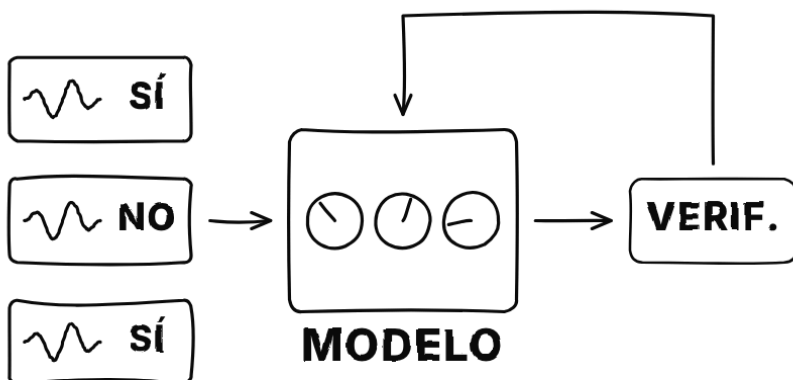
Durante casi toda la historia de la informática, un programa fue una lista rígida de reglas escritas por personas. Piense en una tienda en línea: cuando un cliente añade un artículo al carrito, el programa escribe una fila que une a ese cliente con ese producto, y los desarrolladores piensan en cada caso raro, como alguien que pulsa «comprar» con el carrito vacío. Un programa así puede planearse en una pizarra y lanzarse con confianza antes de que llegue un solo cliente real. Es un logro humano admirable, y cuando las reglas de este tipo funcionan siempre, los autores dicen con claridad que no hace falta aprendizaje automático en absoluto.

Ahora intente escribir las reglas para la palabra de activación. El micrófono registra unos cuarenta y cuatro mil números por segundo, cada uno la altura de la onda sonora en ese instante. ¿Qué regla convierte uno o dos segundos de esos números en un sí o un no seguro? Nadie lo sabe; los autores admiten que tampoco ellos podrían escribirla. El mismo muro se levanta ante muchas tareas: predecir el tiempo de mañana, responder a una pregunta escrita con palabras libres, encontrar a todas las personas de una fotografía o sugerir un producto que le gustaría pero que usted nunca encontraría. A veces la respuesta correcta cambia sin parar con el mundo, y a veces el camino de los píxeles o del sonido a un significado es demasiado enredado para que alguien lo describa, aunque nuestro propio cerebro lo recorre sin esfuerzo. El filtro de spam de su correo se enfrenta a los dos problemas a la vez, porque los spammers cambian sus trucos cada semana.



No hay regla para una voz

Esta es la salida, y es la idea central del capítulo. Usted no puede escribir la regla para reconocer «Oye Siri», pero sí puede reconocerla usted mismo. Así que se reúne un enorme conjunto de grabaciones y se marca cada una: esta contiene la palabra de activación, esta no. Luego, en lugar de un programa fijo, se construye uno flexible cuyo comportamiento depende de muchísimos números ajustables llamados parámetros, que puede imaginar como perillas. Con las perillas puestas al azar, el programa dice sí y no por casualidad. Después, un algoritmo de aprendizaje gira las perillas poco a poco para que las respuestas del programa sobre las grabaciones marcadas mejoren, y sigue hasta que el programa es bueno. Una vez fijadas las perillas, el programa se llama modelo.



Girar las perillas

Los autores llaman a esto programar con datos. Usted no le dice a la máquina cómo oír la palabra de activación; le muestra muchos ejemplos y deja que las perillas encuentren la posición que funciona. El mismo tipo de modelo, con otra posición de las perillas, podría aprender otra palabra igual de bien, pero un trabajo muy distinto, como escribir pies de foto o traducir del inglés al chino, puede necesitar un tipo de modelo completamente diferente. El asistente de voz de su teléfono, el tiempo de viaje previsto en su aplicación de mapas y el feed que adivina qué vídeo verá a continuación se hicieron todos así, a partir de ejemplos y no de reglas escritas a mano.

Esto es lo que el libro llama aprendizaje automático: programas cuyo rendimiento mejora con la experiencia, por lo general en forma de

datos. Las reglas de la tienda en línea nunca mejoran solas, por muchos clientes que pasen; un programa que aprende, sí. El aprendizaje profundo, el tema del libro, es una familia de métodos dentro del aprendizaje automático, y hoy impulsa gran parte del progreso en el reconocimiento de imágenes y de voz, en la comprensión del lenguaje, en la medicina y en la lectura de genomas. El chatbot al que usted escribe preguntas es un modelo de aprendizaje profundo cuyas perillas se ajustaron mostrándole una cantidad enorme de texto. Esta parte ha dejado fuera los detalles de cómo se giran las perillas y cómo se mide la calidad de un modelo; eso viene después.

Pruébelo

Pida a un amigo que diga en voz baja la palabra de activación de su teléfono desde el otro lado de la habitación, y luego una palabra parecida con la misma voz. Cuente cuántas veces despierta el teléfono con cada una. Verá que normalmente despierta con la palabra real y rara vez con la parecida, aunque no a la perfección: un modelo aprendido acierta la mayoría de las veces, no siempre, porque aprendió de ejemplos y no de una regla.

Para recordar

Algunas tareas pueden resolverse con reglas escritas por personas, y entonces no hace falta el aprendizaje automático. Otras tareas, como oír una palabra de activación o reconocer una cara, nos resultan fáciles pero es imposible ponerlas en reglas. Para ellas reunimos muchos ejemplos marcados y dejamos que un algoritmo de aprendizaje ajuste las perillas de un programa flexible hasta que haga bien el trabajo. El programa ajustado se llama modelo, y esta manera de trabajar es programar con datos. El aprendizaje profundo es una familia potente de estos métodos, y es lo que enseña el resto del libro.

Preguntas

1. ¿Por qué es difícil reconocer con reglas escritas a mano una palabra de activación como «Oye Siri»?
 - a. El micrófono recoge demasiados pocos números para distinguir palabras
 - b. Nadie sabe decir qué regla convierte decenas de miles de mediciones del sonido en sí o no, aunque las personas oyen la palabra con facilidad
 - c. Los teléfonos no son lo bastante rápidos para seguir una larga lista de reglas
 - d. La palabra de activación cambia cada vez que se actualiza el teléfono

2. ¿Cuándo dice el libro que no hace falta en absoluto el aprendizaje automático?
 - a. Cuando la tarea implica una gran base de datos de clientes
 - b. Cuando el programa tiene que funcionar en un teléfono
 - c. Cuando se pueden escribir reglas que funcionan siempre, como para el carrito de una tienda en línea
 - d. Cuando los datos se registran como números y no como palabras
3. Un algoritmo de aprendizaje acaba de empezar con un programa nuevo cuyas perillas están puestas al azar. ¿Qué hace el programa?
 - a. Responde sí y no más o menos por casualidad
 - b. Ya reconoce la palabra de activación, pero despacio
 - c. Se niega a responder hasta haber visto todos los datos
 - d. Reconoce igual de bien cualquier palabra inglesa
4. ¿Qué diferencia a un programa que aprende del programa de la tienda en línea con el paso del tiempo?
 - a. El programa de la tienda es más rápido porque sus reglas son fijas
 - b. El programa que aprende mejora con más experiencia, mientras que las reglas de la tienda siguen iguales hasta que los desarrolladores las reescriben
 - c. El programa que aprende no necesita datos una vez lanzado
 - d. El programa de la tienda aprende de los clics de cada cliente

Respuestas

1. **b)** El teléfono registra unos cuarenta y cuatro mil números por segundo, y nadie conoce una regla que los convierta de forma fiable en sí o no, aunque nuestros oídos lo hacen sin esfuerzo.
2. **c)** Si las reglas escritas por personas funcionan en todos los casos, como pueden hacerlo para el carrito de la tienda, el aprendizaje no tiene nada que añadir.
3. **a)** Con perillas al azar el programa no sirve; solo girarlas para que acierte más en los ejemplos marcados lo vuelve bueno.
4. **b)** Aprendizaje automático significa un rendimiento que mejora con la experiencia; las reglas de la tienda nunca mejoran solas, por muchos clientes que pasen.

Ejercicios

1. El libro enumera tareas para las que incluso los mejores programadores tendrían dificultades para escribir reglas, como predecir el tiempo de mañana o sugerir productos que le gustarían. Elija dos y explique, para cada una, cuál de las dos razones se aplica: la respuesta correcta cambia sin parar con el mundo, o el camino de la entrada al significado es demasiado enredado para describirlo.

Pista: Pregúntese para cada tarea: ¿una regla perfecta escrita hoy seguiría siendo correcta el año que viene? ¿Y podría describir paso a paso cómo lo hace en su cabeza?

Respuesta: Las sugerencias de productos entran en la primera razón: los gustos de la gente y la oferta cambian, así que cualquier regla fija se queda vieja y la solución debe adaptarse. Encontrar a todas las personas de una fotografía entra en la segunda: lo hacemos sin esfuerzo, pero los pasos de los píxeles a «esto es una persona» están más allá de lo que nadie puede escribir. La predicción del tiempo tiene un poco de ambas.

2. El libro describe cómo se podría «programar» un detector de gatos sin escribir ninguna regla sobre bigotes u orejas. Explique con sus palabras qué le daría al sistema y qué aprendería a devolver.

Pista: Piense en la receta de la palabra de activación: ejemplos marcados de entrada, perillas ajustadas, un modelo de salida.

Respuesta: Le da muchas imágenes marcadas como gato o perro. Un algoritmo de aprendizaje ajusta sus perillas hasta que devuelve un número positivo grande para un gato, uno negativo grande para un perro y algo cercano a cero cuando no está seguro. El comportamiento sale de los ejemplos, que es lo que el libro llama programar con datos.

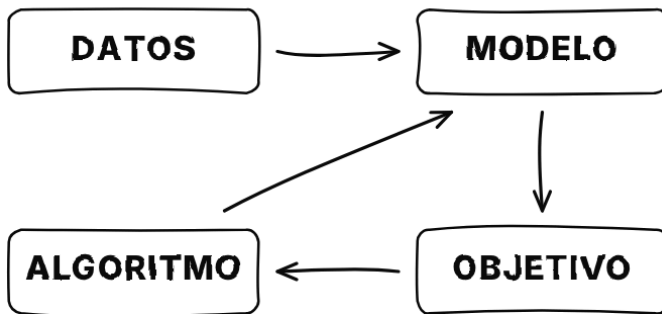
3. Se ha entrenado un modelo para despertar con «Alexa». Ahora sus creadores quieren que despierte con «Apricot» y, por separado, quieren un programa que escriba pies de foto. ¿Para cuál de los dos pueden probablemente conservar el mismo tipo de modelo, y por qué?

Pista: Compare lo que entra y lo que sale en cada trabajo.

Respuesta: Para «Apricot»: es el mismo tipo de trabajo, audio de entrada y sí o no de salida, así que la misma familia de modelos con otra posición de las perillas debería servir. Escribir pies de foto toma imágenes y devuelve frases, un trabajo fundamentalmente distinto que puede necesitar un tipo de modelo completamente diferente.

De qué está hecho todo sistema que aprende

Tome cualquier fotografía de su teléfono y ampliela hasta que se deshaga en cuadraditos. Cada cuadradito, un píxel, no es más que tres números: cuánto brillan la luz roja, la verde y la azul en ese punto. Una fotografía pequeña de doscientos píxeles de ancho y doscientos de alto es, por tanto, una lista de ciento veinte mil números. Eso es lo que de verdad ve un programa que aprende, y es la primera de las cuatro partes que tiene todo sistema de aprendizaje automático, sea cual sea su trabajo.



Una foto son números

Las cuatro partes son estas: los datos de los que aprende, un modelo que transforma los datos, una función objetivo que puntúa lo bien que lo hace el modelo, y un algoritmo que ajusta las perillas del modelo para mejorar esa puntuación. El ejemplo de la palabra de activación de la primera parte tenía las cuatro, y pertenecía al tipo de problema más común, el aprendizaje supervisado, en el que cada ejemplo viene con la respuesta que el programa debe aprender a dar.

Los datos son una colección de ejemplos, cada uno convertido en números. Los números que describen un ejemplo son sus características, y en el aprendizaje supervisado la respuesta que hay que predecir es su etiqueta. En un hospital, las características de un paciente podrían ser la edad, el pulso, los medicamentos y las operaciones recientes, y la etiqueta, si ese paciente sobrevivió los treinta días siguientes. Cuando todos los ejemplos tienen el mismo número de características, los datos son ordenados; pero las fotografías de internet vienen en todas las formas, y el texto es peor, porque una reseña dice solo «¡apesta!» mientras otra ocupa páginas. Manejar con soltura datos de longitud variable es una de las grandes ventajas del aprendizaje profundo sobre los métodos antiguos. Los mensajes que lee un filtro de spam son justo este tipo de datos: unos ocupan una línea, otros son cartas largas.

EDAD	PULSO	FÁRMACOS	¿SOBREVIVIÓ?
64	72	3	SÍ
51	88	1	NO

RASGOS **ETIQUETA**

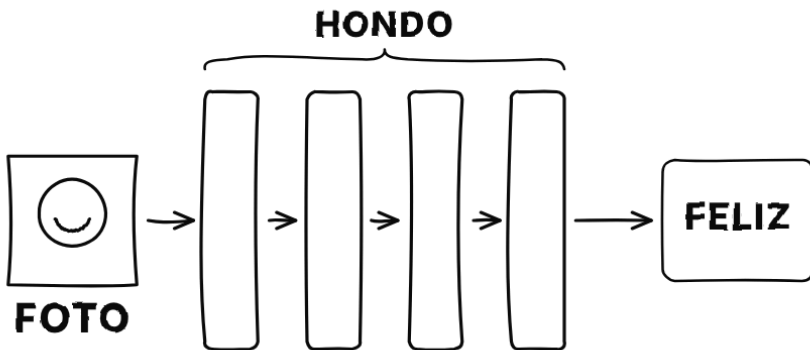
☐ ¡APESTA!

RESEÑA LARGA
BASURA → BASURA

Características y etiqueta

Hay dos cosas más sobre los datos que importan muchísimo. Primero, más es mejor: con más ejemplos se pueden entrenar modelos más potentes, y el paso de pocos datos a muchos es una de las grandes razones del éxito del aprendizaje profundo. Segundo, los datos deben ser los adecuados, porque si entra basura, sale basura. Si falta un grupo de personas entre los ejemplos, como en un detector de cáncer de piel que nunca vio piel oscura, el sistema les falla; y si las decisiones pasadas fueron injustas, como viejas decisiones de contratación usadas para filtrar solicitudes nuevas, el modelo puede aprender y repetir esa injusticia en silencio, sin que nadie lo pretenda. Un feed de recomendaciones entrenado solo con lo que vio un grupo de usuarios servirá mal a todos los demás.

El modelo es la maquinaria que recibe datos de un tipo y entrega predicciones, quizá de otro tipo: entran fotografías, sale una puntuación de sonrisa; entran lecturas de sensores, sale un aviso de algo anómalo. Los modelos sencillos sirven para problemas sencillos, pero los problemas de este libro los superan. El aprendizaje profundo se distingue porque sus modelos son cadenas de muchas transformaciones, una tras otra, cada una pasando su resultado a la siguiente, y esa profundidad es lo que da nombre al aprendizaje profundo. Cómo funcionan la función objetivo y el algoritmo de aprendizaje queda para la próxima parte.



Una cadena de transformaciones

Para recordar

Todo sistema de aprendizaje automático tiene cuatro partes: datos, un modelo, una función objetivo y un algoritmo que mejora el modelo. Los datos son ejemplos convertidos en números, descritos por características, y en el aprendizaje supervisado cada uno lleva una etiqueta. Más datos ayudan, pero los datos equivocados o sesgados llevan al fracaso y pueden repetir viejas injusticias. Un modelo profundo es una larga cadena de transformaciones de los datos, y esa cadena es lo que lo hace profundo.

Preguntas

1. Un equipo construye para un hospital un modelo que predice si un paciente sobrevivirá los próximos treinta días. ¿Cuál de estas es la etiqueta?
 - a. La edad y el pulso del paciente
 - b. Si el paciente sobrevivió los treinta días
 - c. La lista de medicamentos del paciente
 - d. El número de pacientes en los registros
2. ¿Por qué el texto da más problemas a los sistemas que aprenden que las imágenes de un único microscopio estándar?
 - a. El texto no contiene ningún número
 - b. Las reseñas y los mensajes varían muchísimo de longitud, así que no encajan en un número fijo de características

- c. El texto siempre está lleno de errores
 - d. Las imágenes de microscopio no necesitan características
3. Un modelo para filtrar currículums se entrena con las decisiones de contratación pasadas de una empresa, que favorecerían a un grupo. ¿Cuál es el peligro?
- a. El modelo será demasiado lento para usarlo
 - b. El modelo se negará a clasificar a nadie
 - c. El modelo puede aprender y repetir la vieja injusticia, aunque nadie lo pretenda
 - d. El modelo necesitará menos ejemplos de lo habitual
4. ¿Qué hace que un modelo de aprendizaje profundo sea «profundo»?
- a. Es una larga cadena de transformaciones, cada una pasando su resultado a la siguiente
 - b. Se entrena con cantidades muy grandes de datos
 - c. Tiene la función objetivo más precisa
 - d. Solo funciona con fotografías

Respuestas

1. **b)** La etiqueta es la respuesta que el modelo aprende a predecir; la edad, el pulso y los medicamentos son características que entran.
2. **b)** Los datos ordenados tienen el mismo número de características en cada ejemplo; una reseña dice «¡apesta!» y otra ocupa páginas, y la fuerza del aprendizaje profundo está en manejar esa longitud variable.
3. **c)** Si entra basura, sale basura: si los ejemplos reflejan prejuicios pasados, el modelo los aprende en silencio.

4. a) La profundidad es el número de transformaciones sucesivas encadenadas, y de ahí viene el nombre.

Ejemplos resueltos

Una fotografía como números Una fotografía en color mide 200 píxeles de ancho y 200 de alto. Cada píxel se guarda como el brillo de la luz roja, verde y azul. ¿Cuántos números recibe un programa que aprende por esta única fotografía?

Cuenta los píxeles: 200 de ancho por 200 de alto da 40 000 píxeles.

Cada píxel lleva tres números, uno por cada color de luz.

Así que la fotografía son 40 000 por 3 números.

Respuesta: 120 000 números

Ejercicios

1. Una fotografía en color mide 300 píxeles de ancho y 200 de alto, guardada del mismo modo como brillo de rojo, verde y azul para cada píxel. ¿Cuántas características le da al modelo?

Pista: Primero cuente los píxeles; luego recuerde cuántos números lleva cada píxel.

Respuesta: 180 000. Hay 300 por 200, es decir 60 000 píxeles, y cada uno da tres números.

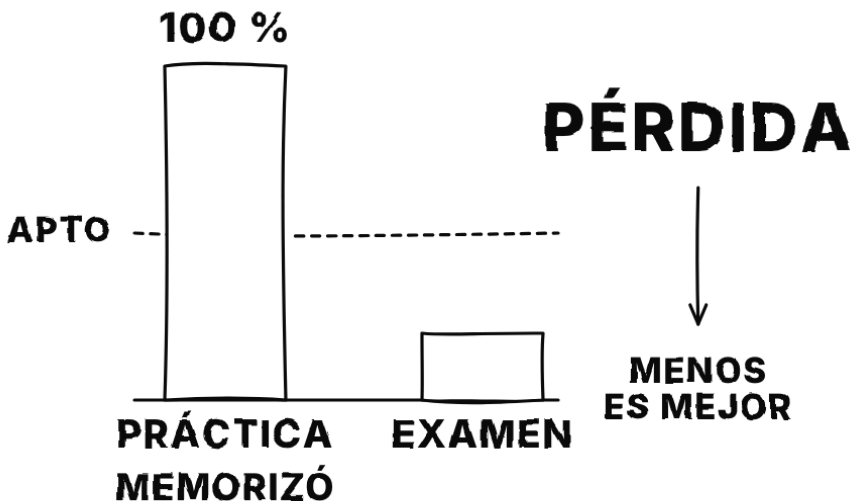
2. Un detector de cáncer de piel funciona bien en las pruebas, pero falla con muchos pacientes cuando se usa en una clínica. Con la idea de «los datos adecuados», dé una razón probable y diga qué habría que cambiar.

Pista: Pregúntese quién estaba, y quién no, entre los ejemplos de los que aprendió el detector.

Respuesta: Probablemente los ejemplos de entrenamiento dejaron fuera a algún grupo de pacientes, por ejemplo a personas de piel oscura, así que el detector nunca aprendió cómo es su piel. Más datos del mismo tipo no ayudarían; los ejemplos deben incluir a las personas con las que se usará el sistema.

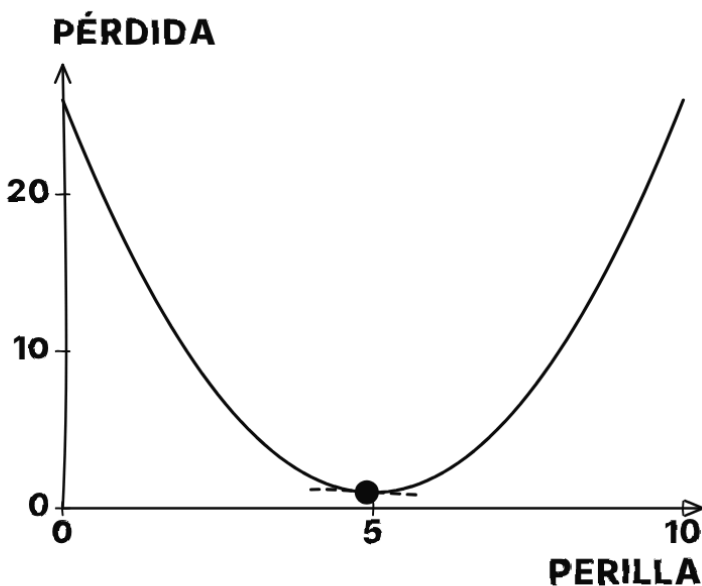
Puntuar el modelo, y el examen que debe aprobar

Imagine a un estudiante que prepara un examen final con un montón de exámenes de práctica. Al cabo de una semana, saca la nota máxima en todos los exámenes de práctica. Luego llega el examen de verdad con preguntas que nunca ha visto, y suspende. No había aprendido la materia; había memorizado las respuestas de práctica. Las máquinas que aprenden caen exactamente en la misma trampa, y la forma en que el libro se protege de ella es el centro de esta parte.



La práctica no es el examen

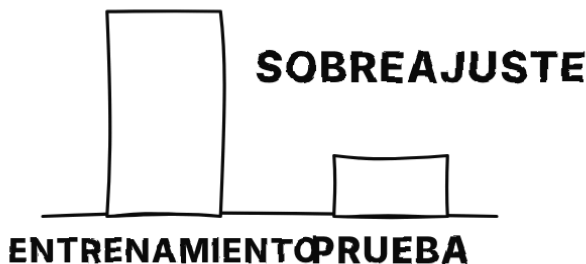
Primero, una máquina no puede mejorar si nadie dice qué significa mejorar. Por eso todo sistema que aprende tiene una función objetivo, un único número que puntúa lo mal que lo hace el modelo, y por convención menos es mejor, por lo que también se llama pérdida. Cuando el modelo predice un número, la pérdida habitual castiga el tamaño de cada fallo, y los fallos grandes cuentan mucho más que los pequeños. Cuando predice una categoría, la puntuación natural es la proporción de ejemplos en que se equivoca, aunque esa puntuación es incómoda de mejorar directamente, así que en la práctica se usa un sustituto más suave. Para un filtro de spam, la pérdida podría ser simplemente cuántos mensajes puso en la carpeta equivocada.



Bajar la colina

Después viene el algoritmo de aprendizaje, la parte que gira las perillas. El método detrás de casi todo el aprendizaje profundo se llama descenso de gradiente. En cada paso pregunta, para cada perilla, si moverla un poquito haría subir o bajar la pérdida, y luego mueve cada perilla un poco en la dirección que la baja. Paso a paso, como quien baja una colina en la niebla tanteando la pendiente con los pies, el modelo mejora con sus ejemplos.

Y ahora, la trampa. Hacerlo bien con los ejemplos usados para entrenar no demuestra que el modelo lo hará bien con otros nuevos. Por eso los datos se dividen en dos: un conjunto de entrenamiento, con el que se ajustan las perillas, y un conjunto de prueba, guardado aparte y mostrado al modelo solo al final, como el examen de verdad. Un modelo que brilla en el conjunto de entrenamiento y falla en el de prueba está sobreajustado: ha memorizado en lugar de aprender.



Conjunto de entrenamiento y de prueba

Esto es el aprendizaje supervisado, el tipo que está detrás de la mayoría de los éxitos del aprendizaje automático en la industria. Reúna ejemplos, consiga la respuesta correcta para cada uno, ya sea de registros que ya existen o de personas que los marcan a mano, déselos al algoritmo de aprendizaje, y sale un modelo. Luego dele al modelo entradas que nunca ha visto y use sus salidas como predicciones: cáncer o no a partir de una exploración, una frase en francés a partir de una en inglés, el precio de una acción a partir del informe de una empresa. Cuando usted marca un mensaje como spam, está añadiendo un ejemplo etiquetado más al conjunto de entrenamiento de alguien. Cómo cambian el problema los distintos tipos de respuesta, números o categorías, queda para la próxima parte.

Para recordar

Un modelo solo puede mejorar frente a una puntuación, la pérdida, que dice lo mal que lo hace; menos es mejor. El descenso de gradiente mejora el modelo moviendo cada perilla un poco en la dirección que baja la pérdida. Las buenas notas en los ejemplos de entrenamiento demuestran poco, así que se guarda aparte un conjunto de prueba, como un examen de verdad. Un modelo que va bien en el entrenamiento y mal con datos nuevos está sobreajustado. Aprendizaje supervisado significa aprender de ejemplos con respuestas conocidas y luego predecir respuestas para entradas nuevas.

Preguntas

1. Un modelo saca una nota casi perfecta en su conjunto de entrenamiento pero mala en el de prueba. ¿Qué ha pasado?
 - a. El conjunto de prueba era demasiado pequeño para contar
 - b. El modelo se ha sobreajustado: memorizó sus ejemplos en lugar de aprender
 - c. La pérdida se definió de modo que más es mejor
 - d. El descenso de gradiente movió las perillas en la dirección equivocada
2. ¿Por qué se mantiene el conjunto de prueba lejos del modelo hasta el final?

- a. Para que muestre cómo le va al modelo con datos que nunca ha visto
 - b. Porque el conjunto de prueba no tiene etiquetas
 - c. Porque el descenso de gradiente no puede trabajar con más datos
 - d. Para que el entrenamiento termine antes
3. ¿Qué hace el descenso de gradiente en cada paso?
- a. Prueba posiciones al azar de todas las perillas y se queda con la mejor
 - b. Añade más ejemplos al conjunto de entrenamiento
 - c. Comprueba cómo cambiaría la pérdida un pequeño giro de cada perilla y mueve las perillas en el sentido que la baja
 - d. Elimina los ejemplos en los que el modelo se equivoca
4. Un modelo predice precios de casas. Con la pérdida habitual para números, ¿qué se castiga más?
- a. Diez fallos pequeños del mismo tamaño total
 - b. Ambos por igual, porque solo importa el total
 - c. Ninguno, porque los precios son categorías
 - d. Un fallo grande

Respuestas

- 1. b)** Brillar con los ejemplos de entrenamiento y fallar con los nuevos es justo el sobreajuste, como el estudiante que memorizó los exámenes de práctica.
- 2. a)** El conjunto de prueba hace el papel del examen real; si el modelo se entrenara con él, ya no se podría distinguir aprender de memorizar.

3. c) Como al bajar una colina en la niebla, tantea la pendiente de cada perilla y da un pequeño paso hacia abajo.

4. d) La pérdida habitual para números hace que los fallos grandes cuenten mucho más que los pequeños.

Ejemplos resueltos

Tasa de error de un filtro de spam Un filtro de spam se prueba con 500 mensajes que nunca vio en el entrenamiento y pone 15 de ellos en la carpeta equivocada. ¿Cuál es su tasa de error en el conjunto de prueba?

La tasa de error es la proporción de ejemplos en los que la predicción no coincide con la respuesta verdadera.

Aquí fallaron 15 de 500 mensajes.

15 dividido entre 500 es 0,03.

Respuesta: 0,03, es decir, el 3 por ciento

Ejercicios

1. Con el error cuadrático, una predicción que falla por 2 cuesta 2 por 2, es decir 4. ¿Cuánto cuesta un fallo de 6 y cuántas veces más es que un fallo de 2?

Pista: Eleve al cuadrado la diferencia entre la predicción y el valor verdadero.

Respuesta: 36, que es 9 veces 4: triplicar el fallo multiplica el coste por nueve, y por eso los fallos grandes cuentan mucho más.

2. Se entrenan dos modelos para reconocer palabras de activación. El modelo A acierta el 99 por ciento en el conjunto de entrenamiento y el 70 por ciento en el de prueba; el modelo B, el 92 por ciento en

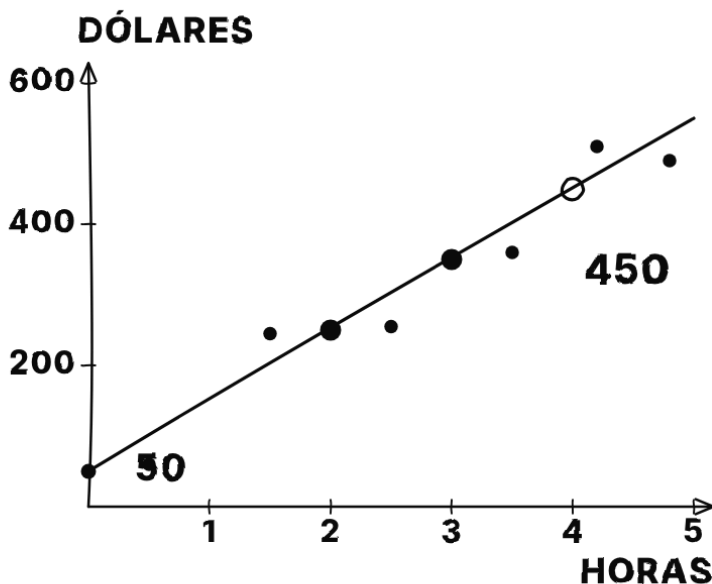
entrenamiento y el 91 en prueba. ¿Cuál pondría en un teléfono y por qué?

Pista: ¿Qué número le dice cómo le irá al modelo con voces que nunca ha oído?

Respuesta: El modelo B. El conjunto de prueba representa los datos nuevos, y ahí B va mucho mejor; la nota alta de A en entrenamiento con una nota mucho más baja en prueba es señal de sobreajuste.

¿Cuánto, o cuál?

Imagine que un fontanero desatasca sus tuberías durante tres horas y le envía una factura de trescientos cincuenta dólares. Un amigo suyo llama al mismo fontanero, que trabaja dos horas, y la factura es de doscientos cincuenta. Un vecino pregunta cuánto costaría un trabajo de cuatro horas. La mayoría lo calcula sin bolígrafo: una hora más añadió cien dólares, así que el fontanero cobra cien por hora más cincuenta dólares por la visita, y cuatro horas deberían salir por cuatrocientos cincuenta. Si lo ha seguido, dice el libro, ya entiende la idea que hay detrás del tipo más sencillo de aprendizaje supervisado.



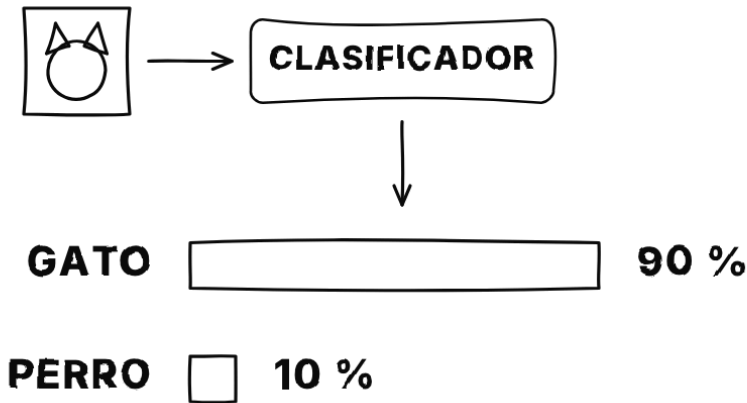
La recta de precios del fontanero

Ese tipo se llama regresión, y lo que hace que un problema sea de regresión es la respuesta que pide: un número que puede tomar cualquier valor. Los ejemplos suelen ser filas de una tabla, y las características de cada fila forman un vector de características. Para casas, la fila podría tener la superficie, los dormitorios, los baños y los minutos a pie hasta el centro, y la etiqueta es el precio por el que se vendió la casa. Una buena regla práctica es que toda pregunta de «¿cuánto?» o «¿cuántos?» es regresión: cuántas horas durará una operación, cuánta lluvia caerá en las próximas seis horas, cuántos días pasará un paciente en el hospital. La puntuación en estrellas que un servicio de streaming adivina que usted daría a una película también es una regresión; el libro recuerda que predecir bien esas puntuaciones en 2009 podía ganar el premio de un millón de dólares de Netflix.

Con el fontanero, dos ejemplos fijaron el precio con exactitud. Los datos reales nunca son tan limpios: otras cosas que las características no recogen empujan las respuestas de un lado a otro. Así que al modelo no se le pide acertar cada ejemplo, solo acercarse lo más posible en conjunto, y la medida habitual de cercanía es el fallo al cuadrado de la parte anterior.

La otra gran familia es la clasificación, y su pregunta es «¿cuál?». La aplicación de un banco fotografía un cheque y debe decidir, para cada trazo escrito a mano, qué carácter es; un lector de dígitos elige entre diez clases, del cero al nueve; el caso más sencillo tiene solo dos clases, como gato o perro. Las respuestas firmes de sí o no resultan difíciles de entrenar, así que un clasificador suele dar probabilidades: ante una foto, puede decir que está seguro al noventa por ciento de que el animal es un gato. Ese número también dice lo inseguro que está el modelo, lo que importa mucho para lo que pasa después. Su

filtro de spam es un clasificador con dos clases, spam y no spam, y también tiene en cuenta lo seguro que está antes de mover un mensaje.



Un clasificador da probabilidades

El libro lo explica con una seta encontrada en un jardín. Un clasificador entrenado con fotografías dice que hay un veinte por ciento de probabilidad de que sea una oronja verde, la seta más mortal que existe. Así que está seguro al ochenta por ciento de que la seta es inofensiva, y aun así sería una locura comérsela. La respuesta más probable no siempre es la que hay que seguir: una decisión debe sopesar las probabilidades frente a lo que cuesta cada resultado, y una pequeña probabilidad de morir pesa más que cualquier cena. La seta era, de hecho, una oronja verde.

ORONJA 20 %

SEGURA 80 %



COMER

TIRAR

RIESGO: MUERTE

COSTE: CENA



Sopesar probabilidad y coste

No todos los errores son iguales también en otro sentido. Cuando las clases forman un árbol genealógico, como el árbol de los seres vivos de Linneo, confundir un caniche con un schnauzer es un pequeño desliz, mientras que confundir un caniche con un dinosaurio es absurdo, así que se puede construir un modelo que prefiera los fallos cercanos. Pero qué árbol importa depende del uso: una serpiente de cascabel y una inofensiva culebra rayada son parientes cercanas, y aun así confundirlas puede ser mortal. Las predicciones con muchas etiquetas y los tipos más complejos quedan para la próxima parte.

Pruébalo

Anote las dos facturas del fontanero: tres horas por trescientos cincuenta dólares y dos horas por doscientos cincuenta. Halle el precio por hora a partir de la diferencia, luego la tarifa fija, y prediga un trabajo de cinco horas. Debería obtener cien por hora, cincuenta por la visita y quinientos cincuenta dólares: acaba de ajustar un modelo a unos datos y usarlo para predecir.

Para recordar

La regresión responde a «¿cuánto?» o «¿cuántos?» con un número, como un precio o un número de horas. La clasificación responde a «¿cuál?» eligiendo entre categorías, como gato o perro, o los dígitos del cero al nueve. Los clasificadores suelen dar probabilidades, que muestran lo seguros que están. Una decisión debe sopesar esas probabilidades frente al coste de cada resultado, así que una seta inofensiva al ochenta por ciento acaba igualmente en la basura. Unos errores son peores que otros, y cuáles depende de cómo se vaya a usar el modelo.

Preguntas

1. ¿Cuál de estas preguntas es un problema de regresión?
 - a. ¿Este correo es spam o no?
 - b. ¿Qué dígito está escrito en este cheque?

- c. ¿Cuántas horas durará esta operación?
 - d. ¿Esta seta es una oronja verde?
2. Un clasificador está seguro al 80 por ciento de que una seta es inofensiva. ¿Por qué aun así no debería comerla?
- a. Porque el 80 por ciento es la respuesta menos probable
 - b. Porque la decisión debe sopesar el 20 por ciento de probabilidad frente al coste, y ese coste es la muerte
 - c. Porque los clasificadores nunca dan probabilidades
 - d. Porque el clasificador se entrenó con gatos y perros
3. ¿Por qué los clasificadores suelen devolver probabilidades en lugar de una respuesta firme?
- a. Las respuestas firmes son difíciles de entrenar, y una probabilidad además muestra lo inseguro que está el modelo
 - b. Las probabilidades necesitan menos datos
 - c. Las respuestas firmes no se pueden mostrar en pantalla
 - d. Las probabilidades hacen que todos los errores cuenten lo mismo
4. En un modelo que distingue serpientes para excursionistas, ¿qué error importa más?
- a. Llamar culebra de collar a una culebra rayada
 - b. Llamar a una serpiente inofensiva con el nombre de otra inofensiva
 - c. Llamar culebra rayada a una serpiente de cascabel
 - d. Llamar serpiente de cascabel a una culebra rayada

Respuestas

1. c) Una pregunta de «¿cuánto?» o «¿cuántos?» pide un número, y eso la hace de regresión; las demás eligen entre categorías.

2. b) La respuesta más probable no siempre es la que hay que seguir; una pequeña probabilidad de un resultado mortal pesa más que el beneficio de una cena.

3. a) Las salidas firmes de sí o no son difíciles de optimizar, y una probabilidad como «noventa por ciento gato» lleva la incertidumbre del modelo, que las decisiones necesitan.

4. c) Las dos serpientes son parientes cercanas, pero tomar una peligrosa serpiente de cascabel por una inofensiva culebra rayada puede ser mortal; qué errores importan depende del uso.

Ejemplos resueltos

Los precios del fontanero Un operario trabaja 3 horas y cobra 350 dólares; para otro cliente trabaja 2 horas y cobra 250 dólares.

Suponga una tarifa fija por visita más un precio por hora. Halle ambos y prediga la factura de 4 horas.

Una hora más, de 2 a 3, añadió 350 menos 250, es decir 100 dólares, así que el precio por hora es de 100 dólares.

Dos horas cuestan 2 por 100, es decir 200 dólares de trabajo, así que el resto de la factura de 250 dólares, 50 dólares, es la tarifa fija.

Comprobación con la otra factura: 50 más 3 por 100 son 350, como se cobró.

Para 4 horas: 50 más 4 por 100.

Respuesta: 100 dólares por hora más una tarifa de 50 dólares; 450 dólares por 4 horas

Comer o tirar Un clasificador da una probabilidad de 0,2 de que una seta sea una oronja verde y de 0,8 de que sea inofensiva. Tirar

una seta inofensiva cuesta 1 (una cena perdida); tirar una oronja verde no cuesta nada; comer una oronja verde cuesta una vida, que ninguna cena puede compensar. ¿Cuál es el coste esperado de tirarla, y qué debe hacer?

Tirlarla: $0,2$ por 0 por la oronja verde más $0,8$ por 1 por la cena perdida.

Eso da $0,8$.

Comerla: una probabilidad de $0,2$ de un coste sin límite, así que su coste esperado tampoco tiene límite.

Un coste finito es mejor que uno ilimitado, así que hay que tirarla.

Respuesta: Tirarla cuesta $0,8$ de media; comerla cuesta sin límite, así que se tira.

Ejercicios

1. Un limpiacristales cobra 380 dólares por un trabajo de 4 horas y 140 dólares por uno de 1 hora, con una tarifa fija más un precio por hora. Halle el precio por hora, la tarifa y la factura de 3 horas.

Pista: ¿Cuánto subió la factura con tres horas más?

Respuesta: Las 3 horas extra añadieron 240 dólares, así que 80 dólares por hora; entonces 140 menos 80 deja una tarifa de 60 dólares; 3 horas cuestan 60 más 240, es decir 300 dólares.

2. Clasifique estos casos en regresión y clasificación, y diga por qué: el precio por el que se venderá un piso; si una exploración muestra cáncer; cuánta lluvia caerá en las próximas seis horas; qué letra es un trazo escrito a mano.

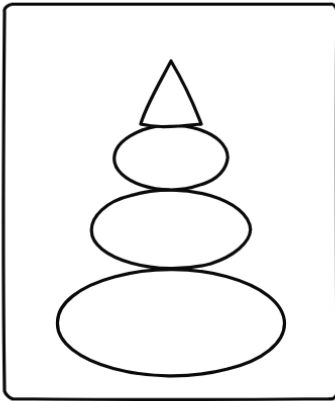
Pista: Pregúntese si la respuesta es un número o una elección entre categorías.

Respuesta: El precio del piso y la cantidad de lluvia son números, así que son regresión; cáncer o no y qué letra son elecciones entre categorías,

así que son clasificación.

Muchas etiquetas, clasificaciones por orden y secuencias

Abra la aplicación de fotos de su teléfono y escriba «perro» en el buscador. Aparecen sus fotos del perro, pero también aquella del parque en la que un perro, un niño, una bicicleta y un árbol comparten el encuadre. Esa fotografía no puede pertenecer a una sola clase. El teléfono la ha etiquetado con varias etiquetas a la vez, y ese pequeño hecho nos lleva más allá de la pregunta «¿cuál?» de la parte anterior.



UNA CLASE



BURRO



PERRO



GATO



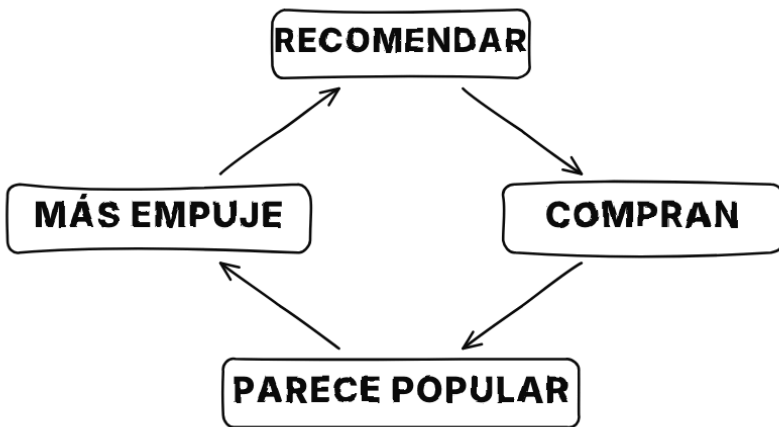
GALLO



Una foto, muchas etiquetas

La imagen del libro es el cuento de los músicos de Bremen: un burro, un perro, un gato y un gallo subidos unos sobre otros. Un clasificador obligado a elegir un solo animal fallará sin remedio, porque la respuesta honrada es los cuatro. Predecir clases que no se excluyen se llama clasificación multitiqueta, y así funciona el etiquetado automático. Una entrada de un blog de tecnología puede llevar de cinco a diez etiquetas, y las etiquetas van juntas: las entradas sobre computación en la nube suelen mencionar a algún gran proveedor de nube, y las de aprendizaje automático, los chips gráficos. En el extremo, la Biblioteca Nacional de Medicina de Estados Unidos etiqueta cada artículo médico indexado con términos elegidos entre unos veintiocho mil encabezados, un trabajo que a los expertos humanos les lleva hasta un año, así que un modelo aprendido puede poner etiquetas provisionales mientras tanto.

La búsqueda plantea otra pregunta distinta: no si una página es pertinente, sino en qué orden mostrar las páginas pertinentes. El ingrediente secreto original de Google, PageRank, daba a cada página una puntuación de autoridad que, curiosamente, no dependía en absoluto de la consulta; un filtro sencillo encontraba primero las páginas pertinentes, y la puntuación las ponía después en orden. Hoy los buscadores aprenden puntuaciones que dependen de la consulta y del comportamiento de la gente.

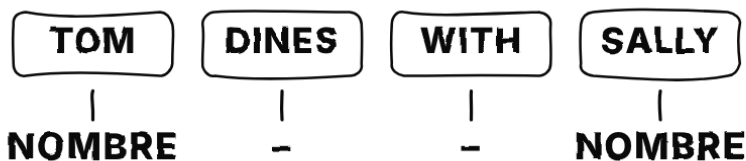


El orden y su bucle

Los sistemas de recomendación son primos cercanos de la búsqueda, con una gran diferencia: la lista se hace para usted personalmente. Un aficionado a la ciencia ficción y un amante de las viejas comedias de Peter Sellers deberían ver páginas muy distintas. Algunas opiniones son explícitas, como las estrellas y las reseñas; otras son implícitas, como saltarse una canción, que puede significar disgusto o simplemente un mal momento. El feed de una aplicación de vídeo que parece saber qué verá usted a continuación es un recomendador justo de este tipo: adivina una puntuación para cada vídeo y le muestra los mejores. Estos sistemas valen una fortuna, pero tienen defectos profundos. La gente valora sobre todo lo que ama u odia, así que las tres estrellas son curiosamente escasas, y las propias sugerencias del sistema influyen en lo que la gente compra, así que un producto que

promueve parece popular y se promueve todavía más, un bucle de retroalimentación que sigue siendo un problema de investigación abierto.

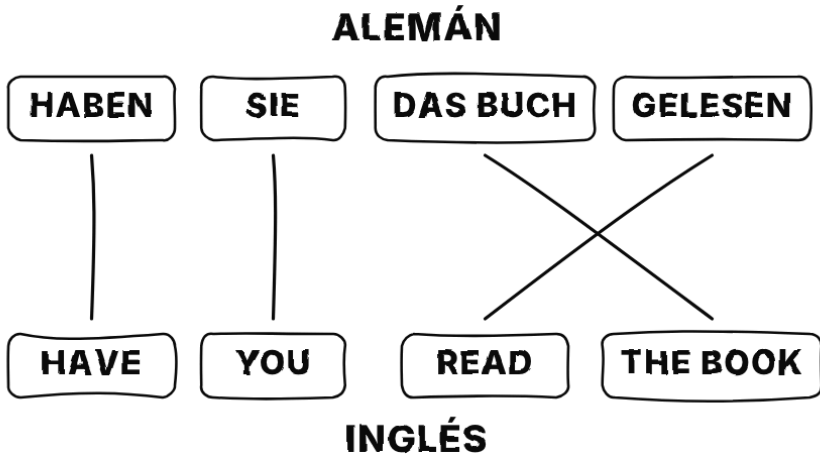
Hasta ahora cada ejemplo se trataba solo y se olvidaba una vez respondido. Pero un vídeo es una sucesión de fotogramas, y lo que ocurre en uno se entiende mucho mejor cuando se ven los que lo rodean. A un paciente en cuidados intensivos se le vigila hora tras hora, y un modelo que avisa del peligro debe recordar toda la historia, no solo la última lectura. Los problemas en los que la entrada, la salida o ambas son una secuencia de longitud variable se llaman aprendizaje de secuencias. Un chatbot lee todo su mensaje como una secuencia de palabras y responde con otra secuencia, así que es aprendizaje de secuencias de principio a fin.



Cómo se corresponden las secuencias

El libro nombra algunos tipos. En el etiquetado, cada palabra recibe una marca, como si forma parte de un nombre, así que entrada y salida se corresponden una a una. En el reconocimiento de voz, miles de mediciones de sonido forman una sola palabra, así que una entrada larga se convierte en una salida corta; la síntesis de voz va al revés, de un texto corto a un audio largo. Cuando usted dicta un mensaje a su asistente de voz, hace lo primero, y cuando le lee una respuesta en voz alta, hace lo segundo. La traducción automática es aún más difícil, porque cambia el orden: al alemán le gusta poner el verbo al final, así que una versión inglesa palabra por palabra diría «Have you yourself already this excellent textbook looked at?», algo como «¿Ha usted ya este excelente libro de texto mirado?». Leer la maquetación de una página web y mantener una conversación que

recuerde lo dicho hace mucho y cómo es el mundo son tareas aún más difíciles y todavía se investigan. Aprender sin ninguna etiqueta queda para la próxima parte.



La traducción cambia el orden

Para recordar

La clasificación multietiqueta permite que un ejemplo lleve varias etiquetas a la vez, y así funciona el etiquetado automático. La búsqueda y la recomendación tratan de poner elementos en orden, y la recomendación añade el gusto personal. Los recomendadores se enfrentan a opiniones sesgadas y a bucles en los que sus propias sugerencias hacen que los productos parezcan más populares. El aprendizaje de secuencias maneja entradas o salidas de longitud variable, en las que importan el orden y el contexto. El etiquetado, el reconocimiento de voz, la síntesis de voz y la traducción son tareas de secuencias que se distinguen por cómo se corresponden entrada y salida.

Preguntas

1. ¿Por qué un clasificador multiclase corriente encaja mal con una imagen de los músicos de Bremen?
 - a. La imagen tiene demasiados píxeles para un clasificador
 - b. Un modelo no puede distinguir animales
 - c. Debe elegir una clase, pero la respuesta honrada es cuatro animales a la vez
 - d. Las imágenes de cuentos nunca están en los datos de entrenamiento
2. ¿Cuál es la diferencia principal entre un sistema de recomendación y un buscador?

- a. Un recomendador ordena los elementos según el gusto de una persona concreta
 - b. Un recomendador nunca da puntuación a los elementos
 - c. Un buscador solo funciona con películas
 - d. Un buscador no puede ordenar páginas
3. El recomendador de una tienda promueve una batidora, así que más gente la compra, así que se promueve aún más. ¿Qué problema es este?
- a. Sobreajuste al conjunto de prueba
 - b. Un bucle de retroalimentación, en el que las propias sugerencias del sistema dan forma a los datos de los que aprende
 - c. Clasificación multietiqueta
 - d. Demasiado pocas valoraciones de tres estrellas
4. ¿Qué tarea convierte una entrada larga en una salida corta?
- a. La síntesis de voz
 - b. Etiquetar cada palabra de una frase
 - c. Traducir del alemán al inglés
 - d. El reconocimiento de voz

Respuestas

1. **c)** Cuando varias clases son ciertas a la vez, el problema es de clasificación multietiqueta, que permite al modelo decir burro, perro, gato y gallo juntos.
2. **a)** Los dos ponen en orden los elementos pertinentes, pero un recomendador es personal: el aficionado a la ciencia ficción y el amante de la comedia ven listas distintas.

- 3. b)** Las compras reflejan en parte las recomendaciones, así que el producto parece mejor de lo que es y se recomienda todavía más.
- 4. d)** En el reconocimiento de voz miles de mediciones de sonido forman una palabra, así que la transcripción es mucho más corta que la grabación; la síntesis de voz va al revés.

Ejercicios

1. El libro etiqueta la frase «Tom has dinner in Washington with Sally» («Tom cena en Washington con Sally»), marcando cada palabra que forma parte de un nombre. ¿Qué palabras reciben la marca, y por qué se dice que entrada y salida se corresponden en esta tarea?

Pista: Busque personas y lugares, y compare cuántas marcas hay con cuántas palabras.

Respuesta: Tom, Washington y Sally reciben la marca, y las otras cuatro palabras no. Hay exactamente una marca por palabra, en el mismo orden, así que la salida se corresponde una a una con la entrada, a diferencia de la traducción.

2. Una aplicación de música aprende de las canciones que la gente salta. Explique por qué saltar una canción es una prueba más débil que una valoración de una estrella, y qué significa eso para el recomendador.

Pista: Piense en otras razones para saltar una canción además de que no guste.

Respuesta: Saltar es una opinión implícita: puede significar disgusto o solo que la canción no encajaba con el momento. Una valoración de una estrella dice claramente que no gustó. Por eso un recomendador debe tratar los saltos como pistas inciertas y no como veredictos claros.

3. ¿Por qué un modelo que avisa de peligro en una unidad de cuidados intensivos tiene que ser un modelo de secuencias y no uno que mire solo las lecturas de cada hora por separado?

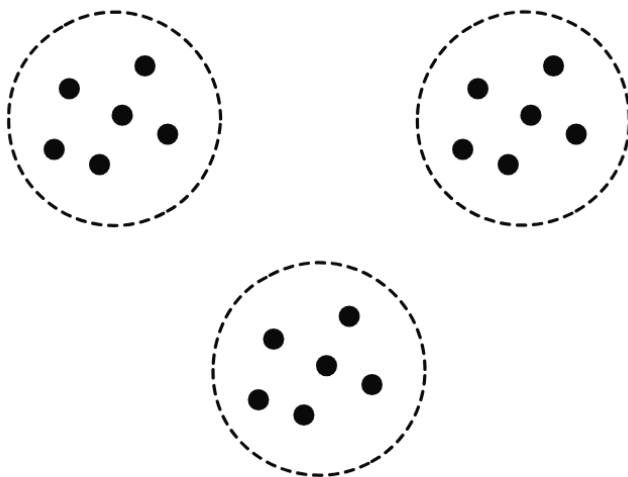
Pista: ¿Cómo se ve una tendencia preocupante en una sola lectura?

Respuesta: El peligro suele verse en la historia, como lecturas que se desvían a lo largo de muchas horas, algo que ninguna lectura sola revela. Un modelo que olvida los datos de cada hora no puede ver esa tendencia, así que debe tener en cuenta toda la secuencia de mediciones del paciente.

Aprender sin jefe, y un mundo que cambia

Abra la galería de fotos de su teléfono y busque los álbumes de «personas» o «lugares». Nadie le dijo al teléfono de quién es cada cara ni cómo es una playa, y aun así ha ordenado miles de fotos en montones que casi siempre tienen sentido. Eso es un aprendizaje de otro tipo que todo lo anterior, porque no había ninguna etiqueta.

AGRUPAMIENTO

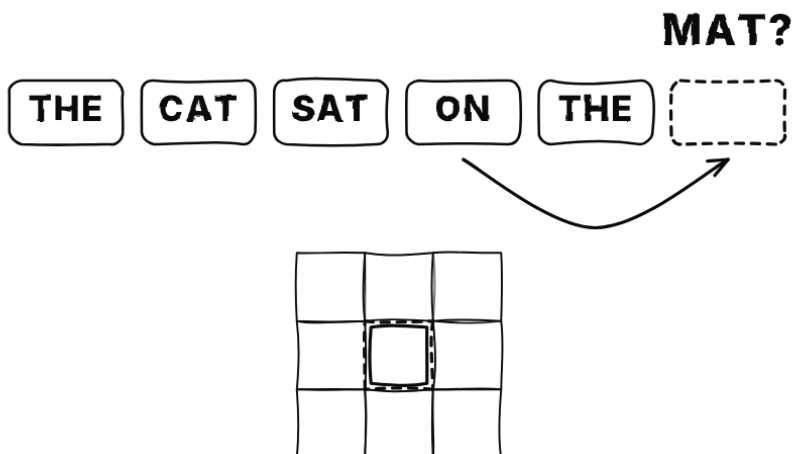


Grupos sin etiquetas

El libro presenta el aprendizaje supervisado como un trabajo con un jefe mandón que está detrás de usted y le dice exactamente qué hacer

en cada caso. Aburrido, pero fácil: usted copia al jefe hasta captar el patrón. Ahora imagine un jefe que le suelta una montaña de datos sobre la mesa y solo dice: «Haga algo de ciencia de datos con esto». Eso es el aprendizaje no supervisado, y las preguntas que puede hacer solo las limita su imaginación. Puede ordenar las cosas en grupos, como hace la galería, lo que se llama agrupamiento. Puede buscar unos pocos números que recojan lo importante, como un sastre describe un cuerpo entero con un puñado de medidas. Puede colocar las palabras en un espacio donde sus significados se comportan como posiciones, de modo que si parte de Roma, quita Italia y añade Francia, llega cerca de París. Puede buscar causas escondidas en tablas de precios, contaminación y salarios, o construir modelos generativos que aprenden tan bien cómo son los datos que pueden producir ejemplos nuevos propios. Las herramientas que crean una imagen nueva a partir de una frase que usted escribe son modelos generativos de este tipo.

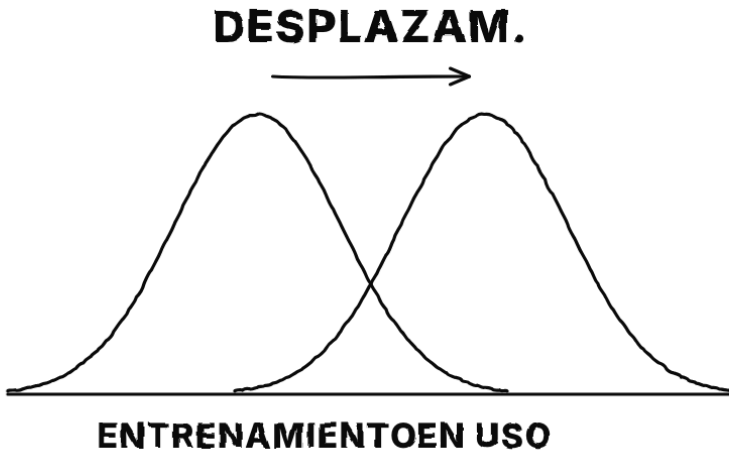
Un ingenioso camino intermedio es el aprendizaje autosupervisado, en el que los datos ponen sus propios ejercicios. Oculte una palabra en una frase y pida al modelo que la adivine por las palabras de alrededor; con todo el texto del mundo para practicar, ninguna persona tiene que etiquetar nada. Con imágenes, puede tapar un trozo y pedir al modelo que lo rellene. Lo que el modelo aprende así se ajusta después para el trabajo que de verdad necesita. El chatbot con el que usted habla empezó exactamente así, aprendiendo a adivinar las palabras que faltan o la siguiente en una enorme cantidad de texto.



Los datos ponen el ejercicio

Todos estos métodos comparten un supuesto silencioso: primero se reúnen los datos, después se aprende, y el modelo nunca toca el mundo mientras aprende. Eso se llama aprendizaje fuera de línea, y es sencillo, pero el mundo no es un montón de datos. Los sistemas reales actúan, y sus acciones cambian lo que ven después. Algunos entornos recuerdan lo que usted hizo, otros le ayudan, como una persona que habla con claridad a una aplicación de dictado, y otros luchan contra usted. Su filtro de spam tiene un enemigo: los spammers estudian lo que se bloquea y cambian sus mensajes para colarse. Cuando los datos que el modelo encuentra en uso difieren de los datos con los que se entrenó, tenemos un cambio de distribución. El ejemplo del libro es el examen que redacta el profesor después de

un trimestre de deberes puestos por los ayudantes. Cómo puede actuar un aprendiz en un mundo así queda para la próxima parte.



Cuando los datos se desplazan

Para recordar

El aprendizaje no supervisado encuentra estructura en datos sin etiquetas: grupos, unos pocos números clave, causas ocultas o ejemplos nuevos. El aprendizaje autosupervisado crea sus propias etiquetas a partir de los datos, como adivinar palabras ocultas, y luego se ajusta para una tarea. La mayor parte del aprendizaje es fuera de línea: primero vienen los datos y el modelo nunca actúa mientras aprende. Cuando el mundo que encuentra el modelo difiere de los datos con los que se entrenó, eso es un cambio de distribución.

Preguntas

1. Un teléfono ordena las fotos de las vacaciones en grupos de playas, montañas y caras sin que nadie las etiquete. ¿Qué tipo de aprendizaje es este?
 - a. Aprendizaje supervisado, porque los grupos son correctos
 - b. Aprendizaje no supervisado, en concreto agrupamiento
 - c. Aprendizaje por refuerzo
 - d. Regresión
2. ¿Por qué el aprendizaje autosupervisado no necesita etiquetado humano?
 - a. Los datos ponen sus propios ejercicios, por ejemplo una palabra oculta que las palabras de alrededor deben revelar
 - b. Solo usa conjuntos de datos pequeños

- c. Copia las etiquetas de un modelo supervisado
 - d. Solo aprende de imágenes
3. Un filtro de spam entrenado el año pasado deja pasar ahora muchos mensajes de spam nuevos. ¿Qué lo explica mejor?
- a. El filtro se entrenó en línea
 - b. El filtro usa agrupamiento
 - c. Un cambio de distribución: los spammers cambiaron sus mensajes, así que los datos nuevos difieren de los de entrenamiento
 - d. El filtro tiene demasiados pocos parámetros para leer correos
4. ¿Qué supuesto comparten, según esta parte, el aprendizaje supervisado y el no supervisado?
- a. Que cada ejemplo tiene una etiqueta
 - b. Que el modelo actúa en el mundo mientras aprende
 - c. Que los datos de prueba los escribe la misma persona que los de entrenamiento
 - d. Que primero se reúnen los datos y el modelo nunca actúa mientras aprende

Respuestas

1. b) No se dio ninguna etiqueta; encontrar grupos de ejemplos parecidos en datos sin etiquetar es agrupamiento, un tipo de aprendizaje no supervisado.

2. a) Ocultar una parte de los datos y pedir al modelo que la adivine convierte cualquier texto o imagen en material de práctica, con la parte oculta como respuesta.

- 3. c)** El entorno contraataca; cuando los datos en uso difieren de los de entrenamiento, el rendimiento del modelo baja.
- 4. d)** Los dos son aprendizaje fuera de línea: primero llega el montón de datos, y las salidas del modelo nunca cambian los datos que ve.

Ejercicios

1. El libro compara el cambio de distribución con un examen que redacta el profesor después de unos deberes puestos por los ayudantes. Explique con sus palabras por qué un estudiante al que le fueron bien los deberes podría tener dificultades, y cómo se traslada eso a un modelo.

Pista: Pregúntese con qué practicó el estudiante y con qué lo examinaron.

Respuesta: El estudiante practicó con preguntas al estilo de los ayudantes, pero el examen viene al estilo del profesor, así que las preguntas difieren de aquellas con las que se preparó. Con un modelo pasa lo mismo: aprende de los datos de entrenamiento, y si los datos que encuentra en uso vienen de otra fuente o de un mundo que ha cambiado, su buen rendimiento en el entrenamiento puede no trasladarse.

2. Para cada entorno, diga si ayuda al aprendiz, lucha contra él o ninguna de las dos cosas: una persona que habla despacio y con claridad a una aplicación de dictado; los spammers que escriben correos; el tiempo atmosférico.

Pista: Pregúntese si el entorno tiene un objetivo relacionado con el modelo.

Respuesta: La persona cuidadosa ayuda, porque quiere que la entiendan. Los spammers luchan, adaptando sus mensajes para burlar el filtro. El tiempo no hace ninguna de las dos cosas: puede cambiar con el tiempo, pero no en respuesta a los deseos del modelo.

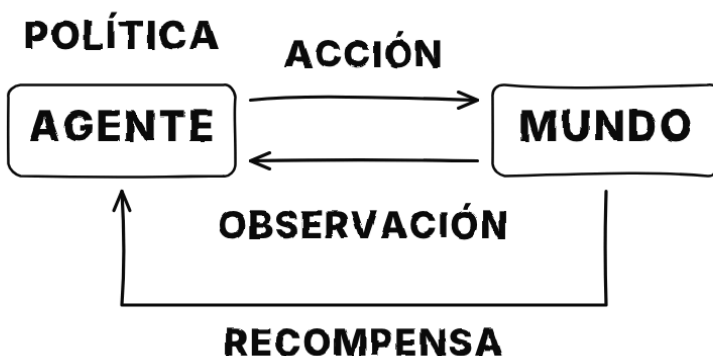
3. Proponga cómo convertir una gran colección de fotografías, ninguna etiquetada, en ejercicios de entrenamiento para un modelo autosupervisado.

Pista: Oculte algo que la propia fotografía pueda revelar.

Respuesta: Tape un trozo de cada fotografía y pida al modelo que lo rellene a partir del resto; la fotografía original da la respuesta correcta, así que nadie tiene que etiquetar nada. El modelo aprende sobre formas y texturas, y después puede ajustarse para una tarea real.

Aprender actuando, y dónde empezó todo

Observe a un niño que aprende un videojuego sin leer las instrucciones. Pulsa botones, la pantalla cambia, suben los puntos o pierde una vida, y al cabo de un rato juega bien. Nadie le dijo qué botón era el bueno en cada momento; solo veía la puntuación. Esa es la idea del aprendizaje por refuerzo, el tipo de aprendizaje pensado para cosas que actúan: robots, jugadores de juegos y programas que mantienen una conversación.

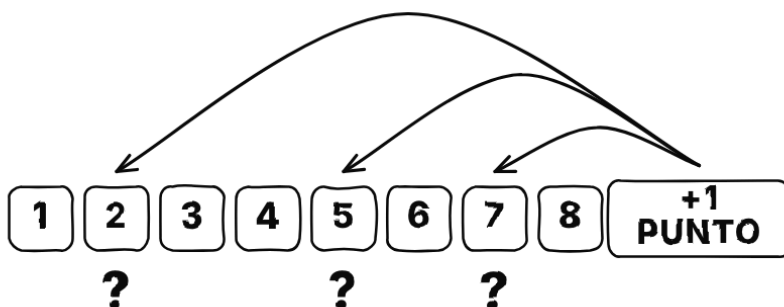


El bucle del agente

La imagen es un bucle. Un agente mira su mundo, que le da una observación; elige una acción; el mundo cambia y le devuelve una recompensa; y el bucle vuelve a empezar. La costumbre del agente de convertir lo que ve en lo que hace se llama su política, y todo el objetivo es aprender una buena. La idea es tan general que el aprendizaje supervisado corriente cabe dentro de ella, con una acción por cada clase y una recompensa igual a lo bien que salió la conjetura. Los éxitos famosos son un programa que aprendió viejos juegos de Atari solo a partir de la pantalla y venció a jugadores humanos, y AlphaGo, que venció al campeón del mundo de go.

Lo que lo hace difícil es lo que el mundo no dice. Nadie le dice al agente la mejor acción, solo la recompensa, y la recompensa puede llegar mucho después de las acciones que la ganaron. En el ajedrez la única recompensa real es ganar o perder al final, así que ¿cuál de cuarenta jugadas merece el mérito? Este es el problema de la asignación del mérito, el mismo al que se enfrenta alguien ascendido tras un año de buenas decisiones que quiere saber cuáles importaron. El agente puede además ver solo una parte de su mundo, como un robot de limpieza atascado en uno de varios armarios idénticos, que debe recordar cómo llegó allí para saber dónde está. Y debe elegir una y otra vez entre usar el mejor plan que conoce y probar algo nuevo que podría ser mejor: explotar o explorar. Un feed de recomendaciones vive con esta elección cada día: mostrarle más de lo que ya le gusta o probar algo nuevo y arriesgarse a que lo salte. Como todo esto a la vez es demasiado, los investigadores estudian casos más sencillos, que el libro nombra sin entrar en ellos.

¿MÉRITO?

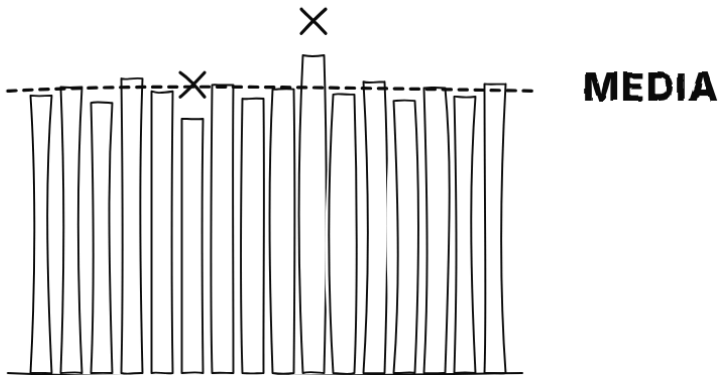


¿Qué jugada ganó la partida?

Ahora el libro da un paso atrás hacia la historia, porque el deseo de aprender de los datos tiene siglos. Gauss nos dio el método de los mínimos cuadrados, que aún se usa en seguros y en medicina, y los modelos sencillos de línea recta describen leyes como la relación entre la corriente y el voltaje en un cable. En el siglo dieciséis, el geómetra Jacob Köbel describió cómo fijar la longitud de un pie: detener a dieciséis hombres a la salida de la iglesia, alinear sus pies y tomar la media; una mejora posterior apartaba primero el pie más corto y el más largo, una de las primeras medias recortadas. Ronald Fisher construyó buena parte de la estadística moderna, y su pequeño conjunto de datos de flores de iris de 1936 aún se usa en demostraciones; también fue partidario de la eugenesia, un recordatorio de que los datos se han usado desde hace mucho para

malos fines además de buenos. La teoría de la información de Shannon y la pregunta de Turing, «¿Pueden pensar las máquinas?», también contribuyeron.

16 PIES



El pie de Köbel

El cerebro fue la otra inspiración. Donald Hebb propuso que las conexiones entre neuronas se refuerzan cuando ayudan, y esa idea de premiar el buen comportamiento y debilitar el malo recorre la forma en que hoy se entrenan las redes neuronales. Las redes neuronales tomaron su nombre del cerebro, aunque el parecido se ha vuelto vago; lo que queda son capas de procesamiento sencillo apiladas unas sobre otras y una manera de ajustar todas las capas a la vez.

Sin embargo, aproximadamente de 1995 a 2005 las redes neuronales pasaron de moda. La potencia de cálculo escaseaba y los conjuntos de datos eran diminutos: una colección de sesenta mil dígitos escritos a mano se consideraba enorme. Con tan pocos datos y tan poco cálculo, otros métodos, como los árboles de decisión y los métodos de núcleo, simplemente funcionaban mejor, se entrenaban en horas en lugar de semanas y venían con garantías sólidas. Muchos filtros de spam de aquella época se basaban en esos métodos clásicos y no en redes neuronales. Cómo se inclinó esa balanza queda para la próxima parte.

Para recordar

El aprendizaje por refuerzo es para agentes que actúan: observan, actúan, reciben recompensas y aprenden una política. Es difícil porque la mejor acción nunca se muestra, las recompensas llegan tarde, el mundo puede verse solo en parte y el agente debe equilibrar probar cosas nuevas con usar lo que funciona. Aprender de los datos tiene siglos, desde promediar pies y los mínimos cuadrados de Gauss hasta la estadística de Fisher y la pregunta de Turing. Las redes neuronales tomaron del cerebro su nombre y algunas ideas. Entre 1995 y 2005, más o menos, perdieron frente a otros métodos, porque los datos y la potencia de cálculo eran demasiado escasos.

Preguntas

1. ¿Qué hace que el aprendizaje por refuerzo sea más difícil que el supervisado?

- a. Al agente nunca se le muestra la mejor acción, solo una recompensa que puede llegar mucho después
 - b. El agente no tiene ninguna observación
 - c. Las recompensas se dan por todas las acciones posibles a la vez
 - d. No se puede usar para juegos
2. Un programa de ajedrez gana una partida tras cuarenta jugadas. ¿A qué problema se enfrenta al aprender de esta victoria?
- a. Cambio de distribución
 - b. Observabilidad parcial
 - c. Asignación del mérito: averiguar qué jugadas merecen el mérito
 - d. Sobreajuste al conjunto de prueba
3. Una aplicación de música pone solo el género que más le gusta y nunca prueba otra cosa. En términos de aprendizaje por refuerzo, ¿qué está haciendo?
- a. Explorar
 - b. Explotar la mejor estrategia conocida y no explorar nunca
 - c. Asignar el mérito
 - d. Entrenar fuera de línea
4. ¿Por qué otros métodos superaron a las redes neuronales entre 1995 y 2005, más o menos?
- a. Las redes neuronales aún no se habían inventado
 - b. Los otros métodos necesitaban más datos

- c. Las redes neuronales no se podían entrenar con recompensas
- d. Los datos y la potencia de cálculo escaseaban, y otros métodos se entrenaban más rápido y con garantías sólidas

Respuestas

- 1. **a)** En el aprendizaje supervisado cada ejemplo lleva la respuesta correcta; aquí el mundo da solo recompensas, a menudo mucho después de las acciones que las ganaron.
- 2. **c)** La única recompensa llega al final, así que el aprendiz debe decidir qué jugadas anteriores la ganaron.
- 3. **b)** Usar el mejor plan que conoce es explotar; al no explorar nunca, renuncia a la posibilidad de encontrar algo mejor.
- 4. **d)** Con conjuntos de datos pequeños y poca potencia de cálculo, los árboles de decisión y los métodos de núcleo funcionaban mejor y tardaban horas en vez de semanas.

Ejemplos resueltos

Una media recortada de pies Al estilo del método de Köbel, se miden los pies de seis hombres: 26, 27, 28, 27, 31 y 21 centímetros. Halle la media simple y luego la media recortada tras apartar el pie más corto y el más largo.

Media simple: las seis longitudes suman 160 centímetros, y 160 dividido entre 6 da unos 26,7 centímetros.

Para recortar, quite el pie más corto, 21, y el más largo, 31, porque los pies poco comunes desviarían la media.

Los cuatro restantes suman 26 más 27 más 28 más 27, es decir 108 centímetros.

108 dividido entre 4 es 27 centímetros.

Respuesta: Media simple de unos 26,7 cm; media recortada de 27 cm

Ejercicios

1. Cinco mediciones de un pie dan 25, 29, 28, 27 y 36 centímetros.

Halle la media recortada tras quitar la más corta y la más larga, y diga por qué aquí es mejor estimación que la media simple.

Pista: Quite primero 25 y 36, y luego promedie lo que queda.

Respuesta: 28 cm: 29 más 28 más 27 son 84, y 84 dividido entre 3 es 28. La media simple, 29 cm, la sube el único pie inusualmente largo, que la media recortada ignora.

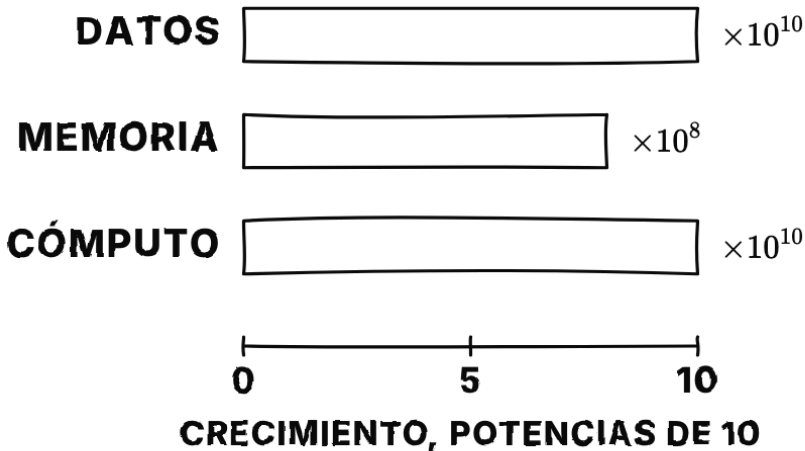
2. El libro describe un robot de limpieza atrapado en uno de varios armarios idénticos. Explique por qué mirar alrededor del armario no basta para que el robot sepa dónde está, y qué necesita en su lugar.

Pista: ¿Qué vería el robot en cada uno de los armarios?

Respuesta: Todos los armarios se ven iguales desde dentro, así que la observación actual no revela el estado del robot; esto es observabilidad parcial. El robot debe usar sus observaciones anteriores, cómo se movió antes de entrar, para averiguar en qué armario está.

Qué cambió: datos, chips y herramientas

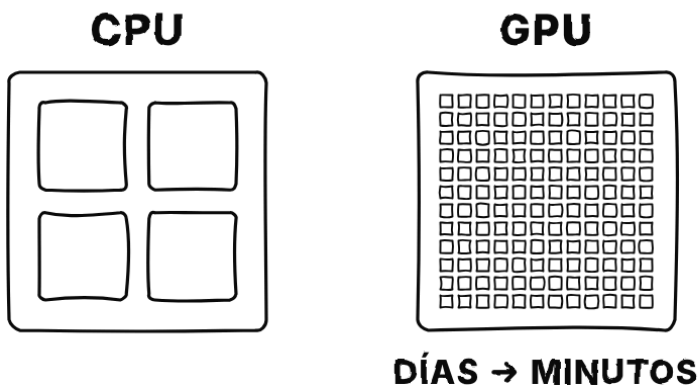
Piense en las fotos de su teléfono: miles, cada una de varios millones de píxeles, guardadas casi sin coste y copiadas a la nube sin pensarlo. Hace treinta años la memoria de un ordenador entero no habría podido contener ni una. Ese cambio, en la cantidad de datos que existen y en lo barato que es guardarlos y procesarlos, es lo que sacó a las redes neuronales del olvido.



Crecimiento de 1970 a 2020

El libro lo muestra con una tabla, una fila por década. En los años setenta un conjunto de datos típico tenía unos cien ejemplos, como los iris de Fisher; en los años veinte de este siglo los datos de una red social llegaban a cerca de un billón. La memoria también creció, pero mucho más despacio: de alrededor de un kilobyte a alrededor de cien gigabytes. La potencia de cálculo fue la que más creció, de cien mil operaciones por segundo a unos mil billones. La conclusión es lo interesante: como el cálculo ha superado a la memoria, los mejores modelos son los que guardan poco pero calculan mucho, y eso es justo lo que hacen las redes neuronales profundas. Por eso muchas ideas antiguas se redescubrieron en la última década, después de años sin usarse.

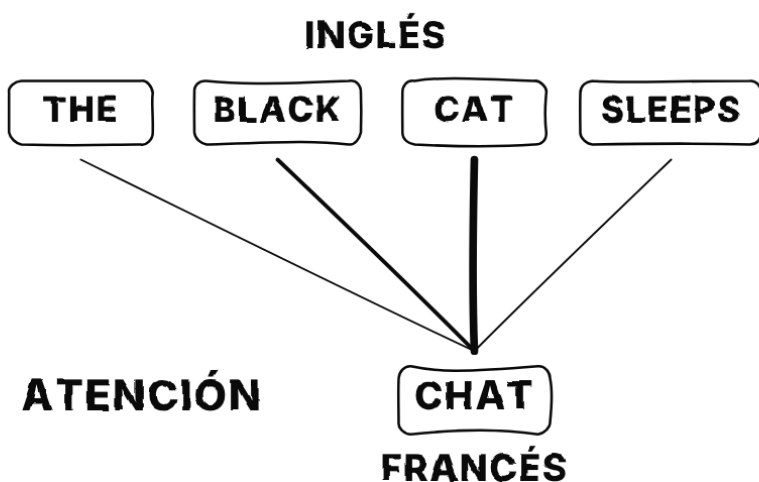
La máquina decisiva fue la tarjeta gráfica. Las GPU se diseñaron para dibujar videojuegos, haciendo enormes cantidades de cálculos sencillos en paralelo, y eso resultó ser justo lo que necesita el entrenamiento de una red neuronal. De pronto, modelos que habían parecido imposiblemente caros quedaron al alcance. Repartir el trabajo entre muchas tarjetas redujo el tiempo de entrenamiento de un modelo de imágenes estándar de días a menos de siete minutos.



Muchos trabajadores pequeños

Pero no fue solo verter más potencia sobre recetas viejas; también llegó una explosión de ideas nuevas, que el libro compara con el súbito florecimiento de la vida en el período Cámbrico. El dropout añade ruido aleatorio dentro de una red mientras se entrena, lo que ayuda contra el sobreajuste. La atención permite que un modelo señale hacia atrás las partes de su entrada que importan en ese momento, en lugar de meter toda una frase en la memoria antes de traducirla. El Transformer, construido solo con atención, sigue mejorando a medida que crecen los datos, el tamaño del modelo y el cálculo, y un único Transformer ha aprendido a conversar, a poner pies de foto, a jugar a Atari y a dirigir un robot. Los modelos de lenguaje, que predicen texto a partir de texto, adquirieron capacidades sorprendentes al escalarlos. ChatGPT, que quizá usted

mismo usa, es un modelo de lenguaje de este tipo, ajustado para seguir lo que la gente realmente le pide. Las redes generativas antagónicas y, más recientemente, los modelos de difusión aprendieron a producir imágenes realistas; los modelos de difusión construyen una imagen quitando ruido paso a paso. Las aplicaciones que dibujan una imagen a partir de una frase que usted escribe usan modelos de difusión de este tipo.



La atención señala hacia atrás

El último ingrediente es el software. Los marcos de aprendizaje profundo gratuitos, desde los primeros como Theano y Torch hasta los actuales PyTorch, TensorFlow y JAX, se encargan por usted de las matemáticas difíciles y del hardware. El libro da una medida de cuánto cambió eso: en 2014, entrenar un clasificador sencillo se ponía

como un verdadero problema de deberes a los nuevos doctorandos de la Universidad Carnegie Mellon; hoy lleva menos de diez líneas de código. Esta parte ha dejado fuera los detalles de cada método, que tratan los capítulos siguientes.

Para recordar

Las redes neuronales volvieron porque los datos se hicieron abundantes y el almacenamiento y el cálculo se abarataron. La potencia de cálculo creció más deprisa que la memoria, lo que favorece a los modelos que guardan poco y calculan mucho, como las redes profundas. Las tarjetas gráficas hechas para juegos resultaron ideales para entrenarlas. Las ideas nuevas, como el dropout, la atención, los Transformers, los modelos de lenguaje y los modelos de difusión, impulsaron el progreso tanto como la potencia bruta. Los marcos gratuitos lo hicieron todo lo bastante fácil para que cualquier programador lo use.

Preguntas

1. La potencia de cálculo ha crecido mucho más deprisa que la memoria. ¿Qué tipo de modelo favorece eso?
 - a. Modelos que guardan muchísimo pero calculan poco
 - b. Modelos que guardan poco pero calculan mucho, como las redes neuronales profundas
 - c. Modelos que no necesitan datos
 - d. Modelos que funcionan sin ordenador

2. ¿Por qué las tarjetas gráficas resultaron tan útiles para el aprendizaje profundo?
 - a. Tienen más memoria que cualquier otro chip
 - b. Se diseñaron para redes neuronales desde el principio
 - c. Hacen enormes cantidades de cálculos sencillos en paralelo, que es lo que necesita el entrenamiento
 - d. Guardan los conjuntos de datos más barato
3. ¿Qué le permite hacer la atención a un modelo de traducción?
 - a. Señalar hacia atrás las partes de la entrada que importan ahora, en lugar de guardar antes toda la frase en memoria
 - b. Añadir ruido aleatorio para evitar el sobreajuste
 - c. Generar imágenes a partir de ruido
 - d. Entrenar en muchas tarjetas gráficas a la vez
4. Según el libro, ¿qué hizo el aprendizaje profundo accesible a los programadores corrientes?
 - a. Conjuntos de datos más pequeños
 - b. El fin del sobreajuste
 - c. El aprendizaje por refuerzo
 - d. Marcos gratuitos que se encargan de las matemáticas y del hardware

Respuestas

1. b) Cuando el cálculo es barato y la memoria relativamente escasa, el punto óptimo pasa a los modelos que dedican muchos cálculos a pocos números guardados, que es lo que hacen las redes profundas.

2. c) Las GPU se construyeron para dibujar juegos haciendo muchos cálculos sencillos a la vez, y entrenar una red neuronal es justo ese tipo de trabajo.

3. a) Con atención el modelo no tiene que meter toda una frase en una memoria fija; vuelve la vista a las partes pertinentes sobre la marcha.

4. d) Una tarea que en 2014 era un problema de deberes de doctorado hoy lleva menos de diez líneas de código, gracias a los marcos.

Ejemplos resueltos

Muchas tarjetas, un gran lote El entrenamiento se reparte entre 1024 tarjetas gráficas, y cada tarjeta procesa un pequeño lote de 32 imágenes a la vez. ¿Cuántas imágenes se procesan juntas en un paso entre todas las tarjetas?

Cada tarjeta toma su propio lote de 32 imágenes.

Las 1024 tarjetas trabajan a la vez, así que los lotes se suman.

1024 por 32 son 32 768.

Respuesta: 32 768 imágenes, unas 32 000

Ejercicios

1. En la tabla del libro, un conjunto de datos típico pasó de 100 ejemplos en 1970 a 1 billón en 2020, mientras que la memoria pasó de 1 kilobyte a 100 gigabytes. Tomando un gigabyte como un millón de kilobytes, ¿por qué factor creció cada uno, y cuál creció más?

Pista: Un billón es un 1 seguido de doce ceros; 100 gigabytes son 100 millones de kilobytes.

Respuesta: Los datos crecieron por un factor de diez mil millones (de 100 a un billón); la memoria, por un factor de cien millones (de 1 a 100 millones de kilobytes). Los datos crecieron unas cien veces más, y por eso la memoria no ha seguido el ritmo y los modelos deben ahorrar memoria.

2. Si 512 tarjetas tomaran cada una un lote de 64 imágenes, ¿cuántas imágenes procesaría un paso, y cómo se compara con 1024 tarjetas de 32 cada una?

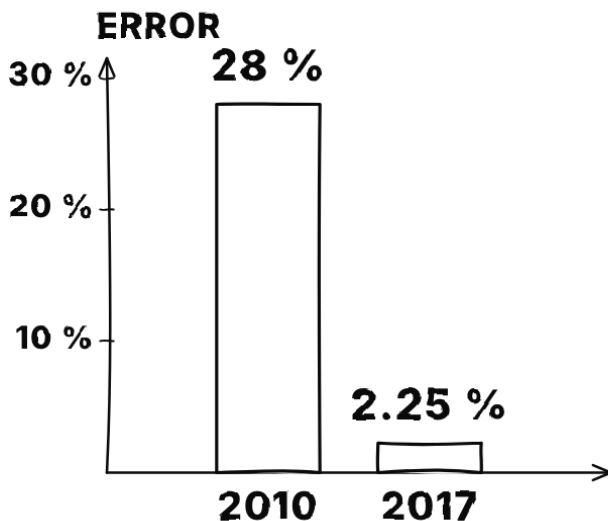
Pista: Multiplique el número de tarjetas por el lote de cada tarjeta.

Respuesta: 512 por 64 son 32 768 imágenes, exactamente lo mismo que 1024 por 32: reducir a la mitad las tarjetas y duplicar el lote de cada una deja el total igual.

Dónde ya funciona, y qué lo hace profundo

Pague hoy algo por internet y, en el segundo antes de que se complete el pago, un modelo lo revisa en busca de fraude. Envíe una carta y, en muchos países, una máquina lee la dirección escrita a mano; justo esa tarea nos dio la famosa colección de dígitos escritos a mano. El aprendizaje automático hace un trabajo silencioso como este desde los años noventa: lee cheques, puntúa solicitudes de préstamo, ordena resultados de búsqueda, elige qué mostrarle a usted. Está en todas partes, pero casi siempre fuera de la vista.

Lo nuevo es que ahora hace cosas que la gente puede ver y que se creían imposibles. Los asistentes de voz responden bastante bien a peticiones habladas, desde encender una luz hasta pedir cita en la peluquería, y el reconocimiento de voz ya iguala a las personas en algunas tareas. Cuando usted le pide a su asistente de voz el tiempo de mañana y le entiende en una habitación ruidosa, está oyendo este progreso. Reconocer objetos en fotos pasó de difícil a rutinario: en una prueba estándar, los mejores sistemas de 2010 fallaban alrededor del veintiocho por ciento de las veces, incluso con cinco intentos; en 2017 fallaban alrededor del dos por ciento. Los mismos métodos identifican ahora cantos de aves y ayudan a diagnosticar el cáncer de piel.



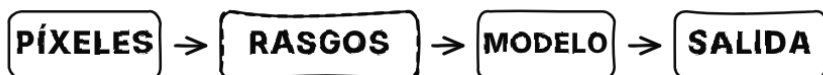
Error en ImageNet

Los juegos han sido desde hace mucho una vara de medir. Un programa de backgammon aprendió por refuerzo hace décadas; Deep Blue venció a Kaspárov al ajedrez con hardware especial y búsqueda por fuerza bruta; el go, con muchísimas más posiciones posibles, cayó ante AlphaGo, que combinó el aprendizaje profundo con el muestreo del árbol de jugadas; y un programa llamado Libratus venció a jugadores de primera fila al póquer, un juego en el que no se ven las cartas de los rivales. Los coches autónomos han avanzado de verdad, aunque la autonomía completa sigue fuera de alcance; allí el aprendizaje profundo se encarga sobre todo de ver, mientras los ingenieros ajustan el resto a mano. La robótica, la logística, la biología, la física de partículas y la astronomía deben avances recientes a estas herramientas.

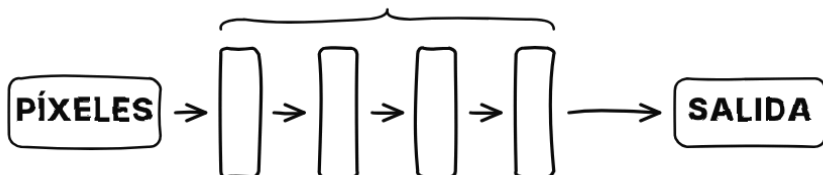
¿Y el miedo a máquinas que cobren conciencia y se vuelvan contra nosotros? Los autores se lo toman con calma. Los sistemas actuales se construyen para objetivos concretos, y detrás de cualquier apariencia de inteligencia general hay reglas, recetas prácticas y modelos estadísticos; no existen herramientas para una máquina que se mejore y se rediseñe a sí misma para resolver cualquier tarea. Las preocupaciones que consideran más urgentes están más cerca: la automatización de trabajos rutinarios, como los robots agrícolas automatizarán la cosecha, con grandes efectos en quienes dependen de ese trabajo; y modelos descuidados que toman decisiones sesgadas sobre préstamos, empleos o fianzas. El mismo cuidado vale para el feed de recomendaciones de su teléfono, que toma cada día pequeñas decisiones automáticas sobre lo que ven millones de personas.

Entonces, al final, ¿qué es el aprendizaje profundo? Es aprendizaje automático con modelos hechos de muchas capas de transformaciones, todas aprendidas juntas a partir de los datos. Antes de él, los expertos pasaban años diseñando a mano los pasos que convierten los píxeles o el sonido en bruto en características útiles, y solo entonces entregaban esas características a un modelo sencillo. El aprendizaje profundo entrena toda la cadena a la vez, de extremo a extremo, desde la entrada en bruto hasta la respuesta, y los filtros que encuentra por sí mismo superan a los que diseñaba la gente. Por eso brilla con el sonido en bruto, los píxeles en bruto y las frases de cualquier longitud, y por eso ha derribado los muros entre la visión, el habla, el lenguaje y la medicina, dándoles un conjunto común de herramientas.

A MANO



APRENDIDO JUNTO



A mano frente a extremo a extremo

También refleja un cambio de estilo. Con abundancia de datos, el campo depende menos de supuestos simplificadores y más de modelos flexibles que se ajustan mucho a los datos, igual que los físicos pasaron de aproximaciones resueltas a mano a simulaciones por ordenador: más precisas, aunque a menudo más difíciles de interpretar. Acepta soluciones que son buenas en lugar de demostrablemente óptimas, y prueba las cosas antes de demostrarlas. Y comparte: las herramientas, los modelos y las redes entrenadas se publican abiertamente, y por eso este libro también es gratuito.

Con esto termina la introducción. A lo largo de estas partes, la historia ha ido de los programas escritos como reglas a los programas que aprenden de ejemplos, pasando por las muchas formas que puede

tomar el aprendizaje, hasta los datos, los chips y las herramientas que por fin hicieron funcionar las redes profundas.

Para recordar

El aprendizaje automático lleva mucho tiempo trabajando fuera de la vista, en controles de fraude, clasificación del correo y búsqueda, y el aprendizaje profundo ha traído ahora éxitos visibles en el habla, las imágenes y los juegos. Los autores ven los riesgos reales no en máquinas conscientes sino en empleos perdidos y decisiones automáticas sesgadas. Aprendizaje profundo significa muchas capas de transformaciones aprendidas juntas, de extremo a extremo, desde los datos en bruto hasta la respuesta. Sustituyó a las características diseñadas a mano y dio a campos como la visión, el habla y el lenguaje un conjunto común de herramientas. Su progreso se apoya en datos abundantes, cálculo barato y tarjetas gráficas, y marcos abiertos que cualquiera puede usar.

Preguntas

1. Según esta parte, ¿qué distingue sobre todo al aprendizaje profundo del aprendizaje automático anterior?
 - a. No usa ningún dato
 - b. Sus muchas capas se aprenden juntas, de extremo a extremo, en lugar de características diseñadas a mano que alimentan un modelo sencillo

- c. Solo funciona con juegos
 - d. Siempre encuentra la solución demostrablemente óptima
2. ¿Qué riesgo de la IA consideran hoy más urgente los autores?
- a. Que las máquinas cobren conciencia y se vuelvan contra sus creadores
 - b. Que los ordenadores se queden sin memoria
 - c. La automatización de trabajos rutinarios y modelos descuidados que toman decisiones sesgadas
 - d. Que los juegos se vuelvan demasiado fáciles para los ordenadores
3. ¿Por qué el póquer era para un ordenador un reto distinto del ajedrez?
- a. El póquer tiene menos posiciones posibles que el backgammon
 - b. El póquer no tiene reglas
 - c. El ajedrez no se puede aprender por refuerzo
 - d. En el póquer no se ven las cartas de los rivales, así que el juego solo se observa en parte
4. ¿Por qué el aprendizaje profundo ha derribado los muros entre campos como la visión, el habla y el lenguaje?
- a. Porque el entrenamiento de extremo a extremo sustituye gran parte de la preparación propia de cada campo, así que las mismas herramientas sirven en todos
 - b. Porque todos esos campos usan ahora los mismos conjuntos de datos
 - c. Porque la visión y el habla son el mismo problema

d. Porque los marcos prohíben los métodos especializados

Respuestas

1. **b)** Antes, los expertos diseñaban a mano los pasos de las características; el aprendizaje profundo entrena toda la cadena desde la entrada en bruto hasta la respuesta a la vez.

2. **c)** No ven herramientas para una inteligencia general que se mejore a sí misma, pero sí efectos reales de la automatización y de las decisiones automáticas sesgadas sobre préstamos, empleos o fianzas.

3. **d)** El ajedrez muestra todo el tablero, pero en el póquer parte de la situación está oculta, lo que hace mucho más difícil razonar.

4. **a)** Cuando las características se aprenden de los datos en bruto en lugar de diseñarlas los expertos de cada campo, un único conjunto de herramientas sirve para todos.

Ejemplos resueltos

Cuánto mejoró el reconocimiento de objetos En la prueba de ImageNet, los mejores sistemas de 2010 tenían una tasa de error top-5 del 28 por ciento; en 2017 era del 2,25 por ciento. De 10 000 imágenes de prueba, ¿cuántas falló cada uno, y cuántas veces menos errores son?

La tasa de error es la proporción de ejemplos en que el modelo se equivoca.

En 2010: el 28 por ciento de 10 000 son 2800 imágenes falladas.

En 2017: el 2,25 por ciento de 10 000 son 225 imágenes falladas.

2800 dividido entre 225 da unos 12,4.

Respuesta: 2800 errores frente a 225: unas doce veces menos

Ejercicios

1. Un sistema con una tasa de error top-5 del 2,25 por ciento se aplica a 40 000 fotografías. ¿Cuántas fallará aproximadamente, y cuántas habría fallado un sistema de 2010 con un 28 por ciento de error?

Pista: Multiplique el número de fotografías por cada tasa de error escrita como fracción.

Respuesta: 900 el sistema de 2017 (0,0225 por 40 000) y 11 200 el sistema de 2010 (0,28 por 40 000).

2. Explique con sus palabras la diferencia entre la forma antigua de reconocer objetos en imágenes, con extractores de características diseñados a mano, y la forma del aprendizaje profundo. ¿Por qué ganó la nueva?

Pista: Pregúntese quién decide en cada caso cómo se convierten los píxeles en bruto en características.

Respuesta: En la forma antigua, los expertos diseñaban a mano los pasos que convierten los píxeles en características, como detectores de bordes, y después un modelo sencillo usaba esas características. En el aprendizaje profundo toda la cadena, características incluidas, se aprende de los datos de extremo a extremo. Ganó porque un algoritmo que prueba millones de opciones automáticamente encuentra mejores filtros que el ingenio humano solo.