

Aprendizaje Automático (ML)

Dr. Juan Carlos Cruz González

Mayo 2023

Índice general

1	Árbol de aprendizaje automático	2
2	Teoría básica del aprendizaje automático	5
2.1	Maquina de aprendizaje	5
2.2	Métodos de aprendizaje automático	6
2.3	Rendimiento esperado de los algoritmos de aprendizaje automático	7
3	Clasificación y regresión	8
3.1	Reconocimiento y clasificación de patrones	8
3.2	Regresión	10
	Lista de algoritmos	11
	Bibliografía	12

Resumen

El aprendizaje automático (más conocido como Machine Learning) es un subconjunto de la inteligencia artificial. Este escrito presenta primero un árbol de aprendizaje automático y luego se centra en los métodos de álgebra matricial en el aprendizaje automático, incluida la optimización de un solo objetivo, la selección de características, el análisis de componentes principales y el análisis de correlación canónica junto con el aprendizaje supervisado, no supervisado y semisupervisado y el aprendizaje activo. aprendiendo. Lo que es más importante, se destaca temas seleccionados y avances en el aprendizaje automático: aprendizaje automático de gráficos, aprendizaje por refuerzo, Q-learning y aprendizaje por transferencia. En este texto encontraras pseudocódigo que podrás implementar en el lenguaje de programación que elijas.

Árbol de aprendizaje automático

El aprendizaje automático (ML) es un subconjunto de la inteligencia artificial, que construye un modelo matemático basado en datos de muestra, conocidos como "datos de entrenamiento", para hacer predicciones o decisiones sin estar programado explícitamente para realizar la tarea. En cuanto al aprendizaje, una buena definición dada por Mitchell [1] es:

Se dice que un programa de computadora aprende de la experiencia E con respecto a alguna clase de tareas T y mide el desempeño P si su desempeño en las tareas en T , medido por P , mejora con la experiencia E .

En el aprendizaje automático, las redes neuronales, las máquinas de vectores de soporte y la computación evolutiva, generalmente se nos proporciona un **conjunto de entrenamiento** (training set) y un **conjunto de prueba** (test set). Para hablar de todos estos términos, es necesario tener a la mano su definición.

Definición 1.0.1. Sean F, C y V conjuntos y F, C finitos. Una **tabla de datos** TD consta del conjunto $F \times C$ junto con una función $\vartheta : TD \rightarrow V$ que llamaremos **función de valores de atributos**. Si $(a, b) \in TD$, entonces $\vartheta(a, b)$ se llama **valor de atributo** de (a, b) . Al conjunto F lo denominamos **fila de la tabla de datos** y lo denotamos como $\text{Fila}(TD) := F$, el conjunto C se llama **columna de la tabla de datos** y lo denotamos como $\text{Col}(TD) := C$ y finalmente el conjunto V se llama **conjunto de valores de la tabla de datos**.

Denotaremos a una tabla de datos como (TD, ϑ) .

Definición 1.0.2. Sea (TD, ϑ) , Una **sub-tabla de datos** (S, ϑ') de TD es por si misma una tabla de datos que cumple que $\text{Fila}(S) \subseteq \text{Fila}(TD)$, $\text{Col}(S) \subseteq \text{Col}(TD)$ y $\vartheta' = \vartheta \upharpoonright_{\text{Fila}(S) \times \text{Col}(S)}$

Definición 1.0.3. Sea (TD, ϑ) una tabla de datos con $\text{Fila}(TD) = F = \{f_1, \dots, f_n\}$ y $\text{Col}(TD) = C = \{c_1, \dots, c_m\}$. Un **vector de datos** es de la forma:

$$(\vartheta(f_i, c_1), \vartheta(f_i, c_2), \dots, \vartheta(f_i, c_m)), \text{ con } i \in \{1, \dots, n\}$$

Y lo llamamos *i-ésimo vector de datos de TD* .

Aquí trabajaremos con tablas de datos cuyo conjunto de valores es \mathbb{R} .

Definición 1.0.4. Sea (TD, ϑ) una tabla de datos. Un **conjunto etiquetado** (labeled set) subyacente de TD es de la forma:

$$(X_l, Y_l) = \{(x_1, y_1), \dots, (x_l, y_l)\} \in \mathbb{R}^D \times (\mathbb{R} \cup \{1, \dots, M\}),$$

donde $l = |\text{Fila}(TD)|$, $D = |\text{Col}(TD)|$, x_i es el *i-ésimo vector de datos D -dimensional* y $y_i \in \mathbb{R}$ o $y_i \in \{1, \dots, M\}$ es la etiqueta correspondiente del vector de datos x_i .

Definiciones 1.0.5. Sea (TD, ϑ) una tabla de datos: Un **conjunto no etiquetado** (unlabeled set) subyacente de TD esta compuesto por los vectores de datos y se denota como:

$$X_u = \{x_1, \dots, x_u\}$$

donde $u = |\text{Fila}(TD)|$ y x_i es el i -ésimo vector de datos.

Definiciones 1.0.6. Sea (TD, ϑ) una tabla de datos:

- Un **conjunto de entrenamiento** (training set) es la unión del conjunto etiquetado y el conjunto no etiquetado de ejemplos disponibles para los aprendices automáticos.
- Un **conjunto de prueba** (test set) consta de vectores de datos no considerados en el conjunto de entrenamiento.

En problemas de regresión, y_i es la regresión o ajuste de x_i . En problemas de clasificación, y_i es la etiqueta de clase correspondiente de x_i entre las M clases de objetivos. Los datos etiquetados x_i , $i = 1, \dots, l$ son observado por el usuario, mientras que y_i , $i = 1, \dots, l$ son etiquetados por expertos o supervisores de etiquetado de datos. El aprendizaje automático tiene como objetivo establecer un regresor o clasificador a través del aprendizaje del conjunto de entrenamiento y luego evaluar el rendimiento del regresor o clasificador a través del conjunto de prueba.

Según la naturaleza de los datos de entrenamiento, podemos clasificar el aprendizaje automático de la siguiente manera.

(1) *Aprendizaje de datos estructurados regular o euclidiano*

- *Aprendizaje supervisado*: Dado un conjunto de entrenamiento que consta de datos etiquetados (es decir, entradas de ejemplo y sus salidas deseadas) $(X_e, Y_e) = (x_1, y_1), \dots, (x_l, y_l)$, el aprendizaje supervisado aprende una regla general que asigna entradas a salidas. Esto es como un "maestro" o un supervisor (experto en etiquetado de datos) dando una estudiante un problema (encontrar la relación de mapeo entre entradas y salidas) y sus soluciones (datos de salida etiquetados) y decirle al estudiante que descubra cómo resolver otros problemas similares: encontrar el mapeo de las características de muestras no vistas a sus etiquetas correctas o valores objetivo en el futuro.
- *En el aprendizaje no supervisado*, el conjunto de entrenamiento consta únicamente del conjunto no etiquetado $X_e = X_u = \{x_1, \dots, x_u\}$. La tarea principal del aprendiz automático es encontrar las soluciones por sí mismo (es decir, patrones, estructuras o conocimiento en datos no etiquetados). Esto es como darle a un estudiante un conjunto de patrones y pedirle que descubra los motivos subyacentes que generaron los patrones.
- *Aprendizaje semisupervisado*: dado un conjunto de entrenamiento

$$(X_e, Y_e) = \{(x_1, y_1), \dots, (x_l, y_l)\} \cup \{x_{l+1}, \dots, x_{l+u}\}$$

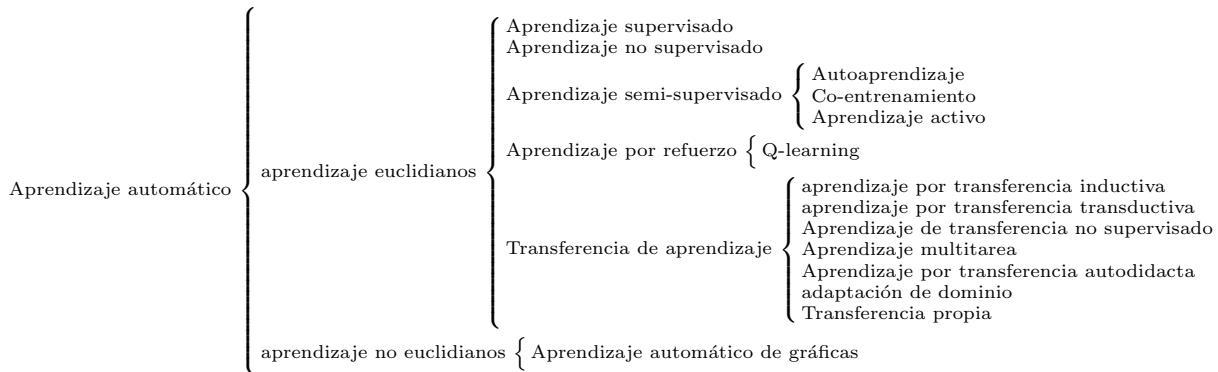
con $l \ll u$, es decir, nos dan una pequeña cantidad de datos etiquetados junto con una gran cantidad de datos sin etiquetar. El aprendizaje semisupervisado se encuentra entre el aprendizaje no supervisado (sin datos de entrenamiento etiquetados) y el aprendizaje supervisado (con datos de entrenamiento completamente etiquetados). Dependiendo de cómo se etiqueten los datos, el aprendizaje semisupervisado se puede dividir en las siguientes categorías:

- *El autoaprendizaje* es un aprendizaje semisupervisado que utiliza sus propias predicciones para enseñarse a sí mismo.
- *El co-entrenamiento* es un aprendizaje débilmente semisupervisado para datos de vista múltiple que utiliza la configuración de co-entrenamiento y usa sus propias predicciones para aprender a sí mismo.
- *El aprendizaje activo* es un aprendizaje semisupervisado en el que el alumno tiene algún papel activo o participativo en la determinación de qué puntos de datos pedirá que un experto o profesor los etiquete.

- *Aprendizaje por refuerzo*: los datos de entrenamiento (en forma de recompensas y castigos) se brindan solo como retroalimentación a un agente de inteligencia artificial en un entorno dinámico. Esta retroalimentación entre el sistema de aprendizaje y la experiencia de interacción es útil para mejorar el desempeño en la tarea que se está aprendiendo. El aprendizaje automático basado en la retroalimentación de datos se denomina aprendizaje por refuerzo. *Q-learning* es un aprendizaje de refuerzo sin modelo popular y aprende una función de recompensa o castigo (funciones de valor de acción, simplemente llamada función *Q*)
- *Transferencia de aprendizaje*: en muchas aplicaciones del mundo real, la distribución de datos cambia o los datos están desactualizados y, por lo tanto, es necesario aplicar transferencia de aprendizaje para considerar la transferencia de conocimiento desde el dominio de origen al dominio de destino. El aprendizaje de transferencia incluye, entre otros, el aprendizaje de transferencia inductivo, el aprendizaje de transferencia transductivo, el aprendizaje de transferencia no supervisado, el aprendizaje multitarea, el aprendizaje de transferencia autodidacta, la adaptación de dominio y la EigenTransfer.

(2) *Aprendizaje automático de gráficas* es un aprendizaje de datos estructurados irregulares o no euclidianos, y aprende la estructura de un gráfica, también llamado construcción de gráfica, a partir de muestras de entrenamiento en casos de aprendizaje semi-supervisado y no supervisado.

La clasificación anterior de aprendizaje automático se puede representar vívidamente mediante un árbol de aprendizaje automático, como se muestra enseguida:



El árbol de aprendizaje automático se llama así porque la figura 6.1 parece un árbol después de girarlo 90° hacia la izquierda.

Hay dos tareas básicas del aprendizaje automático: clasificación (para datos discretos) y predicción (para datos continuos).

El aprendizaje profundo es aprender la ley interna y el nivel de representación de los datos de muestra. La información en diferentes niveles de jerarquía obtenida en el proceso de aprendizaje es de gran ayuda para la interpretación de datos como texto, imagen y voz. El aprendizaje profundo es un algoritmo de aprendizaje automático complejo, que ha logrado resultados mucho mejores en el reconocimiento de voz e imágenes que las tecnologías relacionadas anteriores.

Limitado al espacio, este texto no analizará el aprendizaje profundo (deep learning), sino que se centrará únicamente en el aprendizaje supervisado, el aprendizaje no supervisado, el aprendizaje por refuerzo y el aprendizaje por transferencia.

Antes de tratar el aprendizaje automático en detalle, es necesario comenzar con el conocimiento de preparación del aprendizaje automático: sus problemas de optimización, los algoritmos de minimización de mayoría y cómo impulsar un algoritmo de aprendizaje débil a un algoritmo de aprendizaje fuerte.

Teoría básica del aprendizaje automático

El pionero del aprendizaje automático, Arthur Samuel, definió el aprendizaje automático como un «campo de estudio que brinda a las computadoras la capacidad de aprender sin ser programadas explícitamente».

2.1. Máquina de aprendizaje

Considere una máquina de aprendizaje cuya tarea es aprender una asignación $x_i \rightarrow y_i$. La máquina en realidad está definida por un conjunto de posibles asignaciones $x \rightarrow f(x, \alpha)$ con las funciones $f(x, \alpha)$ etiquetadas por el vector de parámetros ajustables α .

Por lo general, se supone que la máquina es determinista: para un vector de entrada dado x y la elección de α , siempre dará la misma salida $f(x, \alpha)$. Una elección particular de α genera una "máquina entrenada" y, por lo tanto, una red neuronal con arquitectura fija y que contiene α correspondientes a los pesos y sesgos se denomina máquina de aprendizaje en este sentido.

La expectativa del error de prueba para una máquina entrenada se define como

$$R(\alpha) = \int \frac{1}{2} |y - f(x, \alpha)| dP(x, y)$$

donde $P(x, \alpha)$ es una distribución de probabilidad acumulada desconocida a partir de la cual se extraen los datos x e y , es decir, se supone que los datos se distribuyen de forma independiente e idéntica (*i.i.d.*). Cuando existe una densidad de distribuciones de probabilidad acumulada, $P(x, y)$, $dP(x, y)$ puede escribirse como $p(x, y) dx dy$

La cantidad $R(\alpha)$ se denomina riesgo esperado, o simplemente el riesgo de elección de α . El «riesgo empírico» $R_{emp}(\alpha)$ se define como la tasa de error media medida en el conjunto de entrenamiento dado:

$$R_{emp}(\alpha) = \frac{1}{2N} \sum_{i=1}^N |y_i - f(x_i, \alpha)|$$

donde no aparece distribución de probabilidad. La cantidad $\frac{1}{2} |y_i - f(x_i, \alpha)|$ se llama la pérdida. $R_{emp}(\alpha)$ es un número fijo para una elección particular de α y para un conjunto de entrenamiento particular x_i, y_i .

Otro nombre para una familia de funciones $f(x, \alpha)$ es "máquina de aprendizaje". El diseño de una buena máquina de aprendizaje depende de la elección de α . Un método popular para construir α es la minimización empírica del riesgo (**MRE**). Por **MRE**, queremos decir que $R_{emp}(\alpha)$ se minimiza sobre todas las opciones posibles de α , lo que da como resultado un riesgo $R(\alpha^*)$ que está cerca de su mínimo. **La minimización de diferentes funciones de pérdida da como resultado diferentes métodos de aprendizaje automático.** En realidad, la mayoría de los métodos de aprendizaje automático deberían tener tres fases en lugar de dos: entrenamiento, validación y prueba. Una vez que se completa el entrenamiento, puede haber varios modelos disponibles (por ejemplo, redes neuronales artificiales), y uno debe decidir cuál tendrá una buena estimación del error que logrará en un conjunto de prueba. Para este fin, debe haber un tercer conjunto de datos separado: el conjunto de datos de validación.

2.2. Métodos de aprendizaje automático

Existen múltiples enfoques de aprendizaje automático disponibles para modelar los datos del problema subyacente. Estos enfoques se pueden clasificar en las siguientes categorías.

■ Métodos de Aprendizaje Automático basados en Estructura de Red.

- *Redes neuronales artificiales (RNA)*: las RNAs están inspiradas en el cerebro y se componen de neuronas artificiales interconectadas capaces de realizar ciertos cálculos en sus entradas. Los datos de entrada activan las neuronas en la primera capa de la red cuya salida es la entrada a la segunda capa de neuronas en la red. De manera similar, cada capa pasa su salida a la siguiente capa y la última capa genera el resultado. Las capas entre las capas de entrada y salida se denominan capas ocultas. Cuando se utiliza una RNA como clasificador, la capa de salida genera la categoría de clasificación final.
- *Red bayesiana*: Una red bayesiana es un modelo gráfico probabilístico que representa las variables y las relaciones entre ellas. La red se construye con nodos como las variables aleatorias discretas o continuas y aristas dirigidas como las relaciones entre ellas, estableciendo un grafo acíclico dirigido. Cada nodo mantiene los estados de la variable aleatoria y la forma de probabilidad condicional. Las redes bayesianas se construyen utilizando conocimiento experto o algoritmos eficientes que realizan inferencias.

■ Métodos de Aprendizaje Automático basados en Análisis Estadístico.

- *Reglas de asociación*: el concepto de reglas de asociación se popularizó particularmente debido a un artículo de Agrawal et al. en 1993. El objetivo de la minería de reglas de asociación es descubrir reglas de asociación previamente desconocidas a partir de los datos. Una regla de asociación describe una relación $x \Rightarrow y$ entre un conjunto de elementos x y un único elemento y . Las reglas de asociación tienen dos métricas que indican con qué frecuencia se produce una relación determinada en los datos: el soporte es una indicación de la frecuencia con la que aparece el conjunto de elementos en el conjunto de datos y la confianza es una indicación de la frecuencia con la que se ha determinado que la regla es verdadera.
- *Agrupamiento*: el agrupamiento es un conjunto de técnicas para encontrar patrones en datos no etiquetados de alta dimensión. Es un enfoque de descubrimiento de patrones no supervisado en el que los datos se agrupan en función de alguna medida de similitud.
- *Aprendizaje en conjunto*: los algoritmos de aprendizaje supervisado, en general, buscan en el espacio de hipótesis para determinar la hipótesis correcta que hará buenas predicciones para un problema determinado. Aunque pueden existir buenas hipótesis, puede ser difícil encontrar una. Los métodos de conjunto combinan múltiples algoritmos de aprendizaje para obtener un mejor rendimiento predictivo que los algoritmos de aprendizaje constituyentes solos. A menudo, los métodos de conjunto utilizan múltiples alumnos débiles para construir un alumno fuerte.
- *Modelos ocultos de Markov (MOM)*: un MOM es un modelo estadístico de Markov en el que se supone que el sistema que se modela es un proceso de Markov con estados no observados (es decir, ocultos). El principal desafío es determinar los parámetros ocultos a partir de los parámetros observables. Los estados de un MOM representan condiciones no observables que se están modelando. Al tener diferentes distribuciones de probabilidad de salida en cada estado y permitir que el sistema cambie de estado con el tiempo, el modelo es capaz de representar secuencias no estacionarias.
- *Aprendizaje inductivo*: la deducción y la inducción son dos técnicas básicas para inferir información a partir de datos. El aprendizaje inductivo es el aprendizaje supervisado tradicional y tiene como objetivo aprender un modelo a partir de ejemplos etiquetados y tratar de predecir las etiquetas de ejemplos que no hemos visto o que no conocemos. En el aprendizaje inductivo, a partir de observaciones específicas se comienza a detectar patrones y regularidades, se formulan algunas hipótesis tentativas para explorar y finalmente se termina desarrollando algunas conclusiones generales o teorías. Varios algoritmos de ML son inductivos, pero por aprendizaje inductivo generalmente nos referimos a «poda incremental repetida para producir reducción de errores»(PIRPRE) y el algoritmo cuasióptimo (AC).

- *Ingenuo Bayes*: Es bien sabido que un clasificador bayesiano sorprendentemente simple con fuertes suposiciones de independencia entre características, llamado ingenuo Bayes, es competitivo con clasificadores de última generación como C4.5. En general, se supone que las características de entrada son independientes, mientras que en la práctica esto rara vez es cierto. Los clasificadores Ingenuos Bayes pueden manejar un número arbitrario de características independientes, ya sean continuas o categóricas, al reducir una tarea de estimación de densidad de alta dimensión a una estimación de densidad de kernel unidimensional, bajo el supuesto de que las características son independientes. Aunque el clasificador Ingenuo Bayes tiene algunas limitaciones, es un clasificador óptimo si las características son condicionalmente independientes dada la verdadera clase. Una de las mayores ventajas del clasificador Ingenuo Bayes es que es un algoritmo en línea y su entrenamiento se puede completar en tiempo lineal.

■ **Métodos de Aprendizaje Automático basados en Evolución.**

- *Computación evolutiva*: en informática, la computación evolutiva es una familia de algoritmos para la optimización global inspirada en la evolución biológica y el subcampo de inteligencia artificial y computación blanda que estudia estos algoritmos. El término computación evolutiva abarca algoritmos genéticos (AG), programación genética (PG), estrategias de evolución, optimización de enjambres de partículas, optimización de colonias de hormigas y sistemas inmunológicos artificiales, que son los temas de Cálculos evolutivos.

2.3. Rendimiento esperado de los algoritmos de aprendizaje automático

Se espera que un algoritmo de aprendizaje automático tenga el siguiente rendimiento esperado.

- *Escalabilidad*: este parámetro se puede definir como la capacidad de un algoritmo para manejar un aumento en su escala, como alimentar más datos al sistema, agregar más funciones a los datos de entrada o agregar más capas en una red neuronal, sin aumentando ilimitadamente su complejidad.
- *Tiempo de entrenamiento*: esta es la cantidad de tiempo que tarda un algoritmo de aprendizaje automático en estar completamente entrenado y formar la capacidad de hacer sus predicciones.
- *Tiempo de respuesta*: este parámetro está relacionado con la agilidad de un sistema de aprendizaje automático y representa el tiempo que tarda un algoritmo, después de haber sido entrenado, para hacer una predicción para la función de redes auto-organizadas (RAO) deseada.
- *Datos de entrenamiento*: esta métrica del algoritmo de aprendizaje automático es la cantidad y el tipo de datos de entrenamiento que necesita un algoritmo. Los algoritmos compatibles con más datos de entrenamiento suelen tener una mayor precisión, pero también tardan más en entrenarse.
- *Complejidad*: La complejidad de un sistema se puede definir como la cantidad de operaciones matemáticas que realiza para lograr una solución deseada. Este parámetro puede determinar si ciertos algoritmos son más adecuados para implementarse en el lado del usuario.
- *Precisión*: se espera que las redes futuras sean mucho más inteligentes y rápidas, lo que permitirá tipos de aplicaciones y requisitos de usuario muy diferentes. El despliegue de algoritmos que tengan una alta precisión es fundamental para garantizar una buena operatividad de ciertas funciones de red autoorganizadas.
- *Tiempo de Convergencia*: Esta métrica de un algoritmo, a diferencia del tiempo de respuesta, se relaciona con qué tan rápido acepta que la solución encontrada para ese problema en particular es la solución óptima en ese momento.
- *Confiabilidad de convergencia*: este parámetro representa la susceptibilidad de algún algoritmo a quedarse atascado en los mínimos locales y cómo las condiciones iniciales pueden afectar su desempeño.

Clasificación y regresión

En esencia, el aprendizaje automático es aprender las muestras de entrenamiento dadas para resolver uno de dos problemas básicos: regresión (para salidas continuas) o clasificación (para salidas discretas). La clasificación está estrechamente relacionada con el reconocimiento de patrones, su objetivo es diseñar un clasificador pensado en aprender un conjunto de datos de entrada de «entrenamiento» para realizar el reconocimiento o la clasificación de muestras desconocidas. Por regresión, significa diseñar un regresor o predictor basado en los resultados de aprendizaje automático de un conjunto de datos de entrenamiento para hacer una predicción para muestras continuas desconocidas.

3.1. Reconocimiento y clasificación de patrones

El reconocimiento de patrones es el campo dedicado al estudio de métodos diseñados para categorizar datos en distintas clases. Por Watanabe, un patrón se define «como opuesto a un caos; es una entidad, vagamente definida, a la que se le podría dar un nombre.» El reconocimiento de patrones se aplica ampliamente en el reconocimiento de la biometría humana (como el rostro de una persona, la huella dactilar, el iris) y varios objetivos (como aeronaves, barcos). En estas aplicaciones, la extracción de las características de los objetos es crucial. Por ejemplo, cuando un objetivo se considera un sistema lineal, el parámetro del objetivo es una característica de la señal del objetivo.

Dado un patrón, su reconocimiento/clasificación puede consistir en una de las siguientes dos tareas:

- *Clasificación supervisada* (por ejemplo; análisis discriminante) en la que el patrón de entrada se identifica como miembro de una clase predefinida;
- *Clasificación no supervisada* (por ejemplo; agrupamiento) en la que el patrón se asigna a una clase hasta ahora desconocida.

Las tareas de reconocimiento/clasificación de patrones se suelen dividir en cuatro bloques distintos:

1. Representación de datos (adquisición y preprocesamiento).
2. Selección o extracción de características.
3. Agrupamiento.
4. Clasificación.

Los cuatro enfoques más conocidos para el reconocimiento de patrones son: (1) coincidencia de plantillas, (2) clasificación estadística, (3) coincidencia sintáctica o estructural y (4) redes neuronales.

Sea (TD, ϑ) una tabla de datos y $\{x_1, \dots, x_N\}$ el conjunto de vectores de datos subyacentes de TD , es decir, x_i es el i -ésimo vector de datos; donde $N = |\text{Fila}(TD)|$ y $n = |\text{Col}(TD)|$.

Representación de datos

La representación de datos es principalmente específica del problema. En álgebra matricial, procesamiento de señales, reconocimiento de patrones, etc., el preprocesamiento de datos generalmente implica la normalización de media cero (es decir, centrado de datos): para cada vector de datos x_i ,

$$x_{i,j}^{cent} = x_{i,j} - \bar{x}_i, \text{ donde } \bar{x}_i = \frac{1}{n} \sum_{j=1}^n x_{i,j}, \text{ y } i = 1, \dots, N, j = 1, \dots, n \quad (3.1)$$

para eliminar los componentes inútiles de corriente directa (CD) en los datos, donde $x_{i,j} = \vartheta(f_i, c_j)$, es decir, es la j -ésima entrada del vector x_i . El centrado de datos también se refiere al significado cero de datos. Para evitar cambios drásticos en la amplitud de los datos, también es necesario escalar los datos de entrada a rangos similares:

$$x_{i,j}^{esc} = \frac{x_{i,j}}{s_i}, \text{ donde } s_i = \sqrt{\sum_{j=1}^n x_{i,j}^2}, \text{ y } i = 1, \dots, N, j = 1, \dots, n \quad (3.2)$$

Este preprocesamiento se denomina escalado de datos.

Selección o extracción de características

En muchas aplicaciones, los vectores de datos deben convertirse en vectores de baja dimensión a través de algún método de transformación/procesamiento. Estos vectores de baja dimensión se denominan vectores de patrón o vectores de características debido al hecho de que extraen las características de los vectores de datos originales y se utilizan directamente para la clasificación y agrupación de patrones. Por ejemplo, los colores de las nubes y los parámetros de los tonos de voz son patrones o vectores de características en el pronóstico del tiempo y la clasificación de voz, respectivamente.

Debido a la alta dimensionalidad, los vectores de datos originales no están disponibles directamente para el reconocimiento de patrones. Por lo tanto, uno debe tratar de encontrar características invariantes en los datos que describan las diferencias en las clases lo mejor posible.

Según el conjunto de entrenamiento dado, la selección de funciones se puede dividir en métodos supervisados y no supervisados.

- *Selección supervisada de características*: Sea $(X_i, Y_i) = \{(x_1, y_1), \dots, (x_N, y_N)\}$ un conjunto etiquetado de datos de (TD, ϑ) , donde $x_i \in \mathbb{R}^n$ es el i -ésimo vector de datos y $y_i = k$ indica la k -ésima clase a la que pertenece el vector de datos x_i , donde $k \in \{1, \dots, M\}$ es la etiqueta de M clases de destino. El reconocimiento de patrones utiliza aprendizaje automático supervisado. Sea l_p el número de $k = p$, y $l_1 + \dots + l_M = N$, podemos construir la matriz de datos en la p -ésima clase de objetivos:

$$x^{(p)} = [x_1^{(p)}, \dots, x_{l_p}^{(p)}]^T \in \mathbb{R}^{N \times l_p}, p = 1, \dots, M.$$

Por lo tanto, podemos seleccionar vectores de características en los datos utilizando métodos de álgebra matricial como el análisis de componentes principales, etc. Esta selección, como aprendizaje automático supervisado, también se conoce como selección de características o reducción de dimensionalidad, ya que la dimensión del vector de características es mucho menor que la dimensión de los vectores de datos originales.

- *Selección de características no supervisada*: cuando los datos no están etiquetados, es decir, solo los vectores de datos x_1, \dots, x_N , el reconocimiento de patrones es aprendizaje automático no supervisado, que es más difícil que el reconocimiento de patrones supervisado. En este caso, necesitamos usar aplicación, asociación o procesamiento de señales para transformar los datos en características invariantes como la transformada de Fourier de tiempo corto, biespectro, wavelet, etc.

La selección de características consiste en seleccionar las características más significativas de los datos e intentar reducir la dimensionalidad (es decir, la cantidad de características) para los pasos restantes de la tarea.

Discutiremos las extracciones de características supervisadas y no supervisadas más adelante en detalle.

Agrupación

Sea w el vector de pesos de un grupo cuyo objetivo es que los patrones dentro de un grupo sean más similares entre sí que los patrones que pertenecen a otros grupos. Los métodos de agrupamiento se utilizan para encontrar el mapeo real entre patrones y etiquetas (u objetivos).

Esta categorización tiene dos métodos de aprendizaje automático: aprendizaje supervisado (que hace un etiquetado distinto de los datos) y aprendizaje no supervisado (división de los datos en clases), o una combinación de más de una de estas tareas.

Las tres etapas anteriores (representación de datos, selección de características y agrupación) pertenecen a la fase de entrenamiento. Teniendo tal grupo, uno puede hacer una clasificación en la siguiente fase de prueba.

Clasificación

La clasificación es una de las tareas de toma de decisiones más frecuentes de la actividad humana. Un problema de clasificación se produce cuando es necesario asignar un objeto a un grupo o clases predefinidos relacionados con ese objeto después de agrupar una serie de grupos o clases en función de un conjunto de atributos observados. En otras palabras, uno necesita decidir a qué clase entre un número de clases pertenece una muestra de prueba dada. Este paso se denomina fase de prueba, en comparación con la etapa de entrenamiento anterior. Claramente, esta es una forma de aprendizaje supervisado.

3.2. Regresión

En el modelado estadístico y el aprendizaje automático, la regresión es ajustar un proceso estadístico para estimar las relaciones entre variables. Se centra en la relación entre una variable dependiente x y una o más variables independientes (denominadas “predictores”) β . Más específicamente, la regresión es para analizar cómo cambia el valor típico de la variable dependiente cuando se varía cualquiera de las variables independientes, mientras que las otras variables independientes son fijas.

El aprendizaje automático se usa ampliamente para la predicción y el pronóstico, donde su uso tiene una superposición sustancial con el campo del análisis de regresión estadística.

Considere el modelo de regresión múltiple más general con p variables independientes:

$$y_i = \beta_1 x_{i1} + \beta_2 x_{i2} + \cdots + \beta_p x_{ip} + \epsilon_i,$$

Lista de algoritmos

Bibliografía

- [1] Mitchell, T.M. (1997). *Machine Learning*. London, vol. 45. McGraw Hill, Burr Ridge
- [2] Settles, B., Craven, M., Friedland, L. (2008). *Active learning with real annotation costs*. In: *Proceedings of the NIPS Workshop on Cost-Sensitive Learning*, pp. 1–10