TFG-PREDICCION-ENERGIA-EOLICA

josebenjumedarubio

September 2020

1 Introduction

Notas sobre el libro ML for absolute beginners. Cosas que aprender: qué es un problema supervisado y uno no supervisado, qué es un problema de clasificación y uno de regresión y qué es entrenar y predecir un modelo.

Machine learning es un método englobado dentro de "Data Science" o "Ciencia de Datos", donde se busca que una máquina realice operaciones cognitivas, que son las actividades, inicialmente propias del ser humano, están relacionadas con el procesamiento de la información mediante el uso de la memoria, la atención, la percepción, la creatividad y el pensamiento abstracto o analógico.

Para conseguir que una máquina aprenda a realizar dichas tareas, se crea un modelo, que "aprende" con un conjunto de datos para luego llevar a cabo una acción, como una predicción o una respuesta que no estaba incluida en el conjunto de datos de que se disponía.

Es importante también que se hace un reparto del conjunto de datos en dos subconjuntos: conjunto de entrenamiento y conjunto de test. Con el conjunto de entrenamiento se perfila el modelo y se le añaden modificaciones, y luego se comprueba sobre el conjunto de test que funciona correctamente. Si no se obtienen los resultados deseados se cambia el modelo y se vuelve a empezar. Una vez satisfecho con los resultados, el modelo está listo para actuar con nuevos datos.

Para los modelos, hay desarrollados más de 700 algoritmos para realizar estadísticas sobre los datos, y cual de ellos o qué combinación de ellos utilizar es uno de los principales retos a superar para hacer aprendizaje automático. Estos métodos se clasifican en tres categorías generales: aprendizaje automático supervisado, no supervisado y reforzado.

Aprendizaje automático supervisado. Lo primero es decir que los datos se estructuran como un vector X que contiene todas las características excepto la característica considerara valor o resultado, y un vector de un solo elemento Y, que contiene éste último valor o resultado, que se considera dependiente de X. Con esto, el modelo intenta encontrar la función que relaciona esos valores de X con los de Y, para poder obtener los valores de Y para nuevas X que no existían en los datos de los que se disponía inicialmente. El modelo, una vez creado en su versión inicial, se entrena y mejora con los datos del conjunto de entrenamiento, y luego se pone a prueba con los datos del conjunto de test. Algunos ejemplos de algoritmos de aprendizaje automático supervisado son análisis de regresión, árboles de decisión, k vecinos más próximos, redes neuronales y máquinas de vectores de soporte.

Aprendizaje automático no supervisado. En este caso, no todos los atributos (coordenadas del vector X) serán tenidos en cuenta. La máquina, o el modelo, tendrá que encontrar los más útiles y aquellos que escondan relaciones con la variable Y que funcionen mejor. Además, el modelo crea etiquetas para agrupar los datos que sean similares en algún aspecto significativo. La ventaja principal del aprendizaje automático no supervisado es que agrupa los datos según una clasificación "inventada" por la máquina, y que puede haber pasado inicialmente desapercibida para la persona que la construye.

Duda: hay un ejemplo en el que habla de como DataVisor detecta fraudes y hackers mediante el uso de aprendizaje automático no supervisado, analizando patrones de miles de millones de cuentas, y encontrando conexiones entre usuarios, y luego siendo capaz de encontrar cuentas falsas a partir de ver cuales están relacionadas con otra que ha sido probada falsa primero. Creo que aquí en lugar de indicarle el programador qué atributos indican fraude, se deja que el modelo los encuentre solo. Párrafo que no dice ninguna diferencia con aprendizaje automático supervisado: DataVisor and other anti-fraud solution providers therefore leverage unsupervised learning to address the limitations of supervised learning by analyzing patterns across hundreds of millions of accounts and identifying suspicious connections between users—without knowing the actual category of future attacks.

Finalmente: Ahora mismo lo único que sé de esto es que creo que se dan muchos datos de cada usuario, es decir, de cada observación creo que se llama, y se deja que el modelo encuentre qué combinaciones de atributos son necesarias para que el valor sea positivo, y también, independientemente del valor que tenga cada observación, encuentra relaciones entre unas observaciones y otras, para que, en el momento en que una de ellas sea positiva, se prevea que las otras también tienen posibilidades de ser positivas.

Aprendizaje automático con refuerzo. Vale. Encuentro bastante escueto lo que viene. He entendido: El modelo varía constantemente, según va apren-

diendo. Hay una Q, que inicialmente tiene un valor de 0, y el modelo elige opciones que provocan cosas buenas o cosas malas. Las cosas buenas aumentan el valor de Q, y las malas lo decrementan. El modelo recuerda en qué situación ha tomado qué decisión, y si ha sido buena, en el futuro la repetirá, y si ha sido mala, la evitará.

Librerías y herramientas útiles: sección The ML Toolbox, pag 22, y especialmente a partir de la pag 25.

Scrubbing (depuración) es un proceso que se hace cuando el conjunto de datos que tenemos es muy grande, pues también es muy grande el ruido, y consiste en deshacernos de los datos que no nos aportan información útil, como casos muy excepcionales que es difícil que se repitan, por ejemplo.

GPU: originalmente surgieron para procesar imágenes muy pesadas muy rápidamente en videojuegos, pero posteriormente se descubrió que se podían usar también para calcular todas las posibilidades en cascada de una red neuronal, obteniendo un tiempo de un día en lo que con una CPU convencional se tardaba varias semanas. La GPU es unidad de procesamiento gráfico, y dice ahí que de lo que estamos hablando es además un "specialized parallel computing chip" un chip de procesamiento especializado en paralelo. Obtiene una ventaja muy grande en el número de operaciones de coma flotante frente a la CPU.

Algoritmos avanzados. Resulta que no explica ninguno. Dice que se usarán sobretodo redes neuronales para empezar.

Depuración de datos (data scrubbing): modifying and sometimes removing incomplete, incorrectly formatted, irrelevant or duplicated data. Suele llevar mucho tiempo y esfuerzo. Es importante eliminar atributos que no aporten información, o que estén duplicados, y reducir varios atributos a uno también cuando sea posible. También se pueden unir observaciones (rows) pero hay que tener en cuenta que, mientras que al diferir en un campo con valor numérico, se pueden unir mediante la media, cuando son atributos que no son numéricos, como Japón y Corea del Sur, ya es más difícil buscar algo en común, y más aún sin perder mucha información.

Hay una técnica que se llama One-hot encoding, que consiste en coger los posibles valores de un atributo, convertirlos en atributos y ponerle a cada fila, para el valor que tenía, un 1, y para los que no, un 0. Por ejemplo, para un atributo que es color de pelo que tiene valores rubio moreno o castaño, se crean tres atributos que son rubio, moreno y castaño, y al que lo tenía castaño se le pone 0 en los primeros y 1 en el tercero.

Binning: consiste en perder información cierto grado de información al transformar una cantidad numérica a categorías, por ejemplo cuando estudiando el valor de una casa tenemos las medidas de la pista de tenis, pero lo que nos interesa no es esto sino si tiene pista de tenis o no. Cambiamos las medidas por true/false o por 1/0 (one-hot).

Missing data: Cuando falte información, podemos rellenar ese atributo con la media o la moda.

Setting up your data – preparando los datos. Es importante, antes de hacer la división entre conjunto de entrenamiento y de test, barajar las observaciones para evitar que pueda darse un sesgo que se tenga en cuenta en el entrenamiento del modelo pero que luego no se de en la realidad. Porcentajes: 80 o 70% de entrenamiento y el resto de test.

En el caso de aprendizaje supervisado se le da el conjunto de entrenamiento y luego se prueba con el test. El error se puede medir como la media total de error. Si es muy grande, puede ser porque no hayamos barajado bien las observaciones o porque haya que revisar los parámetros de nuestro modelo, que afectan directamente a los patrones que el modelo encuentra.

Cross validation – validación en cruz. Consiste en realizar el proceso de entrenamiento y test con distintos conjuntos sobre los mismos datos, más o menos veces, y luego combinando los modelos obtenidos en uno solo. Esto se llama k-fold validation technique, pues se repite el proceso k veces, haciendo grupos de información de #(observaciones totales)/k, y utilizando todos los grupos al menos una vez como parte del conjunto de entrenamiento y al menos una vez como parte del conjunto de test. De esta manera se evitan errores como el sobreajuste (overfitting).

¿Qué cantidad de datos necesito? Cuantas más combinaciones de los valores de los atributos tengamos, más sabrá el modelo cómo afecta el valor de cada atributo al resultado o variable dependiente Y. En general, un buen número suele ser 10 veces la cantidad de atributos. Es decir, si tenemos 3 atributos, 30 observaciones.

El valor de la y se llama etiqueta motivación/introducción del problema que quiero solucionar explicar el estado del arte, las técnicas que voy a utilizar y qué cosas hace la gente a día de hoy para solucionarlo. Qué es una red neuronal y tal Qué técnicas voy a utilizar yo. Qué resultados obtengo y eso.

2 Análisis de regresión

2.1 Regresión lineal

Consiste en aproximar las etiquetas a una línea, es decir, hacer que el valor de la etiqueta dependa linealmente del valor del vector X, minimizando la distancia entre el valor real de la etiqueta y el que le asocia la función lineal que hallemos. Hay una fórmula para hallar esta recta:

$$a = \frac{(\sum y)(\sum x^2) - (\sum x)(\sum xy)}{n\sum x^2 - (\sum x)^2}$$

$$b = \frac{n\sum xy - (\sum x)(\sum y)}{n\sum x^2 - (\sum x)^2}$$

2.2 Regresión logística

La regresión logística es un método para resolver problemas de clasificación, en los que las observaciones pueden clasificarse en un tipo u otro (siempre una cantidad discreta de tipos). Para esto, se suele utilizar una función sigmoidea, que nos da una curva con forma de S con la que podemos mandar cualquier dominio al intervalo (0, 1). Cuando hay solo dos posibilidades para el tipo de la observación, se llama regresión logística binomial, y cuando hay más, multinomial. **No entiendo aquí qué tiene que ver la función sigmoidea, no entiendo qué predice si siempre tiene la misma forma.**

3 Análisis de grupos/Agrupamiento/Clustering

Es una técnica de clasificación. Tiene un parámetro K que es la cantidad de grupos que va a haber, y el método que utiliza para encontrarlos es asignar K núcleos, inicialmente coincidiendo con observaciones, y asigna a cada observación el grupo cuyo núcleo esté más cerca. Cuando termina, cambia el núcleo por la media de las observaciones de cada grupo, y repite la asignación de observaciones a grupos, tantas veces como haga falta hasta que no se produzca ningún cambio al modificar el núcleo. Además, esto se va haciendo para distintos valores de K, y para escoger el mejor, lo que se hace es una gráfica, que estará formada por segmentos, y nos quedamos con el valor que forme entre los dos segmentos que separa, el ángulo más agudo, es decir, la esquina más pronunciada. La diferencia con análisis de grupos jerárquico es que en análisis de grupos se busca la cantidad de núcleos indicada por K, mientras que en el jerárquico simplemente separa hasta el final, en grupos más pequeños según bajamos de nivel en el árbol.

El libro habla también de K-vecinos más próximos (K-Nearest Neighbours Clustering) como el algoritmo de análisis de grupos para aprendizaje automático supervisado. Aquí los grupos ya están hechos, y se clasifica cada nueva observación añadiéndose al grupo al que pertenezcan la mayoría de sus vecinos. Hay

un parámetro que es la cantidad de vecinos que miramos (siempre los más cercanos desde luego). Es bueno coger una cantidad de vecinos impar, para evitarnos el empate clásico. Desventajas: caro computacionalmente, especialmente si hay varias dimensiones. En este último caso se recomienda unir variables, utilizando por ejemplo Análisis de Componentes Principales (Principal Component Análisis).

4 Sesgo/parcialidad y varianza (Bias and variance)

Un problema clásico en aprendizaje automático es encontrar el punto justo de complejidad para nuestro modelo, de tal manera que tenga cierto error en el conjunto de entrenamiento a cambio de tener el menor error posible en el conjunto de test. Normalmente, si intentamos tener un error muy pequeño en el conjunto de entrenamiento, nuestro modelo se hace complejo, pues intenta en cuenta todas las variaciones, tendremos mayor error en el conjunto de test, puesto que muchas de las variaciones son circunstanciales, y no se repiten como el modelo espera. Una técnica para evitar esto es barajar las observaciones antes de separar entre conjunto de entrenamiento y de test, aunque en general siempre hay que variar los hiper-parámetros del modelo hasta conseguir buenos resultados. Otra forma más de intentar acertar en el punto justo de error en ambos conjuntos es la regularización, que consiste en introducir una penalización que se incrementa proporcionalmente a la complejidad del modelo. Por último, la validación en cruz (cross validation) también ayuda.