Módulo 7

7.1 Introducción al Big Data

Por **Fco. Javier García Castellano**.
Profesor Titular de Universidad. Departamento de Ciencias de Computación e Inteligencia Artificial (DECSAI). Universidad de Granada.

1. ¿QUÉ ES BIG DATA?

El Big Data, también llamados datos masivos, datos a gran escala o macrodatos, es un término de moda, que aparece incluso en los medios de comunicación generalistas, pero se utiliza sin saber qué es realmente.

Big Data hace referencia al problema que surge cuando no se puede trabajar con los datos usando las herramientas tradicionales de ciencia de datos que se han descrito hasta ahora en el curso. A mi me gusta poner el ejemplo de que Big Data es todo ese conjunto de datos tan grande que no caben en tu ordenador.

Lógicamente, los conjuntos de datos que hoy se pueden considerar Big Data, quizás en un futuro próximo (5 o 10 años), no se consideren como tales. ¿Por qué? Porque al aumentar la capacidad de los ordenadores, igual podemos trabajar con dichos datos en un ordenador portátil sin mucho problema.

También dependerá de los medios a nuestro alcance. Para mi un conjunto de datos de unos 300GB es ahora mismo Big Data. Sin embargo, en la nube podemos contratar un servidor de 96 procesadores con 384GB de Memoria RAM y 3TB de almacenamiento en SSD. En consecuencia, alguien que pueda contratar dicho servidor igual no lo considera un problema de Big Data.

Aunque me he centrado en el tamaño de los datos, el problema originado por el Big Data no sólo se refiere al almacenamiento, es bastante más complejo. Puedo tener datos almacenados, sin problemas de espacio, pero que el tiempo que me lleve procesarlos sea de años, o que el tamaño de los datos se duplique cada cierto tiempo, o que se generen tan rápido que no me de tiempo ni siquiera a almacenarlos.

En 2001, se definieron las 3Vs del Big Data que nos permiten tener más claro este concepto:

- Volumen: La cantidad de datos es un problema. Bien porque sean difíciles de almacenar o por que crecen rápidamente o de forma exponencial.
- Velocidad: La velocidad a la que se pueden procesar los datos. Puede que no seamos capaces de procesar los datos en un tiempo aceptable o que no seamos capaces de procesar al ritmo al que se generan.
- Variedad: Los datos pueden ser de diferentes tipos, como texto, imágenes o vídeos.

En 2013, se añadió una cuarta V:

- Veracidad: Tenemos incertidumbre asociada a los datos debido a datos inconsistentes, incompletos, ambiguos o erróneos.

Hay otros autores que usan más Vs (hasta 10) e incluso, en clave de humor, llegaron a definir las 42 Vs del Big Data.

2. BIG DATA Y BIOINFORMÁTICA

Las ciencias ómicas como la genómica, la proteómica, la transcriptómica, la metabolómica o la epigenómica, están generando un Volumen enorme de datos, a una gran Velocidad y con una gran Variedad de formatos, sobre todo cadenas de texto (p.e. nucleótidos) e imágenes. De hecho, actualmente, el cuello de botella en los laboratorios se centra en la gestión e interpretación de los datos que se están generando.

En efecto, las diferentes tecnologías introducidas en los últimos años en las ciencias ómicas nos permiten analizar un gran número de moléculas presentes en una única muestra, generando grandes volúmenes de información en muy poco tiempo. Por ejemplo, en genética, la secuenciación de alto rendimiento, también llamada de nueva generación, nos permite secuenciar miles de millones de fragmentos de ADN al mismo tiempo, generando una cantidad enorme de datos. Usando esta tecnología se han secuenciado gran cantidad de genomas de los más diversos organismos. Por ejemplo, a principios de Marzo de 2020 ya se habían secuenciado 253 genomas distintos del coronavirus SARS-CoV-2 responsable de la COVID-19.

Para secuenciar el genoma de un ser humano se generan unos 180 Gigabytes de datos, es por ello que cuando se realiza algún estudio con varios individuos la cantidad de datos supera fácilmente el Terabyte. En proyectos más ambiciosos, la cantidad de datos generada es

abrumadora, por ejemplo, en 2012 el proyecto de los 1000 genomas había generado 260 Terabytes de datos.

Se estima que la cantidad de datos proveniente de la secuenciación de alto rendimiento se triplica cada año. Es decir, si en el año 2018 se alcanzó el primer Exabyte en datos de secuenciación genómica, para el 2024 habremos llegado al Zettabyte. Otras estimaciones son bastante menos optimistas y calculan que para el 2025 "sólo" tendremos 40 Exabytes de datos de secuenciación genética (40 Exabytes ≈ 40.000 Petabytes ≈ 40.000.000 Terabytes). Si esto lo comparamos con las otras grandes fuentes de Big Data, como son los vídeos de YouTube o los mensajes de Twitter, parece que el problema real de Big Data lo tendremos en la genómica. Y eso sin contar con el resto de ciencias ómicas.

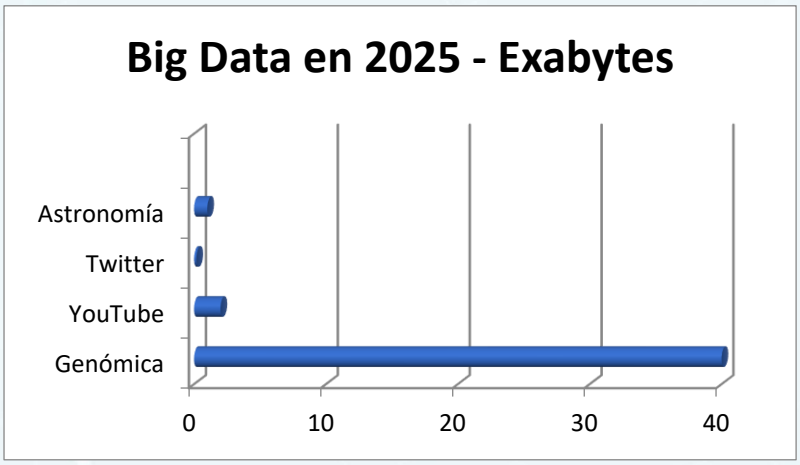

Figura 1. Estimación de almacenamiento de datos para 2025 en distintas áreas

Estas cantidades tan ingentes de datos, son a nivel mundial, claro está. Pero a nivel de laboratorio o de centro de investigación, un estudio donde se secuencien los genomas de los sujetos a investigar, fácilmente nos puede generar varios Terabytes de información. Y para poder guardar estos datos, ya tenemos un problema. Para analizar estas cantidades de datos, raramente se van a poder utilizar técnicas convencionales de ciencia de datos. Se tendrán que usar técnicas de Big Data.

3. HERRAMIENTAS DE BIG DATA. APACHE HADOOP Y APACHE SPARK.

Hay distintas herramientas de Big Data que nos permiten, por ejemplo, trabajar con flujos de datos en tiempo real, como Apache Storm o bases de datos NoSQL (Not only SQL) que son muy escalables y distribuidas, como Apache Cassandra o mongoDB. Pero nos vamos a centrar en las herramientas estándar como son Apache Hadoop y, la que más nos interesa para análisis de los datos, que es Apache Spark.

Apache Hadoop es el entorno estándar para trabajar con Big Data. Se utiliza para gestionar y/o procesar datos masivos, siendo muy escalable. Nos permite, guardar de forma distribuida los datos, usando un sistema de ficheros distribuido (HDFS, Hadoop Distributed File System) o también, nos permite procesar los datos de forma distribuida con la metodología Map-Reduce.

Figura 2. Logotipo Apache Hadoop.

Se basa en trabajar de forma distribuida con diferentes nodos, es decir, con un grupo interrelacionado de ordenadores. De forma distribuida significa que se reparte el trabajo y/o el almacenamiento de datos entre varios ordenadores. Es altamente escalable, lo que viene a decir que con más nodos funcionará mejor. También es tolerante a fallos, lo que quiere decir que, si falla alguno de los nodos, puede seguir funcionando sin problema.

Hadoop es un paquete muy amplio de programas y por esa razón a veces se le denomina ecosistema Hadoop. De entre, estos módulos podemos destacar:

- Hadoop Common: bibliotecas que usan los otros módulos de Hadoop.
- Hadoop Distributed File System (HDFS): un sistema de ficheros distribuido que nos permite guardar físicamente los datos en distintos nodos, pero que, al acceder a ellos, parecen estar en el ordenador local.
- Hadoop YARN: es la utilidad que se encarga de repartir y gestionar de forma distribuida las tareas por la red de ordenadores. Se encarga de gestionar los

recursos hardware de forma transparente.

- Hadoop MapReduce: es una metodología de trabajo para procesamiento en paralelo de grandes volúmenes de datos.

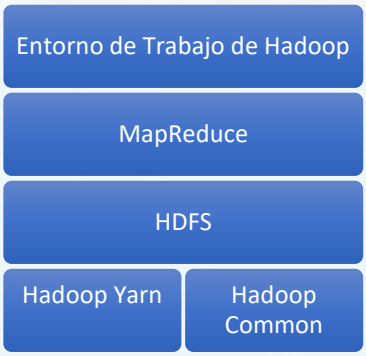

Figura 3. Esquema Apache Hadoop.

Apache Spark, es el sucesor de Hadoop MapReduce para procesamiento de datos, aunque se suelen usar conjuntamente. Es muy habitual usar HDFS para almacenar los datos y Spark para procesarlos. Apache Spark trabaja entre 10 y 100 veces más rápido que Hadoop.

Figura 4. Logotipo Apache Spark.

Apache Spark tiene distintos componentes:

- Spark SQL: nos permite usar SQL para acceder a datos estructurados.
- Spark Streaming: nos permite trabajar con flujos de datos en tiempo real. Algo que no permite Hadoop, pues MapReduce solo procesa datos en lotes.
- GraphX: nos permite realizar cálculos en paralelo usando un grafo.
- MLlib (Machine Learning library): Son un conjunto de algoritmos de Machine Learning y de procesamiento de datos con los cuales vamos a trabajar en este módulo.

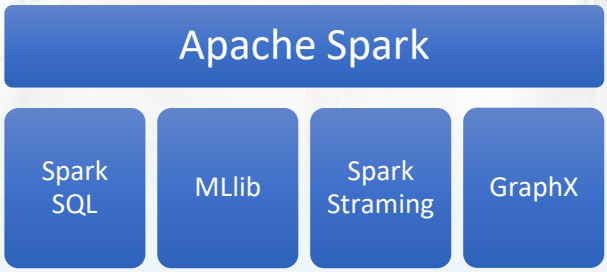

Figura 5. Esquema Apache Spark.

El aumento de la velocidad de Apache Spark respecto a la metodología MapReduce de Hadoop se base en dos características:

- **Grafo Acíclico Dirigido (*Directed Acyclic Graph*, DAG):** permite usar un grafo dirigido sin ciclos para controlar el flujo de datos. En la metodología MapReduce de Hadoop, se crea un grafo dirigido acíclico con dos estados predefinidos (*Map* y *Reduce*). En Spark se puede crear un grafo dirigido acíclico que puede tener cualquier número de estados. En Hadoop, cuando se utilizan varias etapas MapReduce, los resultados intermedios entre etapas se tienen que escribir a disco. Spark, en cambio, trabaja más en memoria, mejorando el rendimiento.

- **Conjunto de datos distribuido resistente (*Resilient Distributed Dataset* , RDD)**: es una colección distribuida inmutable de objetos con tolerancia a fallos. O dicho más claro, es la forma en que Spark guarda en memoria un conjunto de datos repartido por distintos nodos. En éstos se utiliza una evaluación perezosa, es decir, no se aplican las distintas operaciones hasta que no quede más remedio. Dichas operaciones se almacenan en un grafo dirigido acíclico y se optimiza su ejecución, pues hay operaciones más costosas que otras. Por ejemplo, supongamos que queremos hacer un filtro de dos tablas combinadas, es más rápido filtrar las tablas y luego unirlas, que combinar las tablas y luego filtrar.

4. METODOLOGÍA MAP REDUCE

El procesamiento de datos masivos se basa en la metodología MapReduce, creada por Google, e implementada inicialmente por Yahoo en el software libre Hadoop y cuyo código fue donado a la fundación Apache, quien se encarga actualmente de su desarrollo.

Es una metodología de programación, para el procesamiento en paralelo de grandes cantidades

de datos, en un grupo de ordenadores. En esta metodología el procesamiento está basado en dos funciones: mapeo (Map) y reducción (Reduce).

La función de Mapeo, recibe pares del tipo (clave, valor) y devuelve una lista de pares en un dominio diferente. La función de Reducción, junta conjuntos de pares con una misma clave.

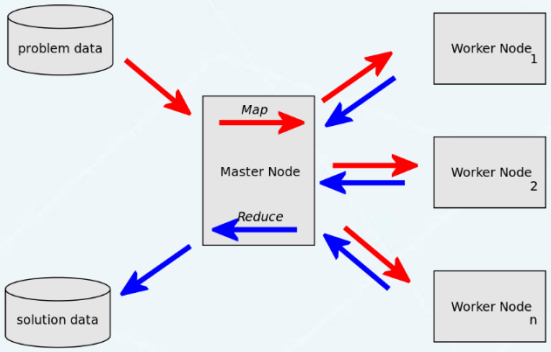

Figura 6. Reparto de trabajo en la metodología MapReduce

El proceso en sí, es algo más complicado que sólo dos pasos, pues hay que partir los datos, mapear, barajar/ordenar pares con igual clave, reducir y generar la salida.

Veamos un ejemplo. Supongamos que tenemos genes que se expresan (1) o no (0) y pacientes que pueden tener linfoma o no. Y supongamos que queremos saber la probabilidad de que, si un gen se activa, se tenga linfoma. En el mapeo nos quedaremos con los genes que se activan y si hay linfoma o no. En la reducción, calcularemos la probabilidad para cada gen.

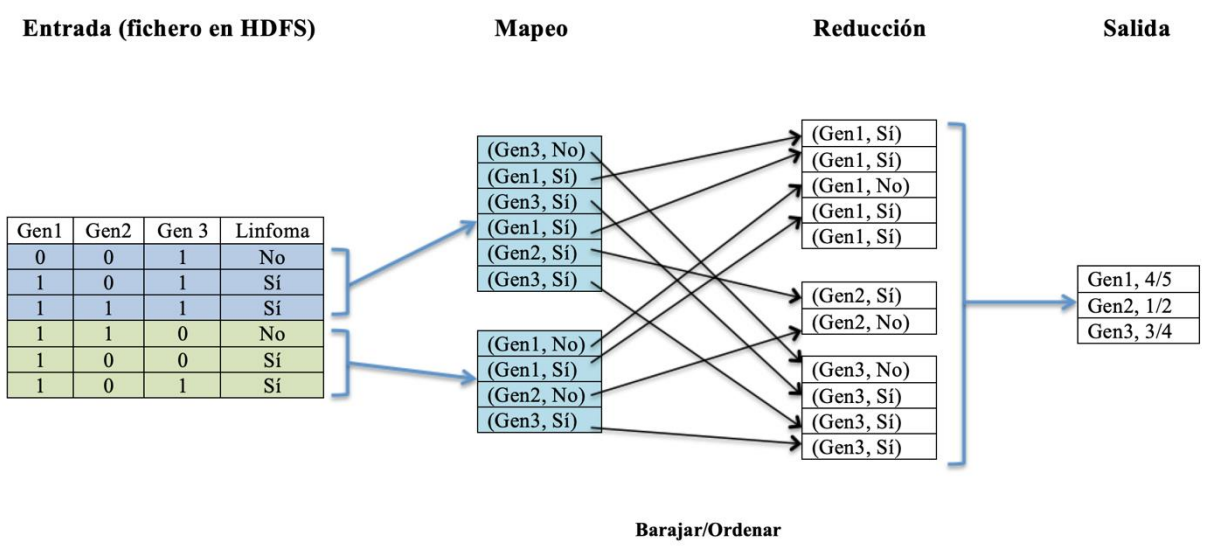

Figura 7. Ejemplo de cálculos usando la metodología MapReduce

No todos los problemas pueden ser abordados con MapReduce. No obstante, si el algoritmo puede ser paralelizado con MapReduce, es muy escalable el procesamiento de datos en múltiples nodos y, por tanto, ofrece un gran rendimiento en entornos distribuidos y con tolerancia a fallos.

Lo interesante de esta tecnología es que, para procesar un problema de Big Data, sólo tendremos que definir las funciones de Mapeo y Reducción. De la gestión de nodos, del procesamiento, de la gestión de errores y demás problemas se encargará Hadoop o Spark. No obstante, nosotros no nos preocuparemos de esto, pues utilizaremos los algoritmos ya programados en la biblioteca MLlib.

5. REFERENCIAS BIBLIOGRÁFICAS

- Laney, D. "3D Data Management: Controlling Data Volume, Velocity, and Variety". META group Inc., (2001). [Acceso 29 de mayo de 2020]. Disponible en https://blogs.gartner.com/doug-laney/files/2012/01/ad949-3D-Data-Management-Controlling-Data-Volume-Velocity-and-Variety.pdf .

- IBM. "The Four V's of Big Data". (2013). [Acceso 29 de mayo de 2020]. Disponible en: https://www.ibmbigdatahub.com/infographic/four-vs-big-data .

- Stephens ZD, Lee SY, Faghri F, Campbell RH, Zhai C, Efron MJ, et al. (2015) Big Data: Astronomical or Genomical?. PLoS Biol 13(7):e1002195. doi:10.1371/journal.pbio.1002195.

- Guo, R., Zhao, Y., Zou, Q., Fang, X. y Peng, S. (2018). Bioinformatics applications on Apache Spark. *GigaScience*, *7*(8), giy098.

- Apache Software Foundation. "Apache Hadoop Project" [Acceso 3 de junio de 2020]. Disponible en: https://hadoop.apache.org/ .

- Apache Software Foundation. "Apache Spark™ - Unified Analytics Engine for Big Data" [Acceso 3 de junio de 2020]. Disponible en: https://spark.apache.org/.

REFERENCIAS ADICIONALES

- Zaharia, M., Chowdhury, M., Das, T., Dave, A., Ma, J., Mccauley, M., Franklin, M.J., Shenker, S. y Stoica, I. (2012). Fast and interactive analytics over Hadoop data with Spark. *Usenix Login*, *37*(4), 45-51.

- Tipos de instancias de Amazon EC2. (2020). [Acceso 28 de mayo de 2020]. Disponible en: https://aws.amazon.com/es/ec2/instance-types/

- Tom Shafer. "The 42 V's of Big Data and Data Science". (2017). [Acceso 29 de mayo de 2020]. Disponible en: https://www.kdnuggets.com/2017/04/42-vs-big-data-data-science.html .

- Forster, P., Forster, L., Renfrew, C. y Forster, M. (2020). Phylogenetic network analysis of SARS-CoV-2 genomes. *Proceedings of the National Academy of Sciences*, *117*(17), 9241-9243.

- Clarke, L., Zheng-Bradley, X., Smith, R., Kulesha, E., Xiao, C., Toneva, I., Vaughan, B., Preuss, D., Leinonen, R., Shumway, M., Sherry, S., Flicek, P. y The 1000 Genomes Project Consortium (2012). The 1000 Genomes Project: data management and community access. *Nature methods*, *9*(5), 459-462.

- Mario Vega. "Procesamiento con MapReduce Part 1". (2017). MOOC Big Data. Universidad Politécnica de Madrid. [Acceso 8 de junio de 2020]. Disponible en: https://www.youtube.com/watch?v=kYsaCCmuYIg

- Santos, P. "Apache Spark VS Hadoop Map Reduce". (2019). [Acceso 29 de mayo de 2020]. Disponible en: https://openwebinars.net/blog/apache-spark-vs-hadoop-map-reduce/