

DATOS MASIVOS Y MINERÍA DE DATOS

Trabajo Final

Profesor:

* PHD David Diaz

Alumna:

* Andrea Yáñez

Marzo 2023

***Pregunta 1. Desarrollo***

* Explique con sus palabras cuáles son las principales diferencias y similitudes entre:
  + Una base de datos transaccional
  + Un Data Warehouse
  + Un Data Lake
  + Un Lake House
* Proponga un ejemplo y caso de uso para cada una de ellas

***Respuesta:***

Diferencias y similitudes.

Las bases de datos transaccionales surgieron de la necesidad de registrar todos los movimientos de las empresas, de forma rápida y confiable, se diseñan con id y llaves, la información no se repite, debe escribir rápido. Es lento obtener reportes porque involucra muchas tablas y conexiones entre ellas. Apoyan la operación,

Los Data Warehouse nacieron para poder obtener reportes agregados de información, para toda la empresa, deben leer rápido, por lo que la información esté redundante no es un problema. Apoyan la toma de decisiones. También hay almacenes mas pequeños llamados Datamart que contendrán los agregados de los distintos departamentos. Los datos son estructurados.

Los Data Lakes permiten almacenar distinta información de la empresa tanto estructurada como no estructurada, no está pensada tanto para acceso de usuarios finales, si no mas bien para obtener mayor conocimiento de la organización al disponer de estos datos para los científicos de datos dado que se genera hoy en día muchísima información de distinto tipo en las organizaciones lo que podría tener información valiosa para el negocio.

Un Lake House es una evolución de un Data Lakes que permitirá tener toda la información de la compañía tanto estructurada como no estructurada permitiendo accederla desde un mismo sistema aprovechando los últimos avances de la tecnología, la información pueden usarla tanto científicos de datos como analistas de datos para inteligencia empresarial dado que la información se puede accederá rápidamente y con una interfaz optimizada, todo en el mismo lugar.

Ejemplos y casos de uso

* + **Una base de datos transaccional:** Para llevar la información de una ferretería, inventario, ventas, compras.
  + **Un Data Warehouse:** Obtener información de una compañía, reportes de gestión alineados a la estrategia, indicadores, Balance Scorecard.
  + **Un Data Lake:** En Google cada día la información aumenta en grandes volúmenes, imágenes/videos, textos, correos, sonidos, datos de sensores búsquedas etc. Requieren Data Lake para almacenar la información y disponibilizarla hoy y en el futuro para análisis.
  + **Un Lake House:** En Codelco como en todas las grandes compañías se genera gran cantidad de información, de los trabajadores, de los equipos industriales, camiones, procesos de extracción etc. Cada día se requiere adquirir nuevos conocimientos de la organización, optimizando sus procesos, descubriendo también información nueva. Aprovechando las capacidades y tecnologías la empresa evoluciona al futuro.

***Pregunta 2. Desarrollo***

* Explique con sus palabras cuales son los *drivers* o causas que hacen necesario la utilización de “clústers de computadoras”
* Refiérase especialmente al uso de HPC (clústers intensivos en cómputo) vs al uso de clústers para Big Data (tipo Hadoop o Spark)
* Proponga un ejemplo o caso de uso para cada uno de ellos.

***Respuesta:***

Las causas que hicieron necesaria la utilización de clústers de computadores son principalmente la necesidad de aumentar el poder de cómputo, paralelizar tareas para ejecución más rápida. Los clústeres tienen la gracia de ser escalables, no será necesario comprarse un mega computador potente, si no varios mas pequeños que trabajarán juntos y en caso de necesitar mas procesamiento crece el cluster.

La principal diferencia entre HPC y los clústers para Big Data, es que en HPC la información se lleva a los módulos de procesamiento, pero cuando se trabaja con Big Data esto ya no es factible porque trasladar grandes cantidades de información es muy costoso que viaje, aunque sea posible procesarla. En los clúster para Big Data la filosofía es al revés, es llevar el procesamiento a los datos. La información se divide y se procesa en las distintas celdas, como se está particionando la información se debe asegurar mantener la información completa, por eso se replica en distintos nodos para prevenir pérdidas en caso de falla de un nodo de cluster o de un cluster completo. En este tipo de procesamiento para Big Data es muy importante mantener el orden y control sobre la información particionada, para que para el usuario sea transparente que la información esta distribuida.

Ejemplos

HPC: Simulaciones y modelos 3D.

Clústeres para Big Data: Análisis de imágenes y videos de las autopistas para analizar y predecir comportamientos.

*Pregunta 3. Ejercicio EDA con Spark SQL*

* Utilizando Apache Spark (en Colab, o DataBricks, o local, …), las librerías de SparkSQL y el dataset [*SII\_roles\_avaluo\_table.zip*](https://www.dropbox.com/s/ianxkffu14dp0ru/SII_roles_avaluo_table.zip)…
* Escriba 3 nuevas consultas (queries) a la data:
  + Debe usar a lo menos un Left Join
  + Debe usar a lo menos un GroupBy
  + Debe usar un a lo menos Where con dos condiciones
  + Debe ordenar sus resultados de manera descendiente

***Respuesta:***

Acceso a Google Colab:

<https://colab.research.google.com/drive/1woSXFQ8CRTEnPEGIelVEEiDqaHrn88ah?usp=sharing>

Consulta 1:

#Avalúo mas caro de cada comuna, ordenados por valor descendente

spark.sql("select A.cod\_comuna, B.comuna, max(A.avaluo\_fisc\_tot) as max\_avaluo from noagricola\_sql A LEFT JOIN  comunas\_sql B ON A.cod\_comuna = B.comunaID GROUP BY A.cod\_comuna, B.comuna order by max\_avaluo desc").show()

Consulta 2:

#Avalúo promedio por metro cuadrado de las comunas de la quinta región ordenado ascendente

spark.sql("select A.cod\_comuna, B.comuna, bround(avg(A.avaluo\_fisc\_tot/A.sup\_total\_terreno\_prod\_m2)) as avaluo\_x\_m2 from noagricola\_sql A LEFT JOIN  comunas\_sql B ON A.cod\_comuna = B.comunaID WHERE B.n\_region = 5 GROUP BY A.cod\_comuna, B.comuna order by avaluo\_x\_m2 asc").show()

Consulta 3:

#Total de predios sin construcciones en Ñuñoa mayores a 1000 metros cuadrados

spark.sql("select count(A.\*) as total\_predios\_sin\_construcciones from noagricola\_sql A LEFT ANTI JOIN constr\_noagricola\_sql B ON A.cod\_comuna = B.cod\_comuna AND A.nr\_manzana = B.nr\_manzana AND A.nr\_predial = B.nr\_predial WHERE A.cod\_comuna=15105 AND A.sup\_total\_terreno\_prod\_m2>1000").show()

*Pregunta 4. Opción A. Ejercicio Clustering Bancario*

* Utilizando Apache Spark (en Colab, o DataBricks, o local, …), las librerías de SparkML y el dataset créditos\_bancarios.xlsx …
* Realice una segmentación (**usando K-means)** de la cartera de clientes que incluya a lo menos 5 variables de su interés.
* En sus resultados comente respecto de:
  + A) qué tipo de preprocesamientos fue necesario realizarle a los datos, o sino fue necesario, el por qué.
    - No fue necesario realizar preprocesamiento dado que no había datos perdidos,, los 1000 datos de la base tenían datos correctos, la mayoría de los datos eran codificaciones, y los valores correspondían, lo que se pudo observer con **describe** y seleccionando los datos distintos.
    - Todos los datos eran numéricos enteros (dummy?)
    - Solo hubo que juntar las columnas que me intesaban en una sola como lo require SPARK.
  + B) Cómo se determinó el número óptimo de clusters a utilizar
  + C) La estadística descriptiva de los segmentos encontrados y qué nombre “comercial” le pondría al segmento dadas dichas características
  + D) Acciones de negocios que podrían ser relevantes a sugerir para los segmentos encontrados

***Respuesta:***

Acceso a Colab:

<https://colab.research.google.com/drive/1ZYxW-cNo1R2uDCQO-iVafIylhBdMStz-?usp=sharing>

Tuve que copiar un archivo .csv en mi dropboc porque no pude leer el archivo con múltiples hojas.