TRABAJO FINAL:
Big Data Analytics & Data Mining.

Curso: Big Data & Data Analytics
Fecha: 06 de marzo 2023
Profesor: Ph.D. David Diaz
Alumnos: Bastián Celedón – Alonso Matteo

Pregunta 1

- **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**

- Las *Bases de Datos Transaccionales* y los *Data Warehouse* almacenan información estructurada, a diferencia los *Data Lake* y *Lake House*, los cuales almacenan información sin estructura.
- Los tipos de usuarios que utilizan las *Bases de Datos Transaccionales* pertenecen a áreas de Negocios, al igual que aquellos que utilizan *Data Warehouses.* En el caso de los *Data Lake* y *Lake House,* sus usuarios no son de áreas de negocios, sino que son especialistas como Data Scientist.
- Las *Bases de Datos Transaccionales,* los *Data Warehouses,* y por otro lado, los *Data Lake* y *Lake House,* se diferencian por los objetivos que buscan satisfacer: las *Bases de Datos Transaccionales* almacenan datos consistente y simultáneamente por varios usuarios, los Data Warehouses buscan que una gran cantidad de usuarios puedan acceder simultáneamente a los datos que tienen estructura y han sido filtrados y procesados previamente para poder hacer análisis y generar reportería para la empresa; los *Data Lake* tienen como objetivo principal almacenar grandes volúmenes de datos, solo eso, no pretender efectuar procesamiento sobre ellos. Y por último, los *Lake House* tienen como objetivo almacenar datos, al igual que un *Data Lake*, pero además buscan efectuar análisis sobre ellos, tal cual los *Data Warehouse*.
- Los *Data Lake* y *Lake House* pueden almacenar los mismos tipos de datos: estructurados, no estructurados o semiestructurados.
- La única similitud entre Data Lake, Lake House y Data Warehouse es que los 3 pueden almacenar una gran cantidad de datos.
- Los Lake House se diferencian de los Data Lakes y Data Warehouse en que los primeros están optimizados para Ciencias de Datos y herramientas de aprendizaje.
- Los Data Warehouse y Bases de Datos Transaccionales son utilizados por usuarios operativos, no así los Data Lake o Lake House que son usados por usuarios analistas.
- Con respecto a los costos, los Data Lake y Lake House son mucho más accesibles que los Data Warehouses.

- **Proponga un ejemplo y caso de uso para cada una de ellas:**
    - **Base de datos transaccional:** un ejemplo corresponde a MySQL. Un caso de uso de este tipo se da en las empresas adquirentes que requieren efectuar la captura y

procesamiento de las ventas efectuadas en los comercios con tarjetas de crédito o débito. Existe un elevadísimo número de transacciones, las cuales pueden corresponder a ventas, anulaciones o reversas por problemas de comunicación.

- **Data Warehouse**: algunos ejemplos son Amazon Redshift y Google BigQuery Este tipo podría ser utilizado por grandes cadenas de Retail, en donde es necesario procesar la información obtenida desde distintos orígenes (internos, externos) para diversos procesos internos como logística, ventas, compras, pricing, marketing o análisis de tendencias, entre otros.
- **Data Lake**: algunos ejemplos son Microsoft Azure Data Lake, Cloud Storage o Amazon S3. Estos tipos pueden ser utilizados para almacenar datos no estructurados como videos, audios, textos o imágenes, los cuales pueden ser utilizados posteriormente por servicios en línea que efectúen procesamiento sobre ellos, como por ejemplo transformar audios de un idioma a otro, generar audiolibros a partir de textos, transformar imágenes u obtener el texto de los videos.
- **Lake House**: ejemplos son Snowflake y Databricks. Dado que este tipo de arquitectura permite almacenar datos estructurados y no estructurados, es altamente recomendable utilizarlo en empresas maduras en las que se requiera implementar cargas de trabajo de análisis avanzado y efectuar aprendizaje automático de éstos.

Pregunta 2

- **Explique con sus palabras cuales son los *drivers* o causas que hacen necesario la utilización de "clústers de computadoras".**

  Los *drivers* que son considerados son:

  - **La capacidad de Cómputo que posee el hardware en una PC**: Esto dependerá de los componentes que posee la PC, entre ellos se encuentran los siguientes y sus principales características:
    - La cantidad de cores de los procesadores: a mayor cantidad de cores de un procesador, mayor cantidad de transistores en los microchips que la componen, aumenta la capacidad de cómputo (**cantidad de FLOPS**).
    - La memoria RAM: mientras más capacidad, mejor es capaz de procesar las tareas sin el rendimiento del procesador, ya que permite mejorar la rapidez del procesamiento de datos y el traslado de estos dentro del flujo o arquitectura de datos.
    - Si posee sólo CPU: respaldándose en la capacidad de procesamiento lineal, respaldando información en el hardisk.
    - Si posee adicionalmente una GPU: esto permite mejorar la capacidad de cómputo al incluir una tarjeta de video que tiene como objetivo ejecutar procesamientos de datos de manera paralela.

  - Cabe destacar que adicionalmente a la cantidad o calidad de los componentes, también es necesario considerar el tipo de procesamiento de datos que puede lograr con estos, por lo tanto, también toma responsabilidad **las diferentes maneras de computar**:
    - Para resolver un problema de tipo lineal y/o secuencial en serie: es posible de realizar con la CPU y la memoria RAM, por lo tanto, es posible realizar un análisis de datos que requieren de terminar una tarea para continuar con otra al procesar una gran cantidad de datos esto requiere mayor tiempo, cuando es Big Data, es necesario un poder de cómputo más efectivo. Por lo tanto,
    - Para resolver este tipo de procesamiento de Big Data, se requiere repartir la data en procesamiento paralelo, y de esta manera es posible utilizar todo el poder de cómputo de la PC, esta es la forma de trabajo de Tarjetas de Video - o GPU, que están diseñadas para el procesamiento paralelo, pero aún así con la cantidad de data creada hasta la actualidad es complejo resolver el análisis de datos en Big Data, por falta de poder de cómputo inclusive con el hardware más avanzado para una sola computadora, por lo tanto, es necesario construir clústers de computadoras. O sea, interconectar una gran cantidad de computadoras con gran capacidad de cómputo y grandes cantidades de procesadores para así disminuir el tiempo en el procesamiento de datos y mejorar el

rendimiento enfocado a las distintas tareas necesarias de ejecutar en las búsquedas, traspaso, lectura y procesamiento de datos.

- **Refiérase especialmente al uso de HPC (clústers intensivos en cómputo) vs al uso de clústers para Big Data (tipo Hadoop o Spark)**

  HPC (High Performance Computing) - el objetivo de estos clústers que tienen un alto rendimiento, es tener un mayor poder de cómputo, lo que multiplica la capacidad dependiendo de cuantas computadoras se unen, las supercomputadoras tienen una capacidad de procesamiento de datos en Petabytes, lo que significan miles de Gb de traspaso de información analizándose simultáneamente. Éstas están interconectadas físicamente, requieren de enfriadores para mantener temperaturas estables, una vez ya unidas, las computadoras pueden resolver problemas complejos con procesamiento de datos en paralelo, seccionados en nodos, utilizando una gran cantidad de datos, pero aquí el problema se traspasa a otra parte, la cola generada por la cantidad de consultas que se deben realizar desde la base de datos y es por eso que para resolverlo, se pretende realizar el procesamiento de la información generando estas particiones de nodos, que graban partes de información seccionada y deben generar muchos respaldos para no perder la información, esto estima que un mismo documento puede estar multiplicado x5 y distribuido en distintos nodos para realizar las búsquedas correspondientes, quedando diversificadas en la cantidad de computadoras que pertenecen al cluster.

  Uso de Clústers para Big Data de Open Source - De precio accesible y menor requerimientos de inversión, muchos de ellos son gratuitos y con limitaciones en la capacidad de procesamiento o características que poseen las computadoras solicitadas o “prestadas” ya que son servicios abiertos, los cuales mediante una membresía son capaces de brindar el servicio de procesamiento de datos, ya sean libres o de “arriendo”, están sustentados en supercomputadoras que se pueden encontrar a grandes distancias, pero no es necesario el aparataje y la inversión multimillonaria en hardware, ya que los nodos de clusters de computadores reales son “prestados” en función de los parámetros que el usuario solicite, la ausencia de inversión en hardware, le diferencia de ser un tipo de procesamiento rápido y versátil, ya que son soportados en cloud.

- **Proponga un ejemplo o caso de uso para cada uno de ellos.**

  Ejemplo de HPC - Son Frontier, la Supercomputadora más potente del mundo, puede usarse principalmente en campos como la biología de sistemas, la ciencia de materiales, la producción de energía, la fabricación aditiva y la ciencia de datos de salud, otros usos relacionados se encuentran en la predicción de clima, el gran inversionista detrás es el Ministerio de Defensa de los Estados Unidos. Otro ejemplo es: el National Laboratory High Performance Computing Chile (NLHPC) el Laboratorio Nacional de Computación de Alto Rendimiento, el Supercomputador más poderoso de Chile, este último tiene por objetivo instalar contar con una red de infraestructura de computación de alto rendimiento, para el procesamiento de datos científicos y sus aplicaciones industriales, al ser poseedor la Universidad de Chile, el uso puede ser diverso, desde lo relacionado a los campos de la

Astronomía, Optimizaciones de Procesos Industriales de alta complejidad, procesamiento de grandes cantidades de datos estadísticos, como a su vez utilizados en economía, entre otros.

Ejemplo de uso de Clústers para Big Data - Son el uso de plataformas de Big Data como lo son Open MPI, un Big Data Open Source, que a través del lenguaje de programación Python es posible utilizar en el procesamiento de datos con Clusters, utilizando modelos de alta complejidad especialmente diseñados para el procesamiento de datos en paralelo. Utilizan una memoria distribuida y permiten sacar el máximo provecho acorde al tipo de arquitectura de datos que posee.

Pregunta 3

Se utilizó Apache Spark en Colab, las librerías de SparkSQL y el dataset *SII_roles_avaluo_table.zip*

El archivo correspondiente se encuentra en Google Colaboratory:
https://colab.research.google.com/drive/1IF9x8e_wekgzYTifrN1xXjUsTNxhbkBP?usp=sharing

- **QUERY 1:** Listado ordenado por avalúo fiscal de todas las propiedades no agrícolas de la Región de Arica y Parinacota que poseen una contribución semestral superior a 100 millones.

```
SELECT B.comuna, A.*
FROM noagricola_sql as A
LEFT JOIN comunas_sql as B
ON A.cod_comuna = B.comunaID
WHERE B.n_region=18
AND contri_semestral >= 100000000
ORDER BY avaluo_fisc_tot desc
```

comuna	_c0	cod_comuna	nr_manzana	nr_predial	dir/nomb_predio	avaluo_fisc_tot	contri_semestral	cod_dest_princ	avaluo_exen_prop	cod_comuna_rol_bien
ARICA	11575	1101	660	4	DIEGO PORTALES 64...	39963745227	222398242	C	0	
ARICA	12	1101	1	21	M LIRA S/N CONCES...	32080750604	178529378	T	0	
ARICA	12000	1101	670	3	AV. DIEGO PORTALE...	19373192385	107811818	C	0	

- **QUERY 2:** Listado ordenado por la Contribución semestral promedio cancelada por los predios agrícolas de cada comuna de la Región de Región Aysén del General Carlos Ibáñez del Campo.

```
SELECT B.comuna, count(A.avaluo_fisc_tot) as conteo_predios,
avg(A.contri_semestral) as contribucion_semestral_promedio
FROM agricola_sql as A
LEFT JOIN comunas_sql as B
ON A.cod_comuna = B.comunaID
WHERE B.n_region=11
AND contri_semestral > 0
GROUP BY B.comuna
ORDER BY contribucion_semestral_promedio DESC
```

comuna	conteo_predios	contribucion_semestral_promedio
TORTEL	9	3128332.6666666665
AISÉN	395	317674.87594936707
LAGO VERDE	166	262837.3493975904
COYHAIQUE	1052	254615.51901140684
CHILE CHICO	288	211497.45833333334
CISNES	286	179569.78321678322
COCHRANE	145	111096.78620689655
OHIGGINS	46	66499.0
RÍO IBÁÑEZ	280	58900.91428571429

- **QUERY 3:** Número de construcciones ordenadas descendentemente por año de construcción y que pertenecen a la Comuna de Lo Barnechea.

```
select anio_linea_constr as anio_construccion, count(*) as
nro_construcciones

FROM terreno_constr_no_agri_sql

WHERE cod_comuna = 15161

GROUP BY anio_linea_constr

ORDER BY anio_linea_constr desc
```

```
+-----------------+------------------+
|anio_construccion|nro_construcciones|
+-----------------+------------------+
|             2020|               915|
|             2019|              2855|
|             2018|              3148|
|             2017|              4844|
|             2016|              3898|
|             2015|              3682|
|             2014|              3183|
|             2013|              4309|
|             2012|              2732|
|             2011|              2634|
|             2010|              1743|
|             2009|              3626|
|             2008|              1724|
|             2007|              1645|
|             2006|              2287|
|             2005|              2716|
|             2004|              1798|
|             2003|              1634|
|             2002|              1454|
|             2001|              2179|
+-----------------+------------------+
only showing top 20 rows
```

Pregunta 4

- 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.

El desarrollo de la Segmentación de la Cartera de Clientes realizado con la Machine Learning de Spark Apache en Google Colab se encuentra en el siguiente link:

https://colab.research.google.com/drive/1045DMZZV1BhJ-jLA6yu6llG-nuSimTIO?usp=sharing

- En sus resultados comente respecto de:
    - A) qué tipo de pre procesamientos fue necesario realizarles a los datos, o si no fue necesario, el por qué.
        - No fue necesario preparar ni transformar los datos para pasarlos a kmeans, sólo la conversión de la data de archivo excel .xlsx a .csv Al observar la data no se observan valores perdidos y como spark sólo puede trabajar con data numérica, se corrobora que todos los valores están en números enteros.
    - B) Cómo se determinó el número óptimo de clusters a utilizar
        - Al iniciar el modelo se establece que deben considerarse sólo 2 clusters, ya que lo que pretendemos hacer es saber si es necesario otorgar o no un crédito bancario dependiendo de los atributos que posee el cliente, por lo tanto la variable numérica de valor 1 identifica la búsqueda de clientes a los que se les pueden aprobar créditos bancarios, en cambio 0, es la variable numérica que de los clientes que pretendemos rechazar porque no cumplen con los atributos necesarios para poder otorgar dichos créditos. Esto será comprobado al terminar la clusterización luego de la clasificación automática que realiza la Machine Learning.
    - C) La estadística descriptiva de los segmentos encontrados y qué nombre “comercial” le pondría al segmento dadas dichas características
        - Como anteriormente mencionamos la creación de los segmentos encontrados nos permite establecer 2 clases de clientes, los que según sus atributos pueden obtener el crédito bancario, considerados por el resultado 1 y los clientes que son rechazados por la clasificación de la machine learning con el valor 0, por lo tanto, es semejante al concepto utilizado en banca como “Créditos preaprobados”, los cuales son rápidamente clasificados luego de considerar una cantidad de atributos, en este caso 5 fundamentales para el otorgamiento o no de créditos.

- D) Acciones de negocios que podrían ser relevantes a sugerir para los segmentos encontrados
    - Generar capacidad de ahorro - Mejora la clasificación de los ahorros.
    - Mejorar Nivel de Ingresos - Clasifica y pondera mejor el nivel de ingresos.
    - Mejorar comportamiento de pago - Permite mejorar la clasificación de riesgo.