Fundamentos de Big Data utilizando MATLAB

Yersinio Jiménez Campos
Analista de datos
Banco Nacional de Costa Rica

Agenda

- ¿Qué es Big Data?
- Buenas prácticas en el manejo de memoria.
- Computación en paralelo con MATLAB.
- Uso de `datastore`.
- Lectura parcial de archivos.
- Mapreduce en MATLAB.
- Conclusiones.

¿Qué es Big Data?

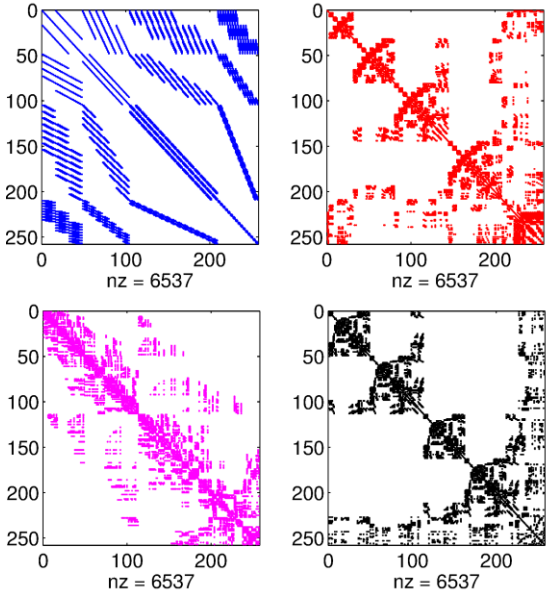
- Datos Masivos.
- Datos Gigantes.
- Macrodatos.
- Grandes volúmenes de datos.
- **Big data** hace referencia a una cantidad de datos tal que supera la capacidad del software habitual para ser capturados, gestionados y procesados en un tiempo razonable.

¿Qué es Big Data?

- Habitualmente se habla de conjuntos de datos que no pueden ser gestionados directamente en memoria RAM.
- Se trata tanto de datos estructurados como no estructurados.
- Se caracteriza por su alto **volumen**, **velocidad** y **variedad** que demandan soluciones de procesado para la mejora del conocimiento y toma de decisiones en las organizaciones.

Memoria en MATLAB: Buenas prácticas

- Utilice MATLAB de 64 bits siempre que sea posible.
- Seleccione adecuadamente sus estructuras de datos:
  - Use solamente la precisión que requiere.
  - Matrices dispersas/ralas.
  - Arreglos categóricos.
  - Considere la sobrecarga adicional por el uso de structs y cellarrays.
- Minimice la cantidad de copias de los datos:
  - Lazy copy.
  - Funciones anidadas.
  - In-place operations.
  - Si utiliza objetos, considere utilizar referencias.



Big Data en MATLAB

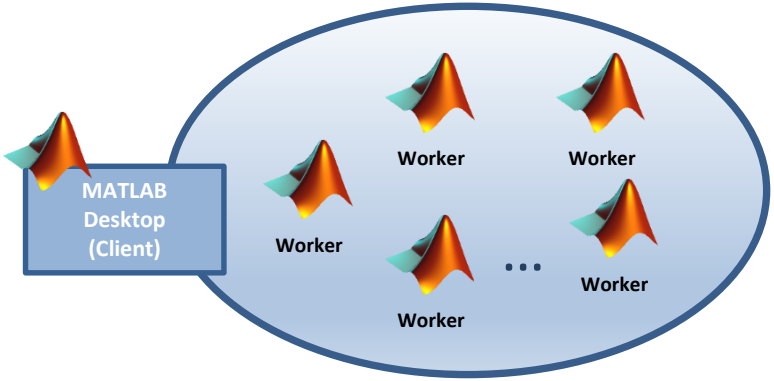
- **Memoria y acceso a datos:**
  - Procesadores de 64bits.
  - Variables mapeadas a memoria.
  - Variables en disco.
  - Bases de datos.
  - **Datastore.**
- **Directrices de programación:**
  - Procesamiento de datos por bloques/lotes.
  - Parfor.
  - Programación en el GPU.
  - SPMD y arreglos distribuidos.
  - **Map Reduce.**
- **Plataformas:**
  - Múltiples núcleos.
  - Clústeres.
  - Cloud (MATLAB Distributed Computing Server y EC2).
  - **Hadoop.**

Aspectos a considerar

- **Características de los datos:**
  - Tamaño, tipo y ubicación.
- **Plataforma de cómputo:**
  - Un único equipo de escritorio o clúster.
- **Características del análisis:**
  - Fácilmente paralelizable.
  - Evaluar segmentos de datos y luego agregar los resultados.
  - Trabajar con el conjunto de datos completo.

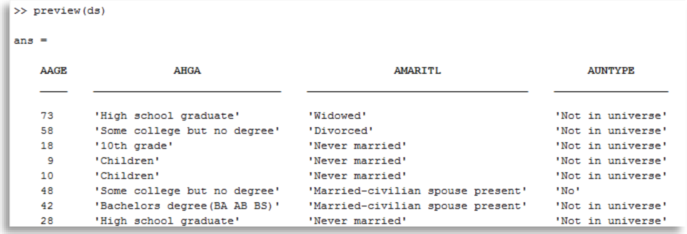
Computación en paralelo con MATLAB: parfor

- Se tiene un nodo cliente (de escritorio).
- Se crea un grupo de peones o nodos esclavos.
- Esto es recomendable en equipos de múltiples núcleos y/o múltiples procesadores.



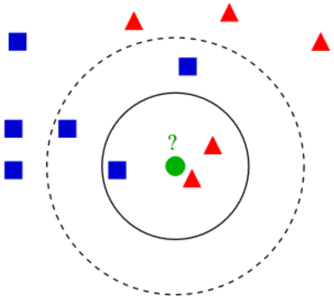
Ejemplo: KNN con varias distancias y vecindarios

- Datos
  - Censo PUMS no ponderados de las áreas de Los Ángeles y Long Beach para los años 1970, 1980 y 1990.
  - 20 mil filas y 42 columnas.
- Análisis
  - Clasificación mediante KNN.
  - Se van a utilizar diferentes distancias.
  - Distintos valores de “k”, entre 2 y 6.
- Se va a realizar la ejecución secuencial y luego con parfor.



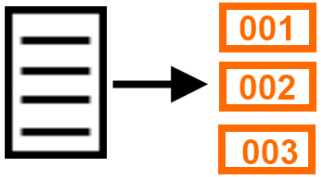
```
>> preview(ds)

ans =
```

AAGE	AHGA	AMARITL	AUNTYPE
73	'High school graduate'	'Widowed'	'Not in universe'
58	'Some college but no degree'	'Divorced'	'Not in universe'
18	'10th grade'	'Never married'	'Not in universe'
9	'Children'	'Never married'	'Not in universe'
10	'Children'	'Never married'	'Not in universe'
48	'Some college but no degree'	'Married-civilian spouse present'	'No'
42	'Bachelors degree(BA AB BS)'	'Married-civilian spouse present'	'Not in universe'
28	'High school graduate'	'Never married'	'Not in universe'



¿Cuándo utilizar `parfor`?

- **Características de los datos:**
  - Pueden ser de cualquier tipo permisible por MATLAB, siempre y cuando se puedan partir en varios grupos.
  - Los datos de cada iteración deben caber en la memoria.
- **Plataforma de cómputo:**
  - Equipo de escritorio (Parallel Computing Toolbox).
  - Clúster (MATLAB Distributed Computing Server).
- **Características del análisis:**
  - Cada iteración del ciclo debe ser independiente.



Accediendo Big Data: `datastore`

- Fácil especificación del conjunto de datos.
  - Uno o varios archivos de texto.
  - Una base de datos (utilizando el Database toolbox)
- Permite la pre visualización de la estructura y del formato.
- Permite la selección de datos a importar mediante el nombre de las columnas.
- Lee de manera incremental los sub grupos de datos.

DemoDatastore.m

```
ds = datastore('census-income-train.csv', 'delimiter', ';');
ds.SelectedVariableNames = {'AAGE', 'AHGA', 'AMARITL', 'AUNTYPE',...
                            'PRCITSHP', 'PTOTVAL'};
X = read(ds);
```

```
>> preview(ds)

ans =

    AAGE              AHGA                          AMARITL                                AUNTYPE
    ____    ____________________________    ____________________________________    ________________

     73     'High school graduate'          'Widowed'                               'Not in universe'
     58     'Some college but no degree'    'Divorced'                              'Not in universe'
     18     '10th grade'                    'Never married'                         'Not in universe'
      9     'Children'                      'Never married'                         'Not in universe'
     10     'Children'                      'Never married'                         'Not in universe'
     48     'Some college but no degree'    'Married-civilian spouse present'       'No'
     42     'Bachelors degree(BA AB BS)'    'Married-civilian spouse present'       'Not in universe'
     28     'High school graduate'          'Never married'                         'Not in universe'
```

Ejemplo: Análisis de los datos del censo PUMS

- Datos
  - Censo PUMS no ponderados de las áreas de Los Ángeles y Long Beach para los años 1970, 1980 y 1990.
  - Aproximadamente 100 mil registros y 42 columnas.
- Análisis
  - Edad promedio de los encuestados.
  - Porcentaje de personas mayores a 65 años.
  - Porcentaje de nativos menores a 10 años.

```
>> preview(ds)

ans =

    AAGE          AHGA                    AMARITL                          AUNTYPE
    ____    ____________________    ____________________________    ________________

      73    'High school graduate'    'Widowed'                     'Not in universe'
      58    'Some college but no degree'    'Divorced'              'Not in universe'
      18    '10th grade'              'Never married'               'Not in universe'
       9    'Children'                'Never married'               'Not in universe'
      10    'Children'                'Never married'               'Not in universe'
      48    'Some college but no degree'    'Married-civilian spouse present'    'No'
      42    'Bachelors degree(BA AB BS)'    'Married-civilian spouse present'    'Not in universe'
      28    'High school graduate'    'Never married'               'Not in universe'
```

¿Cuándo utilizar `datastore`?

- **Características de los datos:**
  - Información textual en archivos planos.
  - Información en bases de datos o HDFS (Hadoop Distributed File System).
- **Plataforma de cómputo:**
  - Escritorio (Desktop).
- **Características del análisis:**
  - Soporta el flujo de trabajo de carga, análisis y descarga.
  - Lectura incremental de bloques de datos, para su procesado dentro de un ciclo `while`.

Lectura parcial de datos desde un archivo

- Archivo de texto, archivo ASCII:
  - `datastore`.
- Archivo `MAT`:
  - Carga y guardado de una parte de una variable utilizando `matfile`.
- Archivos binarios:
  - Lectura y escritura directa desde/hacia un archivo utilizando `memmapfile`.
  - Mapeo del espacio de direcciones hacia un archivo.
- Bases de datos:
  - Manejo de datos mediante ODBC y JDBC (ej. MSSQL Server, ORACLE, MySql, etc).

Análisis de Big Data: `mapreduce`

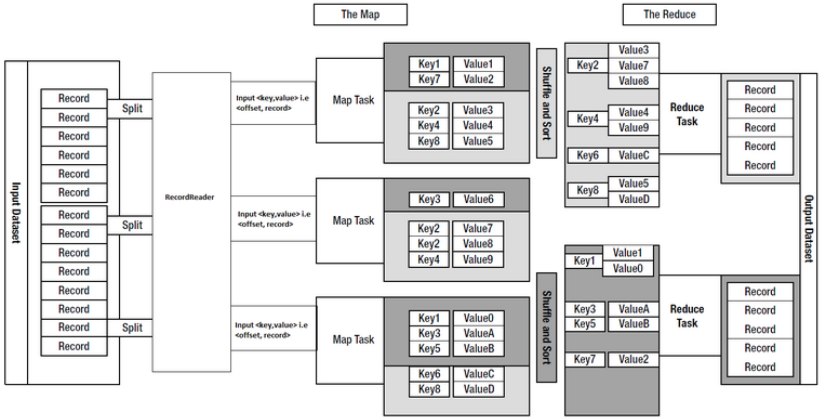
- Utiliza la poderosa técnica de programación MapReduce para analizar grandes conjuntos de datos.
  - `mapreduce` utiliza un `datastore` para procesar datos en pequeños bloques, los cuales caben en memoria de manera individual.
  - Es muy útil para procesar múltiples grupos, o cuando los resultados intermedios no caben en memoria.
- `mapreduce` en equipo de escritorio (Desktop):
  - Analiza grandes tablas de bases de datos (Database Toolbox).
  - Incrementa la potencia de cálculo (Parallel Computing Toolbox).
  - Puede acceder datos en HDFS para desarrollar algoritmos a utilizar en Hadoop.

Análisis de Big Data: `mapreduce`

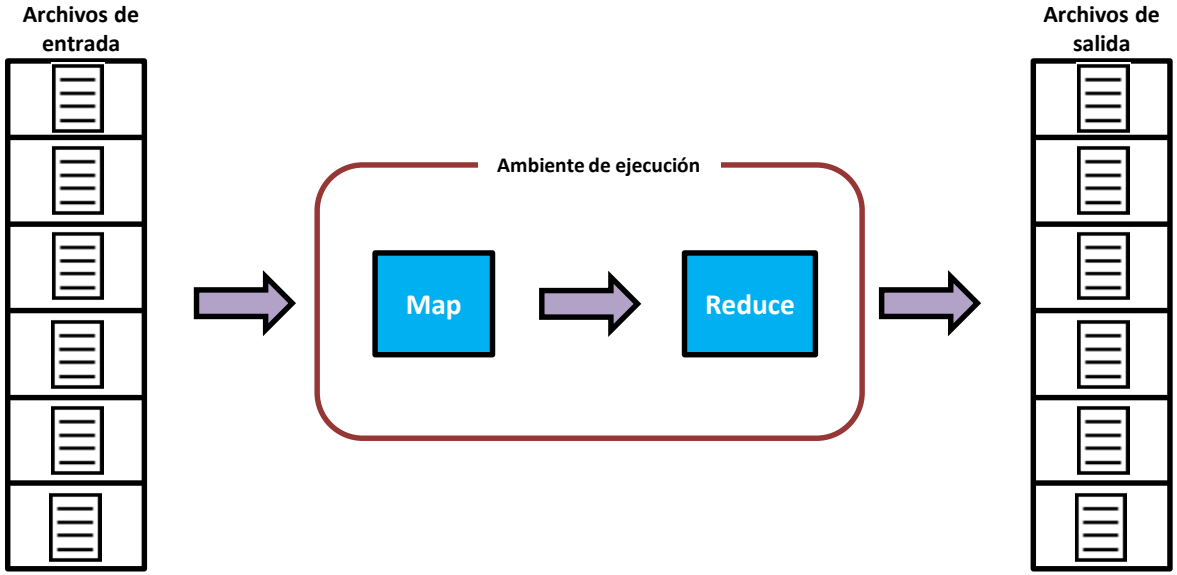
- `mapreduce` con Hadoop
  - Se puede ejecutar en Hadoop utilizando el MATLAB Distributed Computing Server.
  - Permite la generación y exportación de aplicaciones y librerías para Hadoop utilizando el MATLAB Compiler.

Hadoop

- Open-source software framework de Apache, inspirada en:
  - Google Map-Reduce.
  - GFS (Google File System).
- Muy popular para analizar enormes cantidades de datos.
- Inspirado en el proyecto GFS y en el paradigma MapReduce.
- Hadoop está compuesto de tres piezas: HDFS, Hadoop MapReduce y Hadoop Common.



Modelo de programación con mapreduce



Modelo de programación con mapreduce

- `mapreducer`: controla el ambiente de ejecución de la operación MapReduce.

Modelo de programación con mapreduce

- `datastore`: almacena la información relacionada con los archivos de entrada y salida.
  - La entrada es un objeto datastore, que está asociado a los archivos de entrada.
  - La salida es otro objeto datastore, asociado a los archivos de salida.
- `mapreduce`: ejecuta el algoritmo de mapreduce.
  - Utiliza el objeto datastore de entrada.
  - Aplica la función map.
  - Aplica la función reduce.

Conclusiones

- MATLAB cuenta con una gran gama de herramientas para el manejo de Big Data.
- Facilita la computación en concurrente: paralelo (parallel toolbox) y distribuido (MDCS).
- Permite el manejo de grandes conjuntos de datos: memmap file, datastore, etc.
- Recientemente, incorpora características que permiten el uso del paradigma mapreduce.
- Puede trabajar de maneja conjunta con Hadoop.

Recursos adicionales

- Manejo de memoria:
  - http://www.mathworks.com/help/matlab/performance-and-memory.html
- Parallel for loops (parfor)

  http://www.mathworks.com/help/distcomp/parallel-for-loops-parfor.html
- Mapreduce
  - http://www.mathworks.com/help/matlab/mapreduce.html
- Big data con MATLAB
  - http://www.mathworks.com/help/distcomp/parallel-mapreduce-for-big-data.html
  - http://www.mathworks.com/help/matlab/large-files-and-big-data.html

Muchas gracias