Pontificia Universidad Javeriana Cali

Facultad de Ingeniería
Ingeniería de Sistemas y Computación

PROCESAMIENTO DE GRANDES VOLÚMENES DE DATOS

Integrantes:
Ana María García, Andrés Felipe Delgado, Leonardo Sáez, Katherine Camacho

Project 1 (data cleaning + MLlib)

Período 2020-2

1. Descripcion del conjunto de datos inicial

El conjunto de datos "Ávila"se ha extraído de imágenes de la "Biblia de Ávila". El análisis del manuscrito ha identificado la presencia de 12 autores. Cada patrón contiene 10 características (F1,F2,F3,...,F10). La tarea de predicción consiste en asociar cada escrito a uno de los 12 autores (etiquetados como: A, B, C, D, E, F, G, H, I, W, X, Y).
Descripción de los Atributos:
F1 : Distancia entre columnas, F2 : Margen superior, F3 : Margen inferior, F4 : Explotación, F5 : Número de filas, F6 - Relación Modular, F7 - espaciado interlineal, F8 - Peso, F9 - número pico, F10 - Relación Modular/ espaciado interlineal
- Author: A, B, C, D, E, F, G, H, I, W, X, Y (Atributo clasificador, 12 categorias)
Las características que se mencionan a continuación fueron obtenidas con Pyspark:

- Consta de 11 atributos y 20867 registros

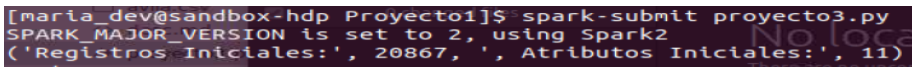

Figura 1: Cantidad inicial de registros y atributos

- Todos los atributos (excepto el clasificador Author) son de tipo double

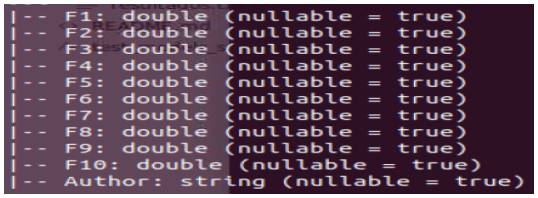

Figura 2: Tipo de datos de los Atributos

- No contiene datos nulos en ningún registro

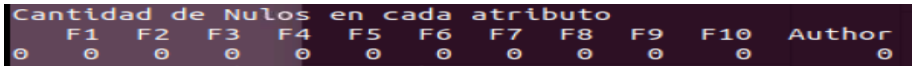

Figura 3: Cantidad de Nulos por Atributo

- El atributo F6 y F10 tienen un índice de correlación de 0,81

Correlacion entre atributos

	F1	F2	F3	F4	F5	F6	F7	F8	F9	F10
F1	1.000000	-0.046363	0.036442	-0.057191	0.447930	-0.053406	-0.026103	-0.057518	0.081785	-0.013904
F2	-0.046363	1.000000	0.300381	0.000425	-0.085892	0.296413	0.459383	0.040535	0.231004	-0.031986
F3	0.036442	0.300381	1.000000	0.144168	0.011871	0.034247	0.185502	0.032930	0.120078	-0.057389
F4	-0.057191	0.000425	0.144168	1.000000	0.096447	0.266778	0.043836	0.358603	0.294643	0.327626
F5	0.447930	-0.085892	0.011871	0.096447	1.000000	0.041697	0.019832	-0.065375	0.286023	0.144194
F6	-0.053406	0.296413	0.034247	0.266778	0.041697	1.000000	0.395452	-0.043804	0.148651	0.806490
F7	-0.026103	0.459383	0.185502	0.043836	0.019832	0.395452	1.000000	0.015823	0.176611	0.274931
F8	-0.057518	0.040535	0.032930	0.358603	-0.065375	-0.043804	0.015823	1.000000	0.519697	0.001312
F9	0.081785	0.231004	0.120078	0.294643	0.286023	0.148651	0.176611	0.519697	1.000000	0.192805
F10	-0.013904	-0.031986	-0.057389	0.327626	0.144194	0.806490	0.274931	0.001312	0.192805	1.000000

Figura 4: Correlación entre Atributos

- El atributo F2 tiene uno o varios registros demasiado atípicos (NOTA: El diagrama mostrado a continuación se generó con Jupyter, pues no se encontró una forma de visualizar este diagrama a través de consola)

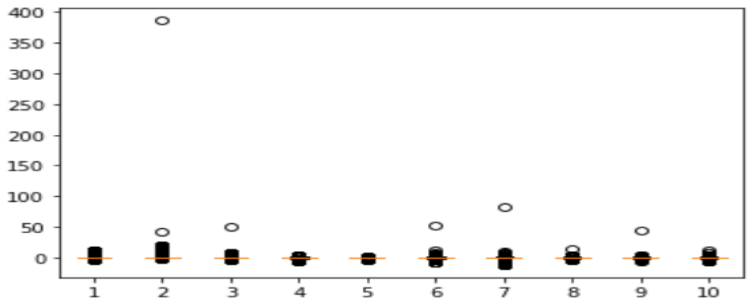

Figura 5: Diagrama de cajas

- El atributo clasificador (Author) se divide de la siguiente manera:

Distribucion del atributo clasificador

Author	count
F	3923
E	2190
B	10
Y	533
D	705
C	206
A	8572
X	1044
W	89
G	893
I	1663
H	1039

Figura 6: Distribución Categórica Inicial

2. Transformaciones Limpieza del Conjunto

- Se eliminaron los datos muy atípicos del atributo F2 (Fue un atributo)

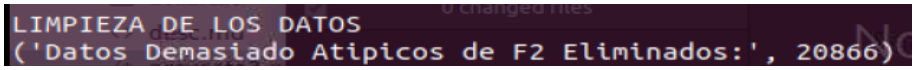

Figura 7: Datos Atipicos Eliminados

- Se mantienen tanto el atributo F6 como el F10, pues se considera que una correlación de 0.8 no es lo suficientemente alta como para eliminar algún atributo
- Se convirtieron los atributos categóricos a numéricos, en este caso fue únicamente el atributo Author el cual es el atributo clasificador
- Se balancearon los datos de manera que cada categoría tuviese entre 1000 y 2000 registros.

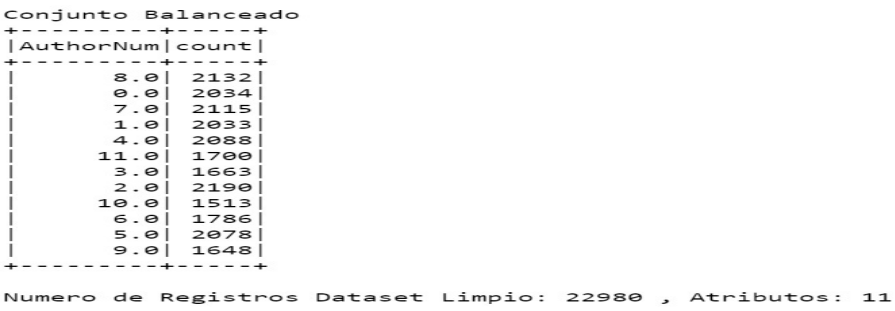

Figura 8: Balanceo de Datos

Descripción del conjunto de datos limpio:

- Consta de 11 atributos y 22980 registros
- Todas las columnas son de tipo numérico
- No contiene datos demasiado atípicos en ningún atributo
- Las categorías del atributo clasificador están balanceadas

3. Técnicas de aprendizaje automático

Las técnicas de aprendizaje automático que se describirán a continuación son técnicas de clasificación supervisada, es decir, son técnicas que dado un conocimiento a priori permitirán asignar un objeto a una de las categorías o clases

especificadas, las cuales son agrupaciones de objetos que tienen características comunes. El conocimiento a priori de cada uno de los modelos se obtiene gracias a un entrenamiento que se realiza con unos datos de muestra.
Las técnicas usarán el conjunto de datos Avila, por lo tanto lo que se quiere lograr es clasificar cada uno de los registros que contienen 10 características en uno de los 12 autores posibles, es decir, dadas ciertas características en particular sobre la escritura de un texto encontrar cuál de los 12 autores existentes fue el que escribió esa parte.

- **Regresión logística multinomial**: Este modelo se utiliza para predecir las probabilidades de que un registro se clasifique en una categoría o clase dado un conjunto de variables.
  Se asume K como el número de las clases o categorías, y un vector x de características, el modelo permite obtener la probabilidad pi que tiene la clase i en el conjunto de datos:

$$p_i = \frac{e^{z_i}}{\sum_{j=1}^{K} e^{z_j}} \quad \text{con} \quad z_i = \alpha_{i,1} x_1 + \alpha_{i,2} x_2 + \ldots + \alpha_{i,n} x_n + \beta_i$$

Figura 9: Fórmula de regresión logística multinomial

  La probabilidad más alta será la que determine a qué clase pertenece.

- **Árbol de decisión:** Es un modelo de predicción que usa un conjunto de datos para construir diagramas de construcciones lógicas, con el fin de representar y categorizar una serie de condiciones que ocurren sucesivamente para finalmente encontrar la clasificación.
  Para nuestro caso que estamos trabajando con el conjunto de datos Avila, los nodos contendrían los valores de los atributos: Intercolumnar distance, upper margin , lower margin, exploitation , row number, modular ratio, interlinear spacing, weight, peak number y modular ratio/ interlinear spacing. Y habrían 12 salidas, donde cada una contiene el nombre de cada uno de los autores existentes.

- **Random forest:** Usa la técnica de bagging, que predice a través de la combinación de los resultados de varios clasificadores, cada uno de ellos modelados con diferentes subconjuntos tomados de la misma población. Siguiendo la idea antes mencionada, entonces Random forest es una combinación de árboles de decisión tal que cada árbol recibe de forma aleatoria un subconjunto de los atributos con la misma distribución para cada uno de estos.
  Cada árbol de decisión realiza la clasificación, es decir vota por alguna clase o categoría, y el resultado es la clase con el mayor número de votos en todo el bosque(forest).

Para realizar la comparación de las técnicas, se evaluaron los resultados obtenidos por cada ténica con la métrica de desempeño llamada accuracy, la cual mide la exactitud del modelo. Accuracy mide el porcentaje de casos que el modelo ha acertado.

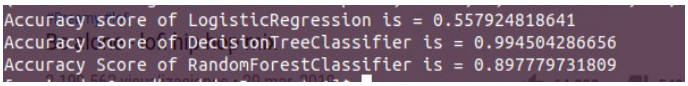

Figura 10: Resultados

El accuracy indica que el modelo que predijo con mayor exactitud fué el modelo de árbol de decisión, el cual indica que el 0.99 aproximadamente de las predicciones realizadas resultaron que estaban correctas.

4. Técnicas de aprendizaje automático

- **Streaming:**

Procesar los datos en tiempo real permite una mejor administración de los datos, debido a que no es necesario almacenar estos datos previamente, de modo que solo ocupan espacio aquellos datos relevantes para el problema que se está queriendo resolver. Es así como este procesamiento no solo permite administrar los datos de forma eficiente y eficaz, sino incluso de ahorrar costes de almacenamiento.A diferencia de lo que ocurre cuando ya se tienen los datos almacenados, los costes de esto serán altos, ya que se podría guardar información que no es útil para el problema.

Otra ventaja importante es que procesar y analizar estos datos en tiempo real permite identificar cualquier error al instante, mientras que si se procesa y analiza los datos con el dataset completo es un poco difícil descubrir el fallo y reaccionar de forma inmediata.

Para procesar los datos en tiempo real del conjunto Avila, se propone dividir el dataset en partes iguales, conteniendo registros escogidos al azar, y pasar cada una de estas divisiones al programa en tiempos distintos.

- **Graph Aplication:**

Para este caso se propone contruir el grafo de la siguiente forma:

* Se crearán X conglomerados de nodos representando las categorías a clasificar en el Dataset. (En este caso corresponde a las categorías de autores)

* Dentro de cada conglomerado se crearán n nodos representando los n registros que tienen como clasificación la categoría que su conglomerado representa.

* Cada nodo contendrá m datos representando los valores de sus respectivos registros en sus m atributos, exceptuando el atributo clasificador.(En este caso correspondería a los valores de los registros en los atributos: F1, F2, F3,..., F10)

* Dentro de cada conglomerado los nodos se conectarán de la siguiente manera: El nodo V se conecta con el nodo U si estos comparten el valor de al menos un atributo.

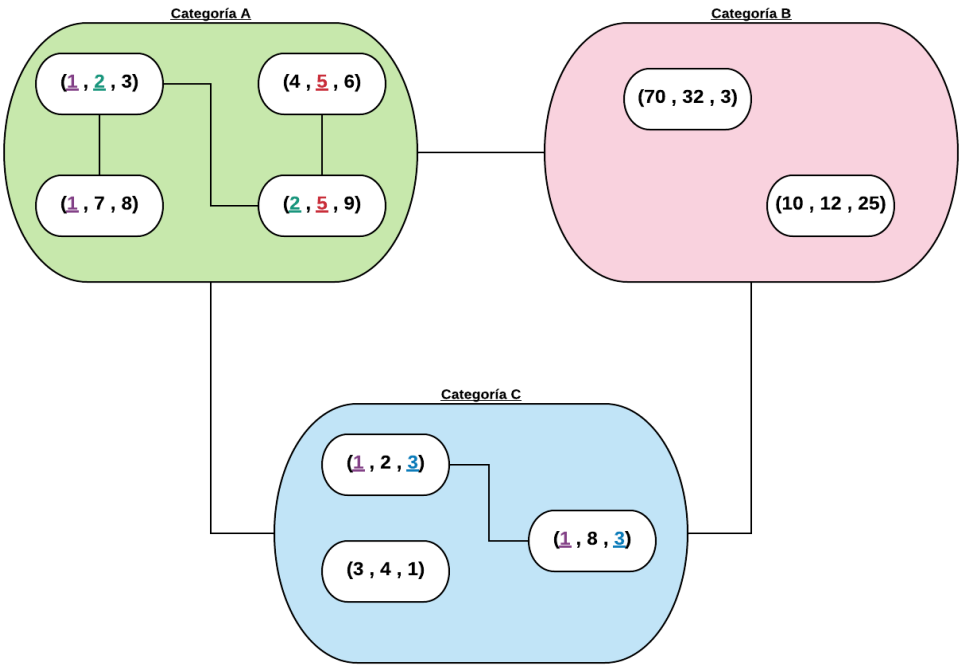

Figura 11: Ejemplo de Graph Aplication