ANÁLISIS COMPARATIVO DE LAS FUNCIONES DE PYTHON Y R UTILIZADAS EN LA CIENCIA DE DATOS

M. A. Ruiz Jaimes[1]
J. J. Flores Sedano[2]
Y. Toledo Navarro[3]
J. A. Ruiz Vanoye[4]

RESUMEN

El enfoque principal del artículo es la comparación de los lenguajes de programación Python y R utilizando los algoritmos de agrupamiento *K*-means y Fuzzy *C*-means. Para realizar esto, se tomaron 9 repositorios descargados de la UCI Machine Learning Repository, cada uno con diferentes tamaños. Para la evaluación de los algoritmos se desarrollaron dos prototipos, uno para cada lenguaje, cada uno de estos programas permiten seleccionar el repositorio que se desea usar, realizar el pre procesamiento necesario, seleccionar los datos del repositorio que se desea evaluar, para finalmente, procesar los algoritmos y mostrar los resultados. Con los resultados obtenidos se determinó que los lenguajes si son comparables entre sí y que además el lenguaje R es el más apto entre los dos para la ciencia de datos. Con el fin de que los estudiantes de ingeniería puedan observar que los dos lenguajes de programación más utilizados para la ciencia de datos tienen diferencias, se utilizaron dos de los algoritmos más famosos (K-means y Fuzzy c-Means) para optar por el que se adecue a las necesidades del análisis e interpretación de los datos.

ANTECEDENTES

En la actualidad, los problemas de análisis de datos, el manejo de una cantidad enorme de datos, así como, su interpretación y procesamiento son muy importantes para la ciencia de datos, un área que en los últimos años ha tomado gran relevancia para las empresas actuales, con esto surge una gran incógnita, es decir, cuál de todos los lenguajes de programación es el más apto para la ciencia de datos. El enfoque principal de este proyecto es comparar dos lenguajes de programación (Python y R) y determinar cuál es el más adecuado para la ciencia de datos, y esto se obtendrá haciendo uso de funciones de agrupamiento como el algoritmo *K*-means y algoritmo fuzzy *C*-means.

En este artículo se proponen dos prototipos que permitan comparar los lenguajes de programación Python y R, haciendo uso de algoritmos de agrupamiento, además de repositorios de datos con distintas instancias, y funciones de preprocesamiento con el fin de que los algoritmos sean capaces de utilizar los datos; la manera en la cual se evaluarán a los lenguajes de programación será:

- Por el tiempo de ejecución bajo las mismas condiciones.
- La calidad de los resultados.
- La precisión de los resultados.
- Número de iteraciones de cada algoritmo
- La capacidad para evaluar grandes cantidades de datos.

El objetivo principal de este trabajo es realizar un análisis comparando dos lenguajes de programación (Python y R) con sus funciones de agrupamiento de datos para determinar si los lenguajes son comparables entre sí.

[1] Profesor de Tiempo Completo. Universidad Politécnica del Estado de Morelos. mruiz@upemor.edu.mx
[2] Estudiante. Universidad Politécnica del Estado de Morelos. fsjo161286@upemor.edu.mx
[3] Profesor de Tiempo Completo. Universidad Politécnica del Estado de Morelos. ytnavarro@upemor.edu.mx
[4] Profesor de Tiempo Completo. Universidad Politécnica de Pachuca. jorge@ruizvanoye.com

La importancia de este proyecto es que acuerdo con la literatura especializada son dos los lenguajes de programación dominantes en los desarrollos de ciencia de datos, Python y R, sin embargo, existen problemas específicos de suma importancia que permitiran conocer cuál es más eficiente para problemas de agrupamiento. Para determinar esta eficiencia es necesario realizar pruebas experimentales para determinar cuál de los dos es el más apropiado para este tipo de aplicaciones en cuanto a calidad de la solución y tiempo de procesamiento.

Este estudio es de suma importancia para la formación de los estudiantes de Ingeniería en Tecnologías de la Información y Comunicación, de manera particular, cuando se requiere solucionar problemas de agrupamiento de datos, grandes cantidades de datos y análisis e interpretación de los datos, en el que se deberá elegir uno de los dos lenguajes de programación predominantes en el desarrollo de ciencia de datos.

Sin embargo, Python y R ofrecen herramientas que pemiten hacer análisis eficiente de datos y desarrollar algoritmos, lo que contribuye a la innovación educativa en el campo de la enseñanza de las Tecnologías de la Información y Comunicación.

METODOLOGÍA

La ciencia de datos emplea Big data, minería de datos y algoritmos para dar soluciones a preguntas concretas, permitiendo predecir tendencias en las personas y sus hábitos, esto puede aplicarse en muchas áreas, como el marketing, la salud, educación, Smart cities, y muchas más. Las compañías se han dado cuenta de la enorme cantidad de datos que se manejan hoy en día, y han observado que los científicos de datos son una gran ayuda para ganar ventaja competitiva (Kotu & Deshpande, 2019).

Técnicas de agrupación

El proceso de Clustering o agrupación consiste en la división de los datos en grupos de objetos similares. Para medir que tan similares son entre objetos, se suelen utilizar diferentes formas de distancia: distancia euclídea, de Manhatan, de Mahalanobis, etc. El representar los datos por una serie de clusters, conlleva la pérdida de detalles, pero consigue la simplificación de los mismos. Clustering es una técnica más de Machine Learning, en la que el aprendizaje realizado es no-supervisado (unsupervised learning) (Aggarwal, 2015).

Los algoritmos que pertenecen a esta técnica son:

- *K*-means.
- Fuzzy *C*-means.
- DBSCAN (Density-based Spatial Clustering of Applications with Noise, Agrupamiento Espacial de Aplicaciones con Ruido basado en Densidad).
- Agrupamiento Jerárquico Aglomerativo (Agglomerative Hierarchical Clustering).
- Mezcla de Gausianos (Mixture of Gaussians).

A continuación, se describen algunos de los pocos trabajos relacionados con esta investigación.

- Classification and regression trees (Loh, 2011).

- A Fast Clustering Algorithm to Cluster Very Large Categorical Data Sets in Data Mining (Huang, 1997).
- R Vs Python (Calltutors, 2019).

Los datos utilizados para realizar las pruebas son repositorios descargados de la página UCI, estos repositorios son de diferentes tamaños, con el fin de poder usar distintas instancias del conjunto de datos.

Los alcances y limitaciones del proyecto son los siguientes:
Alcances

- Implementación de prototipo en Python y R.
- Utilizar funciones de agrupamiento de datos en Python y R.
- Seleccionar los datos con los que se van a trabajar las funciones de agrupamiento.
- Realizar pruebas.
- Contrastar resultados de cada lenguaje.

Limitaciones

- Solo se usarán las siguientes funciones de agrupamiento de Python y R:
  - *K*-means.
  - Fuzzy *C*-means.
- No se busca eficiencia las funciones de agrupamiento.
- Los repositorios de datos no serán modificados más allá del pre procesamiento de datos aplicado.

En la Tabla 1 se muestran los resultados obtenidos por el procesamiento del algoritmo *K*-means en el lenguaje de programación R, donde se puede apreciar el tiempo de ejecución, el error cuadrático de cada iteración, así como el número de iteraciones de cada uno.

Tabla 1: *Resultados del Algoritmo K-means en el lenguaje de programación R*

CENTROIDES INICIALES	REPOSITORIO	TIEMPO	ERROR DE CADA ITERACIÓN	NÚMERO DE ITERACIONES
[20. 10. 30.] [15. 9. 33.] [19. 5. 40.]	Adult	0.0142	34.983175 34.588105 33.587251 33.587251 33.391559	5
[1. 0. 2.] [0. 1. 1.] [3. 1. 2.]	Sponge	0.00146	9.935484, 9.865431	2
[1. 0. 2.] [0. 1. 1.] [3. 1. 2.]	User Identification From Walking Activity Data	0.047	3.393769 3.373669 3.359669 3.353669 3.353432 3.311633	6
[1. 3. 2.1.] [0. 2. 1.2.] [3. 1. 2.1.]	Human Activity Recognition Data	0.2079	7.568771 7.504072 7.504048 7.404578	4

[1. 15] [0. 19] [3. 14] [4.10]	Individual household electric power consumption Data	0.020	5.020903 5.000402	2
[1.15] [0.5] [1.20] [0.13]	SCADI Data	0.001	7.312648 7.216628 7.132446 7.013437	4
[15, 0, 0] [24, 3, 1] [8, 0.1,3] [40, 0, 1]	Diabetes Data	0.025	4.499038 4.486271 4.486271 4.486213	4
[50, 100, 1,0] [90, 150, 3,1] [30, 90, 3,3] [20, 155, 2,0]	Wine Data	0.001	19.019636 18.910798 18.614594	3
[150, 100, 0] [90, 150, 3] [130, 90, 200] [120, 155, 250]	Improved Spiral Test Using Digitized Graphics Tablet for Monitoring Parkinson's Disease	0.002	10.378401 9.132691	2

Fuente: Elaboración propia

En la Tabla 2 se muestran los resultados obtenidos por el procesamiento del algoritmo *K*-means en el lenguaje de programación Python, donde se puede apreciar el tiempo de ejecución, el error cuadrático de cada iteración, así como, el número de iteraciones de cada uno.

Tabla 2: *Resultados del Algoritmo K-means en el lenguaje de programación Python*

CENTROIDES INICIALES	REPOSITORIO	TIEMPO	ERROR DE CADA ITERACIÓN	NÚMERO DE ITERACIONES
[20. 10. 30.] [15. 9. 33.] [19. 5. 40.]	Adult	0.3318	39.6493, 39.8036, 39.8390, 39.8666, 39.8974, 40.1928, 40.3529, 40.3785, 40.5133	8
[1. 0. 2.] [0. 1. 1.] [3. 1. 2.]	Sponge	0.00146	10.4329, 10.3466, 10.3266	3
[1. 0. 2.] [0. 1. 1.] [3. 1. 2.]	User Identification From Walking Activity Data	0.5909	15.2357, 15.2229, 15.2193, 15.2111, 15.2082, 15.1992, 15.1971, 15.1945, 15.1858,	12

			15.1156, 14.5946, 14.4333	
[1. 3. 2.1.] [0. 2. 1.2.] [3. 1. 2.1.]	Human Activity Recognition Data	17.4191	10.8422, 10.8388, 10.6978, 10.6919, 10.6281, 10.5261, 10.5255, 10.2460, 10.2284, 10.2053	9
[1. 15] [0. 19] [3. 14] [4.10]	Individual household electric power consumption Data	89.7849	11.9948, 11.0416, 10.3507, 10.3390	4
[1.15] [0.5] [1.20] [0.13]	SCADI Data	0.0295	10.7237, 10.6388, 10.6388, 10.5583, 10.5445	5
[15, 0, 0] [24, 3, 1] [8, 0.1,3] [40, 0, 1]	Diabetes Data	0.6093	16.7835, 16.7159, 16.6102, 16.4930, 16.2099, 16.1965	6
CENTROIDES INICIALES	**REPOSITORIO**	**TIEMPO**	**ERROR DE CADA ITERACIÓN**	**NÚMERO DE ITERACIONES**
[50, 100, 1,0] [90, 150, 3,1] [30, 90, 3,3] [20, 155, 2,0]	Wine Data	0.0298	37.2434, 37.1891, 37.0637, 36.9953, 36.9673	5
[150, 100, 0] [90, 150, 3] [130, 90, 200] [120, 155, 250]	Improved Spiral Test Using Digitized Graphics Tablet for Monitoring Parkinson's Disease	0.1392	35.1782, 35.0436, 35.0071, 34.9891, 34.9227, 34.7308	6

Fuente: Elaboración propia

En la Tabla 3 se muestran los resultados obtenidos por el procesamiento del algoritmo Fuzzy *C*-means en el lenguaje de programación Python, donde se puede apreciar el tiempo de ejecución, el error cuadrático de cada iteración, así como, el número de iteraciones de cada uno.

Tabla 3*: Tabla de resultados del Algoritmo Fuzzy C-means en el lenguaje de programación Python*

CENTROIDES INICIALES	REPOSITORIO	TIEMPO	ERROR DE CADA ITERACIÓN	NÚMERO DE ITERACIONES

[20. 10. 30.] [15. 9. 33.] [19. 5. 40.]	Adult	0.65 51	11.71328494, 11.51328491, 11.51028098, 10.21328498, 9.98212224480371	28
[1. 0. 2.] [0. 1. 1.] [3. 1. 2.]	Sponge	0.0197	7.88277724, 7.38208577, 7.08097044, 6.38297084, 6.24237284, 6.13494581, 6.08297084, 6.02067080, 5.38297084, 5.38297084	10
[1. 0. 2.] [0. 1. 1.] [3. 1. 2.]	User Identification From Walking Activity Data	0.5909	15.2357, 15.2229, 15.2193, 15.2111, 15.2082, 15.1992, 15.1971, 15.1945, 15.1858, 15.1156, 14.5946, 14.4333	12
[1.3.2.1] [0.2. 1.2.] [3. 1. 2.1.]	Human Activity Recognition Data	10.4839	10.8422, 10.8388, 10.6978, 10.6919, 10.6281, 10.5261, 10.5255, 10.2460, 10.2284, 10.2053	20
CENTROIDES INICIALES	**REPOSITORIO**	**TIEMPO**	**ERROR DE CADA ITERACIÓN**	**NÚMERO DE ITERACIONES**
[1. 15] [0. 19] [3. 14] [4.10]	Individual household electric power consumption Data	43.8101	12.01924440, 10.80968715, 10.70965853, 10.77994591, 10.60968715, … 10.10968715, 10.80607780,	60
[1.15] [0.5] [1.20] [0.13]	SCADI Data	0.0792	13.99089023, 13.85574523, 13.14793173, 13.19035026, 13.12536000, 13.09022013, 12.69039024, 12.36349064, 12.29054402, 12.24046024,	10

			12.24404063, 12.19687020, 12.14486003, 12.13486600, 12.11484033, 11.49089533, 11.46084773, 11.29089023, 11.19089023, 10.39386410	
[15, 0, 0] [24, 3, 1] [8, 0.1,3] [40, 0, 1]	Diabetes Data	1.5089	19.96560642, 19.94420472, 19.92707647, 19.84624640, 19.84627672, … 17.70560680, 17.54627740, 16.31525462, 16.29975044755212 8	40
[50, 100, 1,0] [90, 150, 3,1] [30, 90, 3,3] [20, 155, 2,0]	Wine Data	0.14	39.07748507, 39.07748507, 39.07748507, 39.07748507, 39.07748507, … 37.07748507, 37.07748507	100
[150, 100, 0] [90, 150, 3] [130, 90, 200] [120, 155, 250]	Improved Spiral Test Using Digitized Graphics Tablet for Monitoring Parkinson's Disease	0.3841	37.86650317, 37.86600364, 37.66607377, … 35.26600387, 35.04630390, 34.80223210429781	90

Fuente: Elaboración propia

En la Tabla 4 se muestran los resultados obtenidos por el procesamiento del algoritmo Fuzzy *C*-means en el lenguaje de programación R, donde se puede apreciar el tiempo de ejecución, el error cuadrático de cada iteración, así como, el número de iteraciones de cada uno.

Tabla 4: *Tabla de resultados del Algoritmo Fuzzy C-means en el lenguaje de programación R*

CENTROIDES INICIALES	REPOSITORIO	TIEMPO	ERROR DE CADA ITERACIÓN	NÚMERO DE ITERACIONES
[20. 10. 30.] [15. 9. 33.] [19. 5. 40.]	Adult	0.292	7.344339 7.030381 6.848397 … 4.407090 4.360823 3.977866 3.898627	24
[1. 0. 2.]	Sponge	0.0156	7.088498	10

[0. 1. 1.] [3. 1. 2.]			7.084043 6.094513 5.942212 5.852154 5.453905 5.096896 4.812762 3.352283 2.146316	
[1. 0. 2.] [0. 1. 1.] [3. 1. 2.]	User Identification From Walking Activity Data	0.047	3.393769 3.373669 3.359669 3.353669 3.353432 3.311633	6
[1. 3. 2.1.] [0. 2. 1.2.] [3. 1. 2.1.]	Human Activity Recognition Data	10.0455	7.886133 7.828418 6.756569 6.665537 6.121478 6.074436 6.054832 5.974742 5.959156 5.755969 5.427716 5.054679 4.430617 3.819700	14
[1. 15] [0. 19] [3. 14] [4.10]	Individual household electric power consumption Data	1.9508	12.213350 12.184591 11.677991 11.538813 … 5.258535 5.225330	40
[1.15] [0.5] [1.20] [0.13]	SCADI Data	0.001	11.855745 10.479317 9.180535 8.749111 8.709979 7.416136 7.290610 7.194510 5.906740 5.656664 5.001186 4.897485 4.813794	13
CENTROIDES INICIALES	**REPOSITORIO**	**TIEMPO**	**ERROR DE CADA ITERACIÓN**	**NÚMERO DE ITERACIONES**
[15, 0, 0] [24, 3, 1] [8, 0.1,3] [40, 0, 1]	Diabetes Data	0.9531	14.260949 12.050216 11.320679 11.206118 10.971396 …	40

			6.288214 5.842974 5.003361 4.856111 4.760534 4.554795 3.672846	
[50, 100, 1,0] [90, 150, 3,1] [30, 90, 3,3] [20, 155, 2,0]	Wine Data	0.003	18.386069 18.081435 17.980818 17.764782 17.536551 17.326915 16.267248 16.158348 15.932194 15.914314 15.772254 15.626400	67
[150, 100, 0] [90, 150, 3] [130, 90, 200] [120, 155, 250]	Improved Spiral Test Using Digitized Graphics Tablet for Monitoring Parkinson's Disease	0.003	18.386069 18.081435 17.980818 17.764782 17.536551 17.326915 16.267248 16.158348 15.932194 15.914314 15.772254 15.626400	67

Fuente: Elaboración propia

RESULTADOS

Con los resultados de las pruebas obtenidas, se observa que hay una diferencia significativa en el tiempo de ejecución, número de iteraciones y el error cuadrático de cada iteración, por lo que es factible una comparación de ambos lenguajes de programación, y que, de acuerdo con los datos obtenidos, el lenguaje R es más apto en cuanto a los lenguajes *K*-means y fuzzy *C*-means.

CONCLUSIONES

El trabajo de este artículo permite entender y analizar mejor los lenguajes de programación Python y R, cumpliendo con los objetivos planteados, cabe mencionar que este trabajo, permitirá en un futuro evaluar los lenguajes con más algoritmos de agrupamiento, así como, algoritmos de clasificación y regresión. También se entendió el papel fundamental que juega el preprocesamiento de datos que se realiza antes de procesar los algoritmos.

BIBLIOGRAFÍA

Aggarwal, C. (2015). *Data Mining: The Textbook*. New York: Springer

Calltutors (28 may, 2019). R Vs Python: Why Python preferred over R for data analysis? [blog]. Available from: https://www.calltutors.com/blog/r-vs-python-why-python-preferred-over-r-for-data-analysis/

Huang, Z. (1997). A fast clustering algoritm to cluster very large categorical data set in data mining. *Cooperative Research Center for Advanced Computational Systems*. Available from: http://citeseerx.ist.psu.edu/viewdoc/summary?doi=10.1.1.6.4718

Loh, W. (2011). Classification and Regression trees. En W. Pedrycz, *Wires Data Mining and Knowledge Discovery* (p. 14-23). Available from: https://onlinelibrary.wiley.com/toc/19424795/1/1

Kotu, V & Deshpande, B. (2019). *Data Science: Concepts and Practice*, (2nd ed). Cambridge: Morgan Kaufmann Publisher. Available from: http://asolanki.co.in/wp-content/uploads/2019/04/Data-Science-Concepts-and-Practice-2nd-Edition-3.pdf