Consensus Clustering aplicado a Mapas Auto-Organizados

F. Bobadilla[*1], N. Nanfara[*1], P. Pastore[1,2], EA. Fernandez[1,3]
[1]Fac. Ing. UCC, [2]DeepVisionAi, inc., [3]CIDIE-CONICET-UCC
[*]ambos autores deben ser considerados primer autor

Introducción

En el presente trabajo se expone una **implementación paralelizada** del algoritmo **Consensus Clustering** utilizando **Bootstrap como técnica de muestreo**, para el descubrimiento de la cantidad óptima de grupos para disgregar un conjunto de nodos entrenados con el algoritmo **SOM**.

Mapas auto-organizados

Los **mapas auto-organizados** consisten en un tipo de **red neuronal artificial** formada por una colección de **nodos** dispuestos en un **espacio bidimensional**. Utilizando **aprendizaje no supervisado**, el sistema logra detectar **patrones** en un **conjunto de datos** ingresado. El objetivo es la **exploración de grandes conjuntos de datos complejos**.

Para lograr una **correcta interpretación** de los resultados de un mapa auto-organizado, es necesario descubrir los **clusters** presentes en el mismo. El algoritmo más difundido para lograr esto es **K-means**. Su principal **desventaja** es que el usuario debe especificar la **cantidad de clusters** que espera encontrar, haciendo que **no sea objetiva**.

Consensus Clustering

Consensus Clustering es un método para determinar la **cantidad óptima de clusters presentes en los datos**, calculado en base a **medidas de consenso** y de **estabilidad**. La propuesta de esta técnica es el uso de **algoritmos de clustering no jerárquico** de una manera **reiterativa** sobre distintos **subconjuntos del espacio** de datos de entrada y con **diferentes cantidades de clusters** a formar.

Métodos

Conjunto de datos

Se utilizó un conjunto de datos creado con la utilidad **make_blobs**, de la librería para Python scikit-learn. Esta utilidad genera **conjuntos de datos separables** en una cantidad **predefinida** de **clusters**. Se utilizó un conjunto de datos con **3000 datos** y **8 atributos**, organizados en **5 clusters**.

Utilización de mapas auto-organizados

Al conjunto de datos generado se lo procesó con un **mapa auto-organizado** implementado en **Python 3**, utilizando principalmente la librería **NumPy** para soportar las operaciones matemáticas. Se configuró la red SOM con **30x30 nodos** dispuestos de forma **hexagonal**. Las **100 épocas** de entrenamiento se completaron en aproximadamente **2 minutos**.

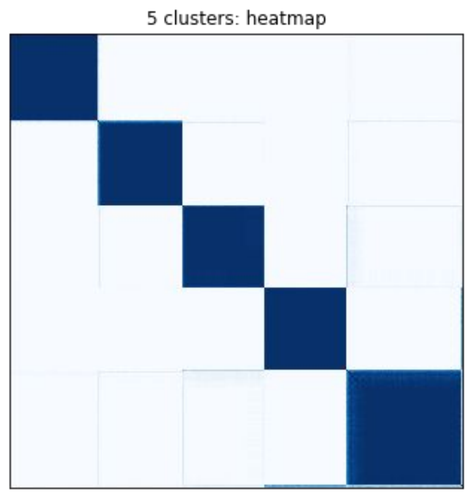
Utilización de Consensus Clustering

Se implementó una versión del algoritmo de Consensus Clustering con las características de: utilizar **muestreo Bootstrap** (muestras con reemplazo) como método de perturbación de datos y de inferencia estadística; utilizar **multiprocesos** para que, al utilizar múltiples núcleos de procesamiento, los tiempos requeridos sean menores.

De la conformación de la **matriz de consenso** para cada uno de los clusters buscados, se obtienen **histogramas** con sus respectivas **CDFs** y el cálculo de la variación del **área debajo de la curva**, los cuales permiten obtener de una manera analítica la justificación de la cantidad óptima de clusters obtenida. Además, esta matriz se representa mediante múltiples **heatmaps** (mapas de calor), para **visualmente** poder conocer cómo fueron conformados los clusters luego de todas las iteraciones de los muestreos.

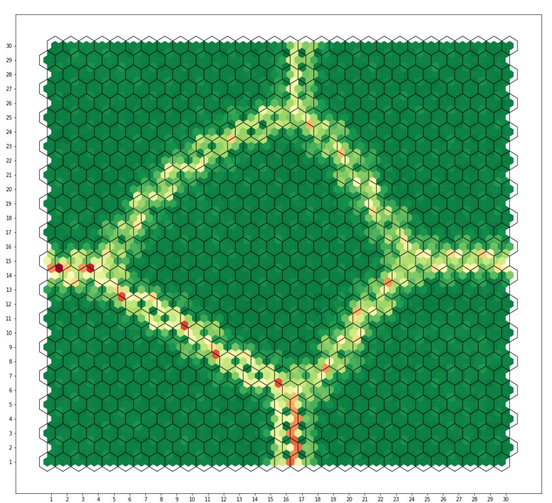
Al conjunto de nodos resultantes de aplicar SOM, se procedió con la determinación de cantidad de clusters óptimos mediante la implementación de Consensus Clustering. El algoritmo fue configurado para evaluar de **2 a 10 clusters**, con **200 muestras Bootstrap** para cada caso. Al hacer uso de **4 unidades de CPU**, el tiempo total del entrenamiento fue de **15 minutos y 20 segundos.**

Resultados



Tras obtener un conjunto de nodos resultante de aplicar la técnica **SOM** al conjunto de datos ya enunciado, se propone el uso de la implementación planteada del algoritmo de Consensus Clustering. De los resultados obtenidos, se presenta el gráfico del **heatmap** para la elección analítica de la cantidad óptima de clusters, éste muestra que los grupos conformados, lo hacen de una manera estable y consolidada, lo que permite validar de una manera visual el resultado obtenido.

Dado que con **Consensus Clustering** se pudo descubrir de manera **objetiva** que la **cantidad óptima de clusters es 5**, a la derecha se expone el resultado de **subdividir la red** en esa cantidad de clusters, junto con la matriz **U-matrix** de la red, para **comparar** ambas técnicas.



Se puede deducir que los **clusters** que pueden ser inferidos visualmente en la red SOM según la técnica **U-matrix** se **corresponden** con un **alto grado de exactitud** a los clusters obtenidos de aplicar la técnica **K-means con el valor de clusters óptimo** encontrado previamente con.

Los resultados obtenidos se **validan** en el hecho de que se conoce con anterioridad la cantidad de clusters en el conjunto de datos analizado, la cual es **coincidente** con el número obtenido aplicando la combinación de SOM y Consensus Clustering propuesta.

Conclusiones

- **Consensus Clustering** es válido para descubrir la **cantidad óptima de clusters** en un conjunto de datos
- También resulta útil como **complemento de la técnica SOM**, aportando más información sobre la **estructura de los datos analizados**.
- La técnica de muestreo **Bootstrap** resulta de utilidad para el usuario, al dejar de ser necesario especificar ciertos **parámetros**.
- El uso de **computación paralela** logró **acelerar el algoritmo** notablemente, con lo cual se hace viable su utilización para **grandes conjuntos de datos**.