Aplicación del algoritmo DBSCAN para la identificación de patrones en datos de rendimiento de jugadores de la NBA

Autor: Salomón García López

18 de noviembre de 2025

Resumen

Este trabajo presenta un estudio sobre la aplicación del algoritmo de agrupamiento no supervisado DBSCAN (Density-Based Spatial Clustering of Applications with Noise) a un conjunto de datos de rendimiento de jugadores de la NBA. El objetivo fue explorar estructuras subyacentes y perfiles de desempeño sin utilizar etiquetas predefinidas, a partir de estadísticas por partido como minutos jugados, puntos anotados, rebotes, asistencias, robos, tapones y la métrica Game Score. Los resultados muestran la formación de varios grupos distinguibles visualmente mediante reducción de dimensionalidad (PCA), en los que se identifican perfiles de anotadores de alto impacto, jugadores versátiles y especialistas defensivos, a pesar de que el algoritmo generó un número mayor de clústeres debido a la sensibilidad de sus parámetros. Se discute el comportamiento del modelo, la influencia del parámetro *eps* y la relevancia de DBSCAN en el análisis de datos deportivos.

1. Introducción

El análisis de datos deportivos profesionales permite descubrir patrones que apoyan la toma de decisiones tácticas, el diseño de alineaciones y la identificación de talento. En particular, la National Basketball Association (NBA) genera de forma continua información detallada sobre el desempeño de los jugadores: minutos jugados, porcentaje de tiro, uso de triples, rebotes, asistencias y otras estadísticas avanzadas. Este tipo de datos puede revelar perfiles de juego característicos que resultan útiles para el *scouting*, la planeación de rotaciones y la valoración de contratos.

El presente trabajo aplica un algoritmo de aprendizaje no supervisado, **DBSCAN**, con el propósito de identificar estructuras latentes en los datos de rendimiento de jugadores NBA durante la temporada 2024–2025. A diferencia de los métodos basados en centroides como K-Means, DBSCAN agrupa observaciones según la densidad de puntos en el espacio de características, lo que lo hace especialmente adecuado para datos con ruido o distribuciones irregulares, donde existen partidos extremadamente buenos o malos que se comportan como *outliers*.

2. Modelo matemático del algoritmo DBSCAN

DBSCAN [1] define los grupos en función de dos parámetros: *eps* (radio de vecindad) y *minPts* (número mínimo de puntos requeridos para formar un clúster).

Sea un conjunto de puntos $D = \{x_1, x_2, \ldots, x_n\}$ en un espacio métrico. Para un punto p, se define su vecindad ε como:

$$N_\varepsilon(p) = \{q \in D \mid dist(p, q) \leq \varepsilon\}.$$

Un punto p es considerado un *punto núcleo* si $|N_\varepsilon(p)| \geq minPts$. Los puntos que no cumplen este criterio pero se encuentran dentro de la vecindad de un punto núcleo son denominados *puntos frontera*. Aquellos que no pertenecen a ningún grupo se etiquetan como *ruido*.

El algoritmo forma clústeres expandiendo recursivamente las regiones de densidad alta. Formalmente, dos puntos p y q son *conectados por densidad* si existe una cadena de puntos núcleo que une a ambos dentro del radio ε. De esta manera, los clústeres corresponden a componentes conexas de alta densidad, mientras que los puntos aislados o muy dispersos se clasifican como ruido.

3. Metodología

3.1. Datos

Se trabajó con un conjunto de datos de partidos de jugadores de la NBA de la temporada 2024–2025. Cada registro corresponde al desempeño de un jugador en un partido, con variables como:

- **Player:** nombre del jugador.
- **Tm:** equipo del jugador.
- **Opp:** equipo rival.
- **MP:** minutos jugados (en formato decimal).
- **FG, FGA, FG %:** tiros de campo encestados, intentos y porcentaje.
- **3P, 3PA, 3P %:** triples encestados, intentos y porcentaje.
- **FT, FTA, FT %:** tiros libres encestados, intentos y porcentaje.
- **ORB, DRB, TRB:** rebotes ofensivos, defensivos y totales.
- **AST, STL, BLK:** asistencias, robos y tapones.
- **TOV, PF:** pérdidas de balón y faltas personales.
- **PTS:** puntos anotados.
- **GmSc:** Game Score, métrica que resume el impacto del jugador en el partido.

Para el agrupamiento se consideraron únicamente las variables numéricas de rendimiento por partido, excluyendo identificadores textuales como el nombre del jugador o el equipo.

3.2. Preprocesamiento

Antes del agrupamiento, las variables numéricas fueron escaladas con *StandardScaler* de la biblioteca `scikit-learn` [2], con el fin de normalizar la influencia de magnitudes y poner todas las características en la misma escala (media cero y desviación estándar uno). Esta etapa es fundamental para algoritmos basados en distancias, como DBSCAN, ya que variables con escalas muy grandes pueden dominar la métrica.

Posteriormente se aplicó una reducción de dimensionalidad mediante Análisis de Componentes Principales (PCA) con dos componentes [3], únicamente con fines de visualización. El modelo de agrupamiento se entrenó en el espacio original estandarizado, mientras que la proyección PCA se utilizó para representar los clústeres en un plano bidimensional.

3.3. Implementación de DBSCAN

Se utilizó el algoritmo DBSCAN disponible en la librería `scikit-learn`. El parámetro *eps* se determinó mediante el método de la **k-distancia**, que consiste en:

1. Calcular, para cada punto, la distancia a su k-ésimo vecino más cercano.
2. Ordenar dichas distancias de menor a mayor.
3. Graficar la curva resultante y localizar el punto de inflexión o "codo".

En este trabajo se utilizó $k = 4$, que coincide con el valor de *minPts* empleado posteriormente en DBSCAN. El punto de cambio significativo se observó alrededor de **eps = 1.5**, como puede apreciarse en la figura 1. Para valores inferiores, muchos puntos quedan etiquetados como ruido y el número de clústeres disminuye abruptamente; para valores muy superiores, los clústeres se fusionan en uno solo. Por ello se adoptó $eps = 1{,}5$ como compromiso razonable.

El parámetro *minPts* se estableció en 4, valor comúnmente recomendado como mínimo para bases con ruido moderado, y coherente con el número de dimensiones utilizadas.

4. Resultados

El algoritmo identificó varios clústeres al usar $eps = 1{,}5$ y $minPts = 4$, además de un conjunto de observaciones etiquetadas como ruido (jugadores o partidos con estadísticas atípicas). Para interpretar la estructura de los grupos se proyectaron las observaciones en el plano de los dos primeros componentes principales (PCA), como se muestra en la figura 2.

A partir de esta proyección se observan varias regiones de alta densidad correspondientes a perfiles de desempeño diferenciados. Para caracterizar cada grupo se calcularon los promedios por clúster de variables clave como puntos anotados (PTS), Game Score (GmSc), rebotes totales (TRB), asistencias (AST), robos (STL) y tapones (BLK). Esta información permite asociar cada clúster con un tipo de jugador o partido representativo (por ejemplo, anotadores de alto volumen, jugadores completos o especialistas defensivos).

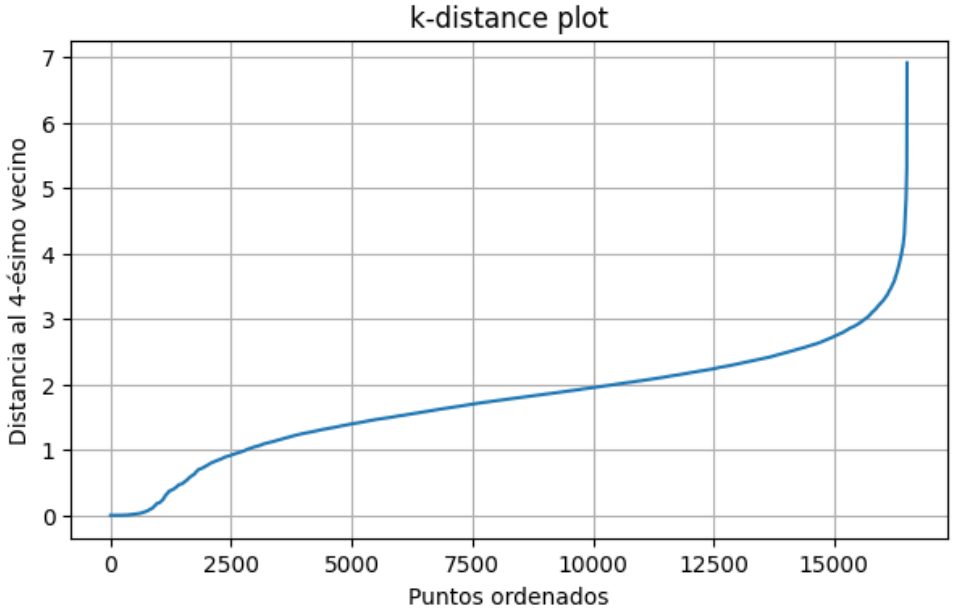

Figura 1: Curva de k-distancia para $k = 4$ en los datos de rendimiento NBA. El cambio de pendiente alrededor de 1.5 sugiere un valor adecuado de *eps*.

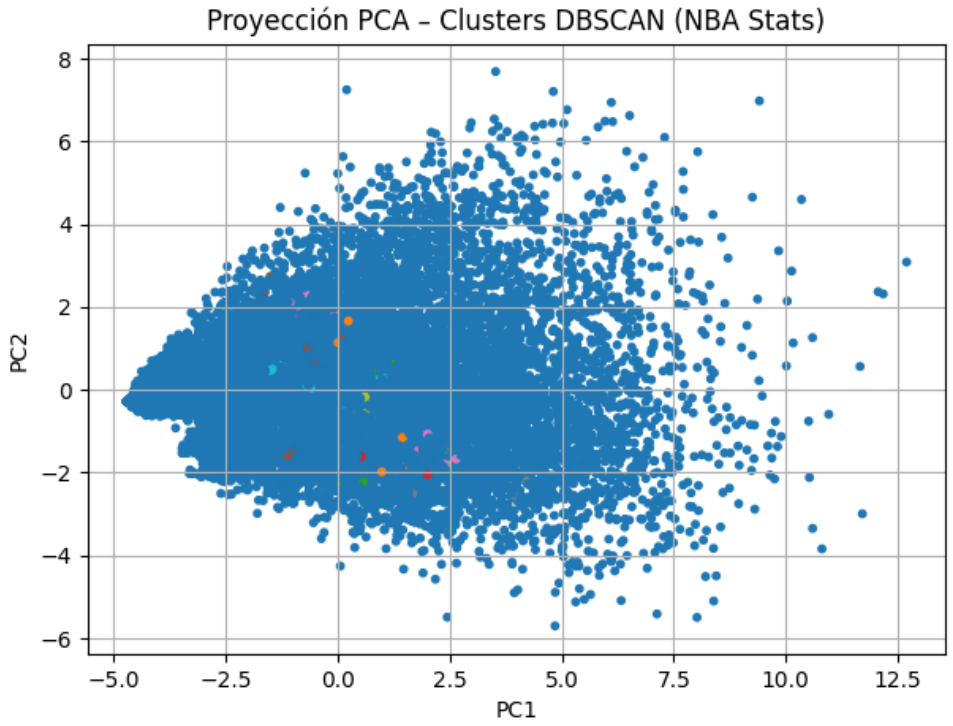

Figura 2: Visualización PCA de los clústeres detectados por DBSCAN sobre estadísticas de jugadores de la NBA. Cada color representa un clúster distinto; los puntos etiquetados como ruido aparecen con un color separado.

5. Discusión

La aplicación de DBSCAN permitió detectar estructuras que difícilmente serían capturadas por métodos basados en centroides. El modelo distingue regiones de alta densidad en torno a ciertos perfiles estadísticos y separa como ruido aquellos partidos muy atípicos, ya sea por rendimiento extraordinario o por minutos de juego muy reducidos.

Entre los clústeres más relevantes se identifican:

- **Anotadores de alto impacto**, con valores elevados de PTS y GmSc, generalmente asociados a jugadores con alto volumen de tiro.
- **Jugadores versátiles**, con valores balanceados de PTS, TRB y AST, que contribuyen en varias facetas del juego.
- **Especialistas defensivos**, caracterizados por mayores promedios de STL y BLK, aunque con menor carga ofensiva.
- **Partidos de bajo impacto**, donde los jugadores disputan pocos minutos, con estadísticas modestas, que en algunos casos son clasificados como ruido por el algoritmo.

El método de k-distancia resultó adecuado para determinar el valor de *eps*, al mostrar un cambio claro en la pendiente de la curva alrededor de 1.5. No obstante, la elección de este parámetro sigue teniendo un componente heurístico y puede requerir ajuste según el criterio del analista y el objetivo del estudio (por ejemplo, privilegiar menos ruido o menos clústeres).

DBSCAN demostró ser eficaz para datos deportivos, donde la forma de los grupos no necesariamente es esférica y donde la presencia de partidos extremos es frecuente. Sin embargo, su desempeño depende críticamente de la correcta elección de *eps* y *minPts*, lo que limita su automatización sin un análisis gráfico o heurístico complementario.

6. Conclusiones

El uso de DBSCAN en datos de rendimiento de jugadores de la NBA permitió identificar patrones de desempeño que se agrupan de manera natural sin necesidad de etiquetas previas. Los clústeres obtenidos reflejan perfiles interpretables de jugadores, como grandes anotadores, jugadores completos y especialistas defensivos, mientras que los partidos muy atípicos se separan como ruido.

Este resultado ilustra cómo los algoritmos basados en densidad pueden capturar relaciones complejas en datos reales y aportar valor al análisis de rendimiento deportivo. Como trabajo futuro se propone comparar los resultados de DBSCAN con otros algoritmos no supervisados, como OPTICS o Spectral Clustering, así como incorporar información temporal (secuencia de partidos) para estudiar la evolución del rendimiento a lo largo de la temporada.

Referencias

[1] M. Ester, H. P. Kriegel, J. Sander y X. Xu, "A Density-Based Algorithm for Discovering Clusters in Large Spatial Databases with Noise," *Proceedings of the 2nd International Conference on Knowledge Discovery and Data Mining (KDD-96)*, 1996.

[2] Scikit-learn developers, “sklearn.cluster.DBSCAN,” *Scikit-learn documentation*, 2024. Disponible en: `https://scikit-learn.org/stable/`.

[3] I. T. Jolliffe, *Principal Component Analysis*, 2nd ed. Springer, 2002.

[4] J. Hollinger, “Game Score Metric,” *ESPN Basketball Analytics*, 2011.