XVII Jornadas de Ingeniería Telemática
JITEL 2025
Universidad de Extremadura

Actas de las XVII Jornadas de Ingeniería Telemática (JITEL 2025), Cáceres (España), 12-14 de noviembre de 2025.

ISBN

Un enfoque simplificado de trazabilidad de criptoactivos con Page Rank RWR, PCA y K-means

Volodymyr Dubetskyy, Álvaro Hernández Riquelme, Adrián Ortega Escalona,
Pilar Manzanares López, Maria-Dolores Cano
Departamento de Tecnologías de la Información y las Comunicaciones
Universidad Politécnica de Cartagena

volodymyr.dubetskyy@upct.es, alvaro.hriquelme@upct.es, adrian.ortega@upct.es,
pilar.manzanares@upct.es, mdolores.cano@upct.es

El auge de las criptomonedas ha multiplicado los riesgos de blanqueo, exigiendo técnicas de trazabilidad que funcionen sin datos etiquetados. Este trabajo aborda la clasificación no supervisada de direcciones Bitcoin en el dataset Elliptic++, explotando únicamente la estructura del grafo transaccional. Aplicamos un PageRank con reinicio aleatorio personalizado para calcular dos probabilidades originales de proximidad a nodos lícitos e ilícitos. Estos vectores alimentan un clustering K-Means con k = 3. El sistema logra un índice Silhouette de 0,955, superando configuraciones con reducción de dimensionalidad y evitando el coste de modelos profundos. La incorporación de PCA no mejora la cohesión y un trimmed K-Means solo eleva la pureza de forma marginal mientras reduce la densidad interna, confirmando la autosuficiencia de las características derivadas de PageRank. La propuesta demuestra que la topología de la red basta para discriminar patrones ilícitos de forma interpretable y con mínimos requisitos computacionales, lo que facilita su adopción en entornos regulatorios.

***Palabras Clave*—bitcoin, blockchain, cryptocurrencies, cybersecurity, forensics, page rank, traceability**

I. INTRODUCCIÓN

Durante la última década, la capitalización agregada de los criptoactivos ha superado con holgura el billón de dólares estadounidenses (USD). Solo Bitcoin [1] ronda ya los USD 1,6 billones de valor de mercado [2], consolidándose como un activo financiero sistémico. En paralelo, redes de pago globales como Visa procesan cientos de millones en compras liquidadas directamente on-chain y han integrado esta modalidad en su tesorería [3]. Como aspecto negativo, este grado de adopción ha ido acompañado de un repunte de fraude, esquemas Ponzi y financiación ilícita. La Financial Action Task Force (FATF) sitúa hoy la monitorización on-chain en el núcleo de los marcos Know-Your-Customer / Anti-Money-Laundering (KYC/AML), con directrices específicas para los Virtual Asset Service Providers (VASP) y la Travel Rule [4].

Para detectar actividad ilícita en la red, la investigación se ha centrado en Graph Neural Networks (GNN) aplicadas al grafo de transacciones de Bitcoin. Estos modelos aprenden representaciones que permiten clasificar direcciones de monederos electrónicos (wallets), que son las claves públicas que identifican cada monedero en la red Bitcoin, según su comportamiento. Weber et al. [5] mostraron que un Graph Convolutional Network (GCN) alcanza un F1-score de 0,83 en el dataset Elliptic, y posteriores variantes basadas en Graph Attention Networks (GAT/GAT-ResNet) mejoran esa cifra [6], [7]. Sin embargo, las GNN exigen miles de nodos etiquetados, hardware GPU dedicado y enfrentan un desequilibrio de clases extremo (aproximadamente 92 % transacciones lícitas frente a menos del 8 % ilícitas), lo que dificulta su adopción industrial [8].

Las aproximaciones no supervisadas eliminan la dependencia de etiquetas, pero la literatura vigente suele emplear vectores de atributos locales o experimentaciones reducidas. Técnicas como k-medoids o spectral clustering se aplican sobre subconjuntos de transacciones sin explotar la conectividad global ni medir con rigor la coherencia de los clústeres [9].

Este trabajo aborda la clasificación no supervisada de direcciones Bitcoin empleando solo la topología del grafo transaccional. Las direcciones de wallet son los vértices del grafo transaccional sobre los que aplicamos el método propuesto. El flujo consta de dos etapas. Primero, aplicamos PageRank con reinicio aleatorio personalizado (Random Walk with Restart, RWR) [10], [11] para pro-

pagar la influencia de un reducido conjunto de semillas lícitas e ilícitas y obtener dos puntuaciones (pl, pi) que cuantifican la proximidad topológica de cada dirección a cada tipo de conducta, lícita o ilícita, respectivamente. Después, agrupamos esas puntuaciones mediante K-Means con k=3 (lícito, ilícito, desconocido). Probamos este proceso en Elliptic++, ampliación pública del corpus Elliptic con más de 200 000 transacciones etiquetadas [6], [12]. El modelo alcanza un índice Silhouette de 0,955 sin reducción de dimensionalidad ni aprendizaje profundo. Además, los resultados muestran que la inclusión de un Análisis de Componentes Principales (PCA) no mejora la cohesión y la variante trimmed K-Means solo incrementa marginalmente la pureza, reduciendo la densidad interna de los grupos. Así, las principales contribuciones de este trabajo son:

1) Marco interpretable y ligero que demuestra que dos características derivadas de PageRank bastan para discriminar actividad ilícita con coste mínimo.

2) Evaluación comparando configuraciones sin PCA, con 2 y 4 componentes principales y con trimmed K-Means, evidenciando el impacto negativo de la reducción de dimensionalidad.

3) Se evidencia que las GNN no son imprescindibles para la trazabilidad cripto, aunque como se indica en la discusión de los resultados, este trabajo tiene limitaciones que serán abordadas en trabajos futuros.

La Sección 2 describe el dataset Elliptic++ y formaliza PageRank personalizado, PCA como técnica comparativa y K-Means. La Sección 3 presenta y discute los resultados. El documento finaliza con las conclusiones y líneas futuras, incluida la extensión a otras cadenas públicas y la incorporación de dinámicas temporales.

II. METODOLOGÍA

A. *Dataset Elliptic++*

Para el análisis, se ha decidido usar el Dataset de *Elliptic++* gracias a su gran volumen de direcciones clasificadas según tienen una naturaleza lícita, ilícita, o desconocida. El principal desafío de trabajar con direcciones es su falta de características, por lo que es conveniente trabajar con sus interacciones y comportamientos.

El dataset Elliptic++ constituye un recurso fundamental para la investigación en trazabilidad de criptoactivos. Contiene una cantidad considerable de transacciones de Bitcoin etiquetadas manualmente, organizadas en un grafo temporal donde los nodos representan transacciones y las aristas flujos monetarios. Su implementación garantiza reproducibilidad y proporciona un benchmark para futuras investigaciones en detección no supervisada de fraudes blockchain. En nuestro caso, para la simplificación del análisis, se ha optado por trabajar únicamente con las direcciones de wallet, que son las claves públicas que identifican cada monedero en la red Bitcoin. Estas direcciones son distintas a las de los nodos del grafo transaccional, por lo que se formará un grafo distinto y las aristas representarán las transacciones entre ellas.

Como el enfoque principal son las direcciones de wallets, habrán tablas y datos que no se usen del Dataset, como es el caso del grafo de interacciones de *Actors*, ya que no proporciona conexiones directas entre direcciones, y agrupa direcciones que podrían ser omitidas. Se usará el grafo de interacciones de *Address-Transaction* que contiene tablas cuyas columnas aportan información relevante para este tipo de algoritmos. En las Tablas I, II y III se muestran los datos más relevantes del dataset para estos algoritmos, que contienen las direcciones de wallet, e información adicional como sus interacciones con otras wallets y características. La tabla I contiene las conexiones entre direcciones con origen y destino, lo que consideramos como una transacción, aunque el análisis de naturaleza no se hace en las transacciones sino en las direcciones de wallet. La tabla II contiene la naturaleza de cada dirección, que puede ser lícita, ilícita o desconocida, y la tabla III contiene las características de cada dirección, como el número de transacciones realizadas, el grado de entrada y salida, o la media de comisiones pagadas, que nos serán útiles para la agrupación posterior.

Tabla I
ADDRADDR_EDGELIST.CSV

input_address	output_address
1A1dfg...kNa	3J9...34Ly
3J...89Az	bc1...aat4

Tabla II
WALLETS_CLASSES.CSV

address	class
1A1dgk..zP1a	lícita
3J98t...WNLy	ilícita
bc1q...rt4	desconocida

Tabla III
WALLETS_FEATURES.CSV

address	tx_count	in_degree	out_degree	...	fees_mean
1A1z...fNa	150	75	75	...	0.00012
3J9...NLy	200	100	100	...	0.00023
bc1q...8f3t4	120	60	60	...	0.00015

B. *Algoritmo PageRank RWR*

El algoritmo que se plantea es una especialización del PageRank RWR (*Random Walk with Restart*), basado en el algoritmo de PageRank original de Google [10] para clasificar páginas web según una importancia recibiendo un peso cada enlace. En este caso, se escoge una semilla con direcciones con la naturaleza definida, con un peso de 1 para cada dirección. Tras la primera iteración, las semillas iniciales se descartan y los nodos que han ganado en peso pasan a ser la nueva semilla. El peso es reducido con un factor de amortiguación α, lo que da mayor valor a los nodos pertenecientes a las primeras iteraciones.

Para empezar, se denotará la definición del estado de la red como S, un conjunto de nodos que actúa como la semilla. En cada iteración el conjunto de nodos S

es sustituido por las direcciones visitadas en la iteración anterior. Por tanto, en caso de que en la primera iteración se escoja como semilla el conjunto de direcciones ilícitas, $S^0 = \{I \in V\}$ y como para cada iteración se descarta la anterior, $S^i = \{U^i \in V | U^i \notin I\}$, donde V es el conjunto de direcciones del dataset y U^i el conjunto de direcciones que se comunica con la semilla de la iteración i.

Para cada estado, se forma una matriz de adyacencia A, donde A_{ij} será 1 si hay una conexión desde el nodo i hasta el nodo j, y 0 en caso contrario. Con estos resultados se forma la matriz diagonal D que representa el grado de salida para cada nodo, siendo $D_{kk} = \sum_i A_{ik}$ y $D_{ij} = 0$ en caso contrario.

Con estos valores, se podrá calcular la matriz de transición, que contiene las probabilidades de conexión de un nodo a otro nodo, con un factor de amortiguación de 0.85 en cada iteración, se definirá como:

$$W = \alpha(D^{-1}A) \tag{1}$$

El factor de amortiguación α juega un papel crucial en la distribución de estas probabilidades, ya que determina cuánto peso se transmite en cada iteración. Un valor más alto de α permitirá que la influencia se propague más lejos en la red, mientras que un valor más bajo concentrará el peso en los nodos más cercanos a las semillas iniciales.

Finalmente, se podrá calcular el vector de probabilidades p buscado, considerando la probabilidad inicial de las semillas:

$$p_i^0 = \begin{cases} \frac{1}{|S(0)|} & \text{si } i \in S(0) \\ 0 & \text{resto} \end{cases} \tag{2}$$

Normalizando y partiendo de p0, la probabilidad en las siguientes iteraciones se define como:

$$p_i^{t+1} = \begin{cases} \frac{1}{|W^T p^t|} & \text{si } i \in S^{(t+1)} \text{ y } |S^{(t+1)}| > 0 \\ 0 & \text{resto} \end{cases} \tag{3}$$

Se obtendrá un vector de probabilidades distinto para las semillas ilícitas y lícitas, por lo que las direcciones desconocidas tendrán de características p_l y p_i, que se emplearán como referencia para la clasificación. Una dirección con un alto valor de p_l y bajo p_i sugiere que tiene más interacciones con direcciones lícitas conocidas, mientras que lo contrario indicaría una mayor conexión con actividades ilícitas. Mientras que, si ambas son de alto valor o bajo valor, podría significar una naturaleza desconocida o una dirección con mucha actividad como mixers o exchanges. Esta dualidad de características permite crear un espacio bidimensional donde cada dirección puede ser representada como una coordenada, facilitando la aplicación posterior de algoritmos de clustering.

El vector de probabilidades obtenido se usará posteriormente como características para la agrupación de direcciones o para la reducción de dimensionalidad, junto a las características base que aparecen en el dataset. En la Fig. 1 se muestra la distribución teórica de los pesos en la red, donde los nodos principales son más importantes y los periféricos van perdiendo peso.

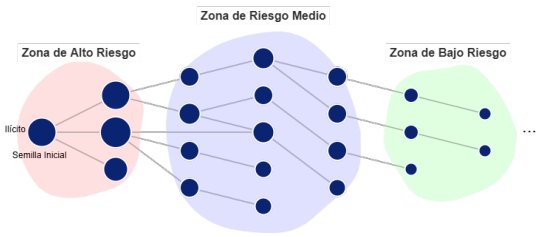

Figura 1. Distribución teórica de PageRank.

C. *Algoritmo PCA*

Como se menciona en la introducción, la escasez de características intrínsecas asociadas a las direcciones de wallets plantea un desafío significativo para el análisis, ya que, a diferencia de los nodos transaccionales que incluyen atributos financieros detallados, las direcciones carecen de metadatos descriptivos. Para abordar esta limitación, se implementa el Análisis de Componentes Principales (PCA), una técnica de reducción de dimensionalidad que transforma variables correlacionadas en un conjunto ortogonal de componentes principales. Estos componentes capturan la máxima varianza de los datos originales mediante una proyección lineal, priorizando la información más discriminativa mientras minimiza el ruido y la redundancia.

Aunque PCA mejora la eficiencia computacional y mitiga correlaciones espurias, su aplicación conlleva inconvenientes. Por un lado, simplifica la estructura de datos, facilitando la convergencia de K-Means y reduciendo el riesgo de sobreajuste. En cambio, la compresión dimensional puede eliminar información relevante no linealmente correlacionada, lo que —como revelan posteriormente los resultados— degrada la cohesión de los clústeres en este contexto específico. Esta dualidad subraya la necesidad de comparar sus resultados frente a enfoques basados únicamente en características topológicas.

Las características que más han aportado al algoritmo vienen en el dataset con las direcciones, y son algunas como `btc_transacted_max` para la cantidad maxima de BTC enviada o `fees_mean` para la media de commisiones de red pagadas.

A falta de determinar el número de componentes óptimo, comentado en la sección de resultados, este algoritmo no sólo sirve para reducir las dimensiones, si no que facilita algunos datos que se usarán en conjunto con K–Means, reduciendo ruido y correlaciones que pueden afectar a la agrupación, junto a mejoras de la estabilidad, reduciendo la probabilidad de convergencia en mínimos locales. [9] [13]

D. *Agrupación con K–Means*

El algoritmo de K–Means es un método de agrupación no supervisado que busca dividir un conjunto de datos en k grupos o clústeres, donde cada clúster se representa por

su centroide. El objetivo es minimizar la varianza dentro de cada clúster y maximizar la varianza entre los distintos clústeres. Se intentará agrupar los nodos desconocidos en tres grupos según la naturaleza y obtener el menor número de desconocidos reales.

Cada clúster C_j se representa por un centroide μ_j, que es el punto "medio" de todos los datos asignados a ese clúster. Formalmente, si $C_j = \{x_i \,|\, \text{etiqueta}(x_i) = j\}$, el centroide se define como

$$\mu_j = \frac{1}{|C_j|} \sum_{x_i \in C_j} x_i, \quad \mu_j \in \mathrm{R}^d \tag{4}$$

Cada iteración de K–Means consta de dos fases:

- *Asignación:* cada punto x_i se asigna al clúster cuyo centroide μ_j minimiza $\|x_i - \mu_j\|$.
- *Actualización:* recalcular cada μ_j usando la fórmula anterior con los puntos recién asignados.

Como los centroides parten de posiciones aleatorias, diferentes semillas pueden llevar a converger a soluciones distintas (mínimos locales).

En trimmed K–Means, tras excluir los puntos más lejanos se recalculan los centroides solo con el subconjunto "internalizado", lo que incrementa la robustez frente a outliers, ya que μ_j no se ve desplazado por valores extremos.

Finalmente se comparará las métricas de evaluación con distintos componentes de PCA, y la diferencia de resultados entre K–Means y trimmed K–Means.

III. RESULTADOS

A. Resultados tras la ejecución de PageRank

Una vez ejecutado el algoritmo de PageRank RWR, se obtienen dos vectores de probabilidades p_l y p_i que representan la proximidad de cada dirección a las semillas lícitas e ilícitas, respectivamente. Estos vectores se normalizan para que la suma de sus valores sea 1, facilitando su interpretación como probabilidades. Es importante destacar que el vector de probabilidades aplica también para las semillas, puesto que es una forma de representar las direcciones totales del Dataset y su naturaleza, aunque la importancia y los resultados de los distintos algoritmos se obtienen para las direcciones desconocidas, que son las que no tienen una naturaleza definida en el dataset.

En la Fig. 2 se puede observar la distribución de direcciones según tengan mayor peso en cada clasificación. La zona central, con mayor probabilidad de ser desconocida, estará compuesta por nodos con muy bajo peso (o ninguno) en los dos posibles, mientras que en las demás existirá una descompensación entre un tipo y otro, siendo la zona de la derecha, donde se encuentran las lícitas, más poblada.

B. Resultados tras la ejecución de PCA

A continuación, se procede a aplicar PCA con respecto a las características del dataset, que son las que se muestran en la Tabla III. El objetivo es reducir la dimensionalidad de los datos y extraer las características más relevantes para la agrupación posterior con K–Means. No se han usado las características p_l y p_i obtenidas de PageRank, ya que se considera que no están correlacionadas con las demás características del dataset, por lo que se tomará como características por separado.

La Fig. 3 muestra una comparativa de número de componentes con respecto a la varianza acumulada, con la finalidad de encontrar un número óptimo de componentes que represente la mayoría de la varianza. Se puede observar que los componentes más importantes son el primero y el segundo componente, ya que entre estos dos se consigue un 40 % de varianza acumulada, mientras que los demás aportan menor cantidad. Ampliando de 2 a 4 componentes, se alcanzaría una varianza acumulada de mas del 50 %. A partir de ahí, y no se estaría añadiendo información relevante suficiente para justificar la reducción de precisión que se introduciria a K–Means.

C. Implementación y función objetivo

Una vez calculada toda la información necesaria, el algoritmo de K–Means se aplicará al espacio de características definido por las dos probabilidades (p_l, p_i) obtenidas con PageRank y las m componentes principales seleccionadas por PCA. También se fijará $k = 3$ grupos (lícito, ilícito y mixto/desconocido) con varias iteraciones desde la inicialización para mitigar mínimos locales.

En trimmed K–Means se introduce un parámetro de recorte α, con α = 0.2, que en cada iteración descarta el $\alpha\,\%$ de puntos más alejados de sus centroides.

En la implementación del algoritmo, se obtiene una etiqueta para cada x_i que marca al clúster que se le ha asignado. La Fig. 4 muestra la clasificación obtenida con K–Means, considerando los dos primeros componentes y las probabilidades de PageRank. Se trata de una representación tridimensional de los datos, mostrando $PC2$, p_i y p_l. Se puede observar la importancia de PageRank en la separación de clústers, ya que los centroides de los clústeres se encuentran en una zona donde las direcciones lícitas tienen un alto valor de p_l y bajo p_i en el caso de las lícitas, mientras que en las ilícitas pasa de forma contraria.

Una vez clasificados, es conveniente usar métricas como Silhouette score o pureza global para evaluar la calidad de los clústeres obtenidos. La pureza global se define como la proporción de puntos correctamente clasificados en cada clúster. En este caso, también se añadirá métricas como Calininski-Harabasz y Davies-Bouldin, que miden la separación entre clústeres y la compactación dentro de ellos.

D. Resultados esperados y métricas de evaluación

La tabla IV contiene la comparación entre las 4 métricas para los casos estudiados, esto es considerando 0, 2 y 4 componenetes principales, ya que según las conclusiones obtenidas en la figura 3, son las opciones que más pueden aportar sin reducir demasiado la precisión. Para todas estas opciones, se empleará K-means y trimmed K-means. Se puede observar como los mejores resultados se obtienen al no considerar componentes principales (m=0). En el caso de usar PCA, se observa que la pureza global y el Silhouette score disminuyen al usar la variante trimmed,

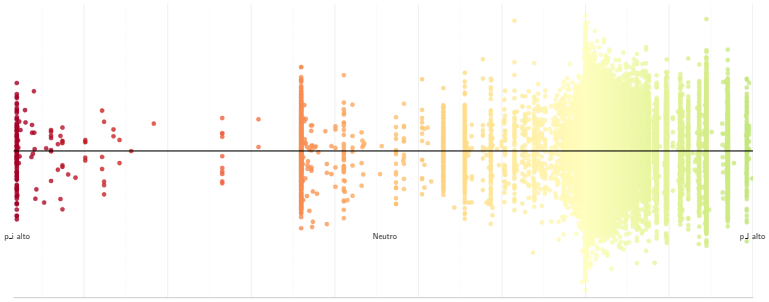

Figura 2. Distribución de conexiones con semilla ilícita.

mientras que el índice Calinski-Harabasz aumenta, lo que indica una mejor separación entre clústeres. Estos resultados sugieren que, para el caso propuesto, la reducción de dimensionalidad, mediante la reducción de características, no llega a mejorar una métricas basadas únicamente en los pesos de PageRank.

Tabla IV
COMPARATIVA: K–MEANS VS. TRIMMED K–MEANS

Métrica	K–Means	trimmed K–Means
Silhouette score (NO PCA)	**0.955**	0.941
Silhouette score (PCA 2)	0.868	0.891
Silhouette score (PCA 4)	0.771	0.667
Pureza global (NO PCA)	0.712	**0.715**
Pureza global (PCA 2)	0.712	0.692
Pureza global (PCA 4)	0.714	0.667
Calinski–Harabasz (NO PCA)	865274.6	**2131396.8**
Calinski–Harabasz (PCA 2)	133226.3	740376.9
Calinski–Harabasz (PCA 4)	72527.5	180458.7
Davies–Bouldin (NO PCA)	**0.358**	0.386
Davies–Bouldin (PCA 2)	0.490	0.440
Davies–Bouldin (PCA 4)	0.968	0.646

Aunque el flujo propuesto es eficaz en Elliptic++ [6], su validez externa es incierta al no probarse en otras cadenas (p. ej. Ethereum) ni en datasets con diferentes distribuciones. Además, depende de semillas lícitas e ilícitas cuyo sesgo puede afectar las puntuaciones (pl, pi), cuya sensibilidad a selección y tamaño debe evaluarse en futuros trabajos. La evaluación se limita a métricas internas (Silhouette, pureza, densidad), por lo que hace falta verificar su precisión con etiquetas de referencia o en entornos KYC/AML reales. Finalmente, el análisis usa un grafo estático y omite dimensión temporal y atributos financieros, que podrían mejorar la detección de conductas

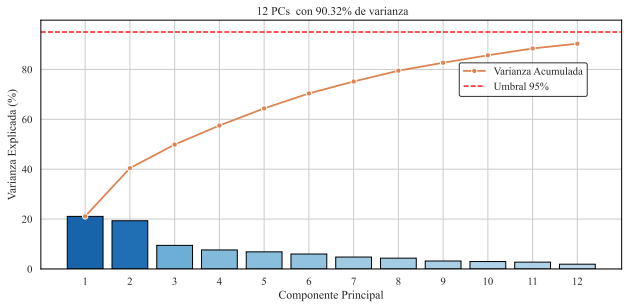

Figura 3. Varianza acumulada por número de componentes principales.

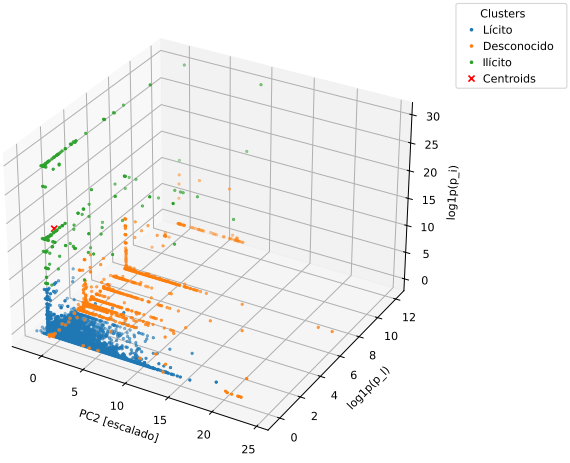

Figura 4. Varianza acumulada por número de PCs.

ilícitas en extensiones futuras.

IV. CONCLUSIONES

Los resultados muestran que las probabilidades p_l y p_i obtenidas del algoritmo PageRank constituyen características suficientes para la agrupación en clusters, alcanzando un Silhouette score de 0.955 sin aplicar PCA.

La aplicación de PCA deteriora consistentemente el rendimiento en las métricas evaluadas, sugiriendo que p_i y p_l, es decir, las dos características empleadas por PageRank capturan la información esencial para la clasificación. El algoritmo trimmed K–Means mejora la pureza global (0.715 vs 0.712) y la separación entre clústeres (índice Calinski-Harabasz), a costa de una ligera reducción en la cohesión interna. Si se quisiera aplicar PCA, sería recomendable modificar el algoritmo de K–Means a un algoritmo de weighted K–Means, que permite corregir las desviaciones para enfocarnos en lo esencial dependiendo de los intereses del análisis.

La metodología propuesta reduce significativamente el número de direcciones no clasificadas, proporcionando una herramienta práctica y simple, que puede funcionar como base para el análisis forense de redes blockchain. Sería interesante, junto a esta metodología, explorar otras técnicas de clustering más avanzadas, así como la incorporación de características temporales de las transacciones para mejorar la precisión de la clasificación.

AGRADECIMIENTOS

Este trabajo forma parte del proyecto Laboratorio de I+D+i en Ciberseguridad, Privacidad y Comunicaciones Seguras (TRUST Lab), financiado por la Unión Europea NextGeneration-EU, Plan de Recuperación, Transformación y Resiliencia, a través del INCIBE.

REFERENCIAS

[1] S. Nakamoto, "Bitcoin: A Peer-to-Peer Electronic Cash System," white paper, Oct. 31, 2008. [Online]. Available: https://bitcoin.org/bitcoin.pdf (Accessed May 26, 2025).

[2] European Securities and Markets Authority, Crypto Assets: Market Structures and EU Relevance, Risk Analysis Article ESMA50-524821-3153, Apr. 2024. [Online]. Available: https://www.esma.europa.eu/sites/default/files/2024-04/ESMA50-524821-3153_risk_article_crypto_assets_market_structures_and_eu_relevance.pdf (Accessed: May 26, 2025).

[3] J. Forestell, "Visa's role in stablecoins," Visa Perspectives, 30-Apr-2025. [Online]. Available: https://corporate.visa.com/en/sites/visa-perspectives/innovation/visas-role-in-stablecoins.html (Accessed: May 26, 2025).

[4] Financial Action Task Force (FATF), Targeted Update on Implementation of the FATF Standards on Virtual Assets and Virtual Asset Service Providers, Paris, France, Jun. 2023. [Online]. Available: https://www.fatf-gafi.org/en/publications/Fatfrecommendations/targeted-update-virtual-assets-vasps-2023.html (Accessed: May 26, 2025).

[5] M. Weber, G. Domeniconi, J. Chen, D. K. I. Weidele, C. Bellei, T. Robinson, and C. E. Leiserson, "Anti-money laundering in Bitcoin: Experimenting with graph convolutional networks for financial forensics," in Proc. KDD '19 Workshop on Anomaly Detection in Finance, Anchorage, AK, USA, Aug. 2019, pp. 1-7. Doi: 10.48550/arXiv.1908.02591

[6] Y. Elmougy and L. Liu, "Demystifying Fraudulent Transactions and Illicit Nodes in the Bitcoin Network for Financial Forensics," in Proc. 29th ACM SIGKDD Conf. Knowledge Discovery and Data Mining (KDD '23), New York, NY, USA: ACM, 2023, pp. 3979-3990. doi: 10.1145/3580305.3599803

[7] A. Asiri and K. Somasundaram, "Graph convolution network for fraud detection in bitcoin transactions," Scientific Reports, vol. 15, Art. no. 11076, 2025, doi: 10.1038/s41598-025-95672-w

[8] Nicholls J., Kuppa A., and Le-Khac N.-A., "FraudLens: Graph Structural Learning for Bitcoin Illicit Activity Identification," in Proc. 39th Annu. Computer Security Applications Conf. (ACSAC '23), Austin, TX, USA, 4–8 Dec. 2023, pp. 324-336. doi: 10.1145/3627106.3627200

[9] G. Vlahavas, K. Karasavvas, and A. Vakali, "Unsupervised clustering of bitcoin transactions," Financial Innovation, vol. 10, Art. no. 25, Jan. 2024, doi: 10.1186/s40854-023-00525-y

[10] L. Page, S. Brin, R. Motwani, and T. Winograd, "The PageRank citation ranking: Bringing order to the Web," Stanford InfoLab, Tech. Rep. 1999-66, Nov. 1999. [Online]. Available: http://ilpubs.stanford.edu:8090/422 (Accessed: May 26, 2025).

[11] H. Tong, C. Faloutsos, and J.-Y. Pan, "Random walk with restart: fast solutions and applications," Knowledge and Information Systems, vol. 14, no. 3, pp. 327–346, 2008, doi: 10.1007/s10115-007-0094-2

[12] Y. Elmougy and L. Liu, "Elliptic++ Dataset: A graph network of Bitcoin blockchain transactions and wallet addresses," GitHub repository, git-disl/EllipticPlusPlus, Apr. 2024. [Online]. Available: https://github.com/git-disl/EllipticPlusPlus (Accessed: May 26, 2025).

[13] C. Ding and X. He, "K-means Clustering via Principal Component Analysis," in Proc. 21st Int. Conf. Machine Learning (ICML 2004), Banff, Alberta, Canada, Jul. 2004, pp. 225–232. [Online]. Available: https://icml.cc/Conferences/2004/proceedings/papers/262.pdf