LUCoS: Selección Latente no Supervisada de Contexto para Modelos Fundacionales Tabulares

Oroel Ipas[1 [0000−0002−5963−779X]],
Guillermo Gomez-Trenado[1 [0000−0003−3366−6047]],
Rocío Romero-Zaliz[2,1,3,4 [0000−0003−4500−9781]], e
Isaac Triguero[2,1 [0000−0002−0150−0651]]

[1] Instituto Andaluz Interuniversitario en Ciencia de Datos e Inteligencia Computacional (DaSCI)
[2] Departamento de Ciencias de la Computación e Inteligencia Artificial (DECSAI)
[3] Centro de Investigación en Tecnologías de la Información y las Comunicaciones (CITIC)
[4] Instituto de Investigación Biosanitaria Ibs.GRANADA
Universidad de Granada, Granada, España
{oroel,guillermogomez,rocio}@ugr.es, triguero@decsai.ugr.es

Resumen. Decidir qué instancias etiquetar desempeña un papel clave en el aprendizaje tabular con pocas etiquetas. En los recientes modelos fundacionales tabulares (*Tabular Foundation Models*, TFMs), como TabPFN, que predicen mediante aprendizaje en contexto, el conjunto de soporte etiquetado influye directamente en su rendimiento. Sin embargo, esta selección ha sido poco estudiada en escenarios de arranque en frío, donde las instancias deben elegirse antes de conocer sus etiquetas. En visión por computador y procesamiento del lenguaje natural, la selección no supervisada suele apoyarse en espacios latentes generados por modelos fundacionales. En datos tabulares, en cambio, se realiza habitualmente en el espacio de características original, donde la heterogeneidad de tipos, escalas, ruido e interacciones no lineales limita la utilidad de esa geometría para seleccionar instancias. **LUCoS** (***L****atent* ***U****nsupervised* ***Co****ntext* ***S****election*) selecciona instancias en el espacio latente de una *Prior-Fitted Network* no supervisada. Estas instancias se anotan posteriormente y se usan como contexto de un TFM. De este modo, LUCoS aprovecha una geometría más informativa que la del espacio tabular original sin acceder a etiquetas ni a datos de test durante la selección. Evaluado sobre 67 conjuntos de datos de OpenML-CC18 y seis tamaños de subconjunto, LUCoS obtiene el mejor valor medio de AUC y la mejor posición media en el ranking en todos los tamaños considerados. Además, supera a la selección aleatoria y a la selección en el espacio de características original. Código disponible en `https://github.com/ari-dasci/S-LUCoS`.

Palabras clave: Aprendizaje tabular · Selección no supervisada · Modelos fundacionales tabulares · Aprendizaje en contexto · *Prior-Fitted Networks* · Aprendizaje con pocos ejemplos

1 Introducción

Los datos tabulares son ampliamente utilizados en dominios como salud, finanzas, energía e industria, pero sus etiquetas suelen ser costosas de obtener, ya que requieren expertos, accesos regulados o procesos de anotación largos [4,3]. Por ello, los escenarios con pocas etiquetas son habituales.

Ante esta situación, los modelos fundacionales tabulares (*Tabular Foundation Models*, TFMs) como TabPFN [13,14] y TabICL [17] resultan atractivos porque predicen mediante aprendizaje en contexto (*in-context learning*, ICL): reciben un pequeño conjunto etiquetado de soporte (contexto) y evitan modificar los pesos del modelo para cada tarea específica. Este comportamiento se conecta con el aprendizaje con pocos ejemplos (*few-shot*) de los grandes modelos de lenguaje [5] y con sus adaptaciones al dominio tabular, como en TabLLM [12].

Esta propiedad hace que la construcción del contexto sea clave, pues, cuando nos limitamos a conjuntos etiquetados de tamaño reducido, el rendimiento depende tanto de la calidad del predictor como de las instancias seleccionadas para su anotación. De hecho, los resultados empíricos muestran que TabPFN puede ser muy sensible a la composición del contexto. Esto plantea una pregunta práctica: **con un tamaño fijo del conjunto etiquetado y antes de conocer las etiquetas, ¿qué instancias conviene seleccionar para ser etiquetadas?**

La selección de instancias antes de disponer de etiquetas se conoce habitualmente como el problema de arranque en frío (*cold-start*) [16]. En otros dominios, este problema se ha abordado mediante criterios geométricos de cobertura, diversidad o densidad [21,19]. En visión por computador, estos criterios suelen aplicarse sobre representaciones latentes de modelos fundacionales, donde las distancias capturan similitud semántica. En datos tabulares, en cambio, la selección suele realizarse en el espacio de características original, que combina variables numéricas y categóricas, escalas heterogéneas, valores perdidos e interacciones no lineales [15,20]. Como resultado, las distancias pueden no reflejar similitud predictiva, limitando la eficacia de los selectores geométricos.

Motivado por esta carencia, este trabajo estudia la selección no supervisada de contexto para TFMs tabulares: seleccionar instancias reales de un conjunto no etiquetado para que posteriormente sean anotadas y usadas como contexto. Los métodos recientes de optimización de contexto para TFMs suelen asumir etiquetas para ajustar representaciones de contexto con información supervisada [7,22], por lo que no resuelven directamente este escenario de arranque en frío.

LUCoS (***L****atent* ***U****nsupervised* ***Co****ntext* ***S****election*) sustituye la geometría del espacio de características original por representaciones latentes de un TFM tabular no supervisado [23]. Sobre ese espacio latente, LUCoS aplica K-Medoids para seleccionar instancias reales, que posteriormente se etiquetan y usan como contexto de un TFM de clasificación supervisada. El método no utiliza etiquetas ni datos de test y se integra directamente en el proceso de inferencia de los TFMs.

Las contribuciones principales son:

i) Mostramos que el rendimiento de los TFMs con pocas etiquetas es muy sensible al contexto, lo que hace que su selección sea un componente clave del ICL tabular.
ii) Proponemos LUCoS, un método de selección no supervisada de contexto que hace uso del espacio latente de una TFM no supervisado para identificar subconjuntos informativos mediante métodos geométricos como K-Medoids.
iii) Mostramos que LUCoS obtiene la mejor posición media en el ranking para los seis diferentes tamaños de subconjunto etiquetado evaluados sobre 67 conjuntos de datos de OpenML-CC18, con mejoras consistentes frente a muestreo aleatorio y selección en el espacio de características original.

2 Trabajo relacionado

Selección no supervisada de instancias. La selección no supervisada de instancias busca identificar subconjuntos informativos a partir de datos no etiquetados. Métodos recientes como USL [21], RDSS [19] y ZCore [10] emplean criterios geométricos de cobertura, densidad o diversidad aplicados sobre espacios de representación inducidos por modelos fundacionales. La eficacia de estos métodos depende de que las distancias reflejen similitud relevante para la tarea. En datos tabulares, sin embargo, la selección suele realizarse directamente sobre el espacio de características original [9], donde la mezcla de tipos de datos, las escalas heterogéneas y las relaciones no lineales puede dificultar la existencia de una geometría significativa. Esto sugiere que la calidad de la representación es tan importante como el propio criterio de selección.

Modelos fundacionales tabulares y representaciones latentes. Las *Prior-Fitted Networks* (PFNs), entrenadas para aproximar inferencia bayesiana sobre distribuciones de tareas, han dado lugar a modelos tabulares eficaces con pocos datos etiquetados, como TabPFN [13,14]. Más allá del aprendizaje supervisado, las PFNs también resultan prometedoras para el aprendizaje de representaciones, especialmente difícil en datos tabulares por la ausencia de una estructura espacial o secuencial natural [15]. En esta línea, TabClustPFN [23] muestra que una PFN no supervisada puede inducir representaciones latentes capaces de capturar estructura de *clustering* sin necesidad de etiquetas. Estos resultados sugieren que las PFNs pueden proporcionar geometrías latentes significativas para datos tabulares. Sin embargo, el potencial de estas representaciones para tareas distintas del *clustering*, como la selección no supervisada de instancias, apenas se ha explorado.

Optimización de contexto para TFMs. Dado que el rendimiento y el coste computacional de los TFMs dependen de los datos de contexto, trabajos recientes han propuesto métodos para optimizarlo, como TuneTables [7] y TACO [22]. Sin embargo, estos enfoques asumen conjuntos de datos completamente etiquetados y optimizan representaciones de contexto en lugar de seleccionar instancias reales, por lo que no son aplicables en escenarios de arranque en frío.

3 Metodología

Objetivo. Los TFMs reciben un conjunto etiquetado como contexto en inferencia. En regímenes de pocas etiquetas, dos subconjuntos de soporte del mismo tamaño pueden producir rendimientos muy distintos. Se estudia la *selección no supervisada de contexto*: escoger, antes de conocer las etiquetas, un pequeño subconjunto de entrenamiento para anotarlo y usarlo como conjunto de contexto.

Requisitos. La selección no supervisada de contexto se diferencia del aprendizaje activo (*active learning*), la selección de *core-sets* basada en etiquetas y la destilación de conjuntos de datos. El método debe **(i)** no utilizar etiquetas, **(ii)** no acceder a los datos de test, **(iii)** basarse en instancias reales y **(iv)** usar una única consulta al oráculo.

3.1 LUCoS

Se propone **LUCoS** (***L****atent* ***U****nsupervised* ***Co****ntext* ***S****election*), un método de cuatro etapas que combina una proyección latente $\phi : \mathcal{X} \to \mathcal{Z}$ con un selector geométrico σ que devuelve K instancias en $\mathcal{Z}$. La hipótesis central es que el espacio tabular original ofrece una geometría poco fiable para la selección no supervisada, mientras que las representaciones latentes de PFNs no supervisadas proporcionan un espacio más adecuado.

La Figura 1 resume el flujo: las instancias de entrenamiento no etiquetadas se proyectan a un espacio latente inducido por una PFN no supervisada, se seleccionan muestras representativas mediante un método geométrico y se etiquetan con una sola consulta al oráculo. Por último, el subconjunto etiquetado, expresado en el espacio de características original, se emplea como conjunto de contexto para un TFM supervisado. LUCoS utiliza la representación latente únicamente para la selección. La inferencia posterior se realiza en el espacio de características original. Esta separación permite aislar el efecto de la estrategia de selección manteniendo fijo el modelo clasificador.

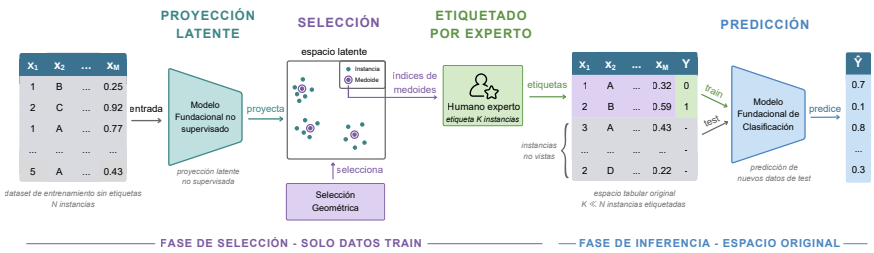


Fig. 1. Esquema general de LUCoS. El método consiste en cuatro etapas: **(i)** Las instancias no etiquetadas se proyectan a un espacio latente de forma no supervisada, **(ii)** se realiza una selección geométrica en ese espacio, **(iii)** se etiquetan las instancias seleccionadas en una única llamada y **(iv)** se usan como contexto de un modelo fundacional tabular supervisado. La selección no usa etiquetas ni datos de test.

Elección de componentes. Para ϕ se utiliza el *PIN encoder* de TabClustPFN [23], que produce una representación latente de 512 dimensiones por instancia. Para σ se emplea K-Medoids con distancia euclídea, donde los medoides corresponden a instancias reales seleccionadas del conjunto de datos. El número de medoides se fija en K, coincidiendo con el tamaño del conjunto a etiquetar.

Definición formal. Dado un conjunto de entrenamiento $\mathcal{D} = \{x_i\}_{i=1}^{N}$, cada instancia se proyecta primero al espacio latente:

$$z_i = \phi(x_i), \qquad z_i \in \mathbb{R}^{512}. \tag{1}$$

A continuación se aplica K-Medoids sobre $\{z_i\}_{i=1}^{N}$ usando distancia euclídea, con el número de medoides fijado al tamaño del conjunto etiquetado K. El conjunto de índices seleccionado viene dado por

$$\mathcal{I}_K = \underset{\mathcal{I}\subseteq\{1,\ldots,N\},\,|\mathcal{I}|=K}{\operatorname{argmin}} \sum_{i=1}^{N} \min_{j\in\mathcal{I}} \|z_i - z_j\|_2, \tag{2}$$

lo que produce el subconjunto de contexto $\mathcal{Q}_K = \{x_i \mid i \in \mathcal{I}_K\}$. Como los medoides corresponden a instancias reales de entrenamiento, cada elemento de $\mathcal{Q}_K$ se puede rastrear hasta el espacio de características original y enviarse al oráculo para su anotación. Una vez etiquetado, $\mathcal{Q}_K$ se usa como conjunto de contexto por el TFM supervisado encargado de la predicción. Todo el proceso de representación latente, selección y construcción del subconjunto se realiza sin acceso a las instancias de test.

3.2 Benchmark y método de evaluación

Se utiliza TabPFN-2.5 [11] como clasificador para la predicción. La evaluación se realiza sobre OpenML-CC18 [2], restringido a conjuntos de datos con un máximo de diez clases, limitado por el modelo TabPFN-2.5. Esto produce 67 conjuntos de datos de clasificación con variables numéricas y categóricas, valores perdidos y tamaños diversos. Los valores numéricos faltantes se imputan con la media de entrenamiento y los categóricos con la moda, siguiendo prácticas estándar [1].

Sea C el número de clases. Se consideran los tamaños del conjunto etiquetado:

$$K \in \{i \cdot C \mid i \in \{1, 2, 4, 8, 16, 32\}\},$$

de modo que K escala exponencialmente con el número de clases. A lo largo del artículo, iC denota un tamaño total de $i \times C$ instancias etiquetadas, sin que eso signifique un número fijo de ejemplos por clase.

Para la evaluación se utilizan las 10 particiones oficiales proporcionadas por OpenML-CC18. Los resultados se promedian sobre las 10 particiones para obtener un único valor por conjunto de datos. En los métodos estocásticos (como la selección aleatoria), cada partición se ejecuta con 5 semillas aleatorias y los resultados se promedian entre semillas antes de agregarse sobre las particiones.

La métrica principal es el área bajo la curva ROC (*Area Under the ROC Curve*, AUC) con estrategia *one-vs-rest* para tareas multiclase, como también se hace en [14]. Se informa también de la posición media en el ranking de cada conjunto de datos para resumir el rendimiento relativo de cada método (R_{AUC}).

3.3 Métodos de referencia

LUCoS se compara con métodos de selección no supervisada compatibles con el mismo escenario de arranque en frío. Para cada método basado en geometría, se evalúa tanto el espacio de características original como el espacio de representaciones latentes del *PIN encoder* de TabClustPFN.

- **Random:** Muestreo aleatorio uniforme a partir del conjunto no etiquetado. Independiente de la geometría.
- **K-Medoids:** K-Medoids euclídeo, con K fijado al tamaño del conjunto a etiquetar.
- **ZCore:** Selección no supervisada que favorece muestras representativas y no redundantes mediante subespacios de baja dimensión [10].
- **RDSS:** Selección de subconjuntos representativos y diversos mediante aproximación de la distribución global [19].

4 Experimentos

Dos preguntas guían esta sección.

1) ¿Cuánto afecta el conjunto de las instancias de contexto a TabPFN? La Sección 4.1 lo comprueba con un oráculo supervisado basado en un algoritmo genético, mostrando la importancia de la selección de contexto en regímenes con pocas etiquetas.

2) ¿Puede recuperarse parte de ese margen de rendimiento sin etiquetas? La comparación principal muestra que LUCoS (K-Medoids sobre representaciones latentes del *PIN encoder* de TabClustPFN) obtiene el mejor rendimiento global en todos los tamaños del conjunto etiquetado evaluados.

4.1 Sensibilidad de TabPFN al contexto

Dado el buen rendimiento de TabPFN con pocos datos, primero se evalúa la sensibilidad de su rendimiento a la elección del contexto bajo un régimen de pocas etiquetas: ¿importa la elección del contexto o cualquier subconjunto razonable produce un comportamiento similar? Para comparar tareas con distintos niveles de dificultad, se usa la métrica **Improvability** [18]. Esta métrica permite cuantificar la pérdida de rendimiento asociada al uso de pocas etiquetas, donde valores cercanos a cero indican un comportamiento próximo al obtenido con el conjunto completo etiquetado.

La Figura 2 muestra que TabPFN mantiene un rendimiento competitivo con contextos pequeños, superando a clasificadores clásicos como XGBoost [6] y KNN

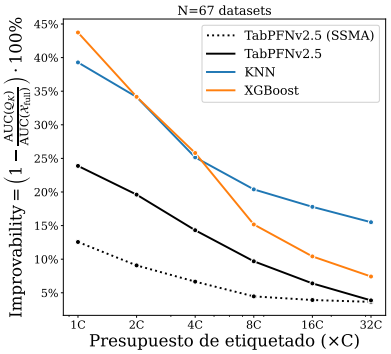


Fig. 2. Rendimiento de los modelos frente al número de instancias seleccionadas. Improvability mide la brecha porcentual de AUC respecto a TabPFN con todo el entrenamiento etiquetado. Las líneas continuas usan selección aleatoria balanceada; la línea punteada muestra TabPFN con el mejor subconjunto encontrado por SSMA.

bajo selección aleatoria balanceada. Sin embargo, el resultado también evidencia que el tamaño del contexto no es el único factor relevante: distintos subconjuntos del mismo tamaño pueden producir rendimientos muy diferentes. Esta sensibilidad es especialmente marcada con los conjuntos etiquetados más pequeños, donde cada instancia tiene un impacto considerable en la capacidad predictiva.

Para cuantificar este margen de mejora, se compara la selección aleatoria balanceada con un procedimiento supervisado de tipo oráculo basado en el *Steady-State Memetic Algorithm* (SSMA) [8], un método evolutivo utilizado en selección de instancias [9]. SSMA optimiza la búsqueda del subconjunto que maximiza el AUC de validación mientras reduce su tamaño. Aunque requiere etiquetas y retroalimentación supervisada, inaccesibles en el escenario objetivo, su utilidad aquí no es práctica sino analítica: se emplea como una cota superior del beneficio potencial asociado a una mejor selección del contexto.

Los subconjuntos encontrados por SSMA mejoran sustancialmente la selección aleatoria, con una brecha de aproximadamente 0.19 ΔAUC en 1C. Esta brecha se reduce conforme aumentan el tamaño del conjunto etiquetado y la cobertura de la distribución. La conclusión es doble: TabPFN es el mejor modelo en regímenes de pocas etiquetas, pero también es muy sensible a la composición del soporte. Esto motiva directamente la siguiente sección: si existe un margen real asociado a la elección del contexto, la cuestión clave es cuánto puede recuperarse mediante selección no supervisada.

4.2 Selección de contexto sin etiquetas

En este experimento, evaluamos LUCoS frente a varios métodos de referencia: selección aleatoria, K-Medoids en el espacio de características original, RDSS y ZCore. RDSS y ZCore se aplican tanto en el espacio de características original como en el espacio latente. La Tabla 1 presenta el AUC medio y la posición media

Tabla 1. AUC medio y posición media en el ranking, R_{AUC}, según el tamaño del conjunto etiquetado. LUCoS corresponde a K-Medoids sobre representaciones latentes del *PIN encoder* de TabClustPFN. En negrita se indica el mejor valor por columna.

Espacio	Método	1C		2C		4C		8C		16C		32C	
		AUC↑	R_{AUC}↓	AUC	R_{AUC}	AUC	R_{AUC}	AUC	R_{AUC}	AUC	R_{AUC}	AUC	R_{AUC}
–	*Random Unsupervised*	0.61	*3.45*	0.68	*3.36*	0.74	*3.34*	**0.80**	*3.18*	0.83	*3.24*	**0.86**	*3.34*
Original	RDSS	0.60	4.26	0.65	4.15	0.70	4.10	0.75	4.07	0.79	4.40	0.83	4.38
	ZCore	0.54	5.49	0.59	5.70	0.64	5.68	0.71	5.34	0.78	5.28	0.82	5.05
	K-Medoids	**0.65**	*2.77*	0.69	*3.13*	0.74	*3.13*	0.78	*3.48*	0.82	*3.43*	0.85	*3.37*
TabClustPFN PIN encoder	RDSS	0.58	4.76	0.62	4.52	0.68	4.93	0.73	4.98	0.80	4.76	0.84	4.75
	ZCore	0.57	4.67	0.63	4.62	0.71	4.28	0.77	4.33	0.82	4.13	0.85	*4.01*
	K-Medoids (LUCoS)	**0.65**	**2.60**	**0.71**	**2.52**	**0.76**	**2.54**	**0.80**	**2.63**	**0.84**	**2.75**	**0.86**	**3.09**

en el ranking de AUC. LUCoS obtiene el mejor rendimiento global en todos los tamaños de selección evaluados. Además, supera de forma consistente a K-Medoids en el espacio de características original, lo que sugiere que la geometría latente mejora la selección de contexto. Las posiciones en el ranking obtenidas con AUC se correlacionan fuertemente con *Accuracy* (ACC) y F1 (Spearman $\rho = 0.87$ y $\rho = 0.95$), por lo que solo se presenta AUC.

La comparación más informativa es la de LUCoS frente a K-Medoids en el espacio de características original, ya que el selector se mantiene fijo y solo cambia la representación. La Figura 3 muestra una comparación pareada por conjunto de datos para cada tamaño del conjunto etiquetado. La ventaja de LUCoS es reducida con los conjuntos más pequeños y aumenta en los tamaños interme-

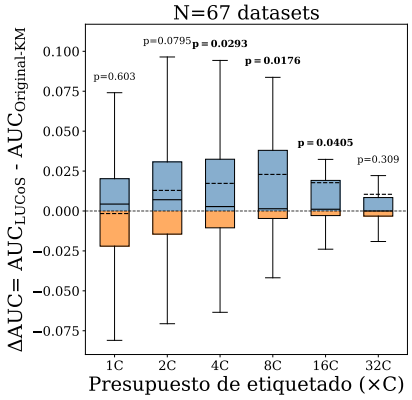


Fig. 3. Diferencia de AUC en el conjunto de prueba (ΔAUC = AUC_{LUCoS} – AUC obtenida mediante K-Medoids en el espacio original de características) pareada por conjunto de datos, en función del presupuesto de etiquetado ($\times C$). La parte azul de cada diagrama de cajas representa diferencias positivas, indicando un mejor rendimiento de LUCoS, mientras que la parte naranja representa diferencias negativas. Sobre cada diagrama se muestra el p-valor del test pareado de rangos con signo de Wilcoxon entre ambos métodos. Los p-valores significativos ($p < 0.05$) se muestran en **negrita**.

dios. Cuando el contexto crece, esta ventaja vuelve a disminuir, reflejando una convergencia general de los distintos métodos. Esto sugiere que la representación latente resulta especialmente útil cuando el conjunto etiquetado ya cubre varias regiones de la distribución, pero el espacio de características original no proporciona una geometría suficientemente fiable.

El patrón principal es claro: K-Medoids mejora cuando se aplica sobre representaciones latentes de TabClustPFN en lugar de sobre características originales, mientras que aplicar el mismo espacio latente a RDSS o ZCore no basta para superar a LUCoS. El resultado sugiere que la combinación de una representación latente adecuada y un selector de cobertura basado en instancias reales es especialmente efectiva para construir contextos pequeños.

5 Conclusiones, limitaciones y trabajo futuro

Este trabajo ha estudiado la selección no supervisada de contexto para TFMs tabulares en escenarios de arranque en frío. Los resultados confirman que el contexto no solo condiciona el rendimiento, sino que puede optimizarse incluso antes de disponer de etiquetas.

LUCoS demuestra que una representación latente adecuada mejora la selección no supervisada de contexto con K-Medoids y obtiene el mejor rendimiento global en los seis tamaños de selección sobre 67 conjuntos de datos de OpenML-CC18. Más allá de los resultados empíricos, el análisis revela que la calidad de la representación es un factor determinante para la selección no supervisada de instancias tabulares. Mientras que los métodos geométricos aplicados directamente sobre el espacio de características original muestran un comportamiento inconsistente, una geometría latente más informativa permite que incluso selectores sencillos identifiquen subconjuntos de alto rendimiento. Así, los resultados sitúan la selección no supervisada de contexto como un puente natural entre los abundantes datos no etiquetados y el potencial de los TFMs para aprender con muy pocas etiquetas.

Limitaciones y trabajo futuro. El estudio se limita a tareas de clasificación con un máximo de diez clases, condicionado por las restricciones de TabPFN-2.5, y no aborda problemas de regresión. Además, K-Medoids presenta limitaciones de escalabilidad en conjuntos de datos de gran tamaño. Su elevado coste computacional y su requerimiento de memoria de orden $\mathcal{O}(N^2)$ para almacenar la matriz de distancias motivan el desarrollo de aproximaciones más escalables. Quedan abiertas varias cuestiones relevantes, como el uso de otras representaciones latentes tabulares, selectores geométricos alternativos o mecanismos que mejoren la cobertura de regiones minoritarias sin acceder a etiquetas. Como trabajo futuro, también queda por estudiar si las representaciones pueden optimizarse explícitamente para la selección de instancias, en lugar de derivarse de objetivos relacionados con el *clustering*.

Agradecimientos. Esta investigación es parte del Proyecto TSI-100927-2023-1, financiado por Plan de Recuperación, Transformación y Resiliencia Europeo, de la Unión Europea-Next Generation EU a través del Ministerio para la Transformación Digital y de la Función Pública y está parcialmente financiada por el Ministerio de Ciencia y Tecnología de España en el marco del proyecto PID2023-149128NB-I00, financiado por MICIU/AEI/10.13039/501100011033 y por FEDER, UE.

R. Romero-Zaliz también está financiada por la Consejería de Universidad, Investigación e Innovación de la Junta de Andalucía (DGP_PIDI_2024_00563).

Declaración de intereses. Los autores declaran que no tienen intereses en conflicto relevantes para el contenido de este artículo.

Referencias

1. Bahri, D., Jiang, H., Tay, Y., Metzler, D.: SCARF: Self-Supervised Contrastive Learning using Random Feature Corruption. In: International Conference on Learning Representations (ICLR) (2022)
2. Bischl, B., Casalicchio, G., Feurer, M., Gijsbers, P., Hutter, F., Lang, M., Mantovani, R.G., van Rijn, J.N., Vanschoren, J.: OpenML Benchmarking Suites. In: Advances in Neural Information Processing Systems (NeurIPS) Datasets and Benchmarks Track (2021)
3. Borisov, V., Leemann, T., Seßler, K., Haug, J., Pawelczyk, M., Kasneci, G.: Deep Neural Networks and Tabular Data: A Survey. IEEE Transactions on Neural Networks and Learning Systems **35**(6), 7499–7519 (2022)
4. van Breugel, B., van der Schaar, M.: Position: Why Tabular Foundation Models Should Be a Research Priority. In: Proceedings of the 41st International Conference on Machine Learning (ICML) (2024)
5. Brown, T., Mann, B., Ryder, N., Subbiah, M., Kaplan, J.D., Dhariwal, P., Neelakantan, A., Shyam, P., Sastry, G., Askell, A., et al.: Language Models are Few-Shot Learners. In: Advances in Neural Information Processing Systems (NeurIPS). vol. 33, pp. 1877–1901 (2020)
6. Chen, T., Guestrin, C.: XGBoost: A Scalable Tree Boosting System. In: Proceedings of the 22nd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining (KDD). pp. 785–794 (2016)
7. Feuer, B., Schirrmeister, R.T., Cherepanova, V., Hegde, C., Hutter, F., Goldblum, M., Cohen, N., White, C.: TuneTables: Context Optimization for Scalable Prior-data Fitted Networks. Advances in Neural Information Processing Systems (NeurIPS) **37**, 83430–83464 (2024)
8. García, S., Cano, J.R., Herrera, F.: A Memetic Algorithm for Evolutionary Prototype Selection: A Scaling Up Approach. Pattern Recognition **41**(8), 2693–2709 (2008)
9. García, S., Derrac, J., Cano, J., Herrera, F.: Prototype selection for nearest neighbor classification: Taxonomy and empirical study. IEEE Transactions on Pattern Analysis and Machine Intelligence (TPAMI) **34**(3), 417–435 (2012)
10. Griffin, B.A., Marks, J., Corso, J.J.: Zero-Shot Coreset Selection via Iterative Subspace Sampling. In: Proceedings of the IEEE/CVF Winter Conference on Applications of Computer Vision (WACV). pp. 2114–2124 (2026)
11. Grinsztajn, L., Flöge, K., Key, O., Birkel, F., Jund, P., Roof, B., Jäger, B., Safaric, D., Alessi, S., Hayler, A., et al.: TabPFN-2.5: Advancing the State of the Art in Tabular Foundation Models. arXiv preprint arXiv:2511.08667 (2025)

12. Hegselmann, S., Buendia, A., Lang, H., Agrawal, M., Jiang, X., Sontag, D.: TabLLM: Few-shot classification of tabular data with large language models. In: International Conference on Artificial Intelligence and Statistics (AISTATS). pp. 5549–5581. PMLR (2023)
13. Hollmann, N., Müller, S., Eggensperger, K., Hutter, F.: TabPFN: A Transformer That Solves Small Tabular Classification Problems in a Second. In: International Conference on Learning Representations (ICLR) (2023)
14. Hollmann, N., Müller, S., Purucker, L., Krishnakumar, A., Körfer, M., Hoo, S.B., Schirrmeister, R.T., Hutter, F.: Accurate Predictions on Small Data with a Tabular Foundation Model. Nature **637**(8045), 319–326 (2025)
15. Jiang, J.P., Liu, S.Y., Cai, H.R., Zhou, Q.L., Ye, H.J.: Representation Learning for Tabular Data: A Comprehensive Survey. IEEE Transactions on Pattern Analysis and Machine Intelligence (TPAMI) **48**(6), 6488–6508 (2026). https://doi.org/10.1109/TPAMI.2026.3657217
16. Jin, Q., Yuan, M., Li, S., Wang, H., Wang, M., Song, Z.: Cold-start active learning for image classification. Information Sciences **616**, 16–36 (2022)
17. Qu, J., Holzmüller, D., Varoquaux, G., Le Morvan, M.: TabICL: A Tabular Foundation Model for In-Context Learning on Large Data. In: Proceedings of the 42nd International Conference on Machine Learning (ICML) (2025)
18. Qu, J., Holzmüller, D., Varoquaux, G., Le Morvan, M.: TabICLv2: A better, faster, scalable, and open Tabular Foundation Model. arXiv preprint arXiv:2602.11139 (2026)
19. Shao, Q., Kang, J., Chen, Q., Li, Z., Xu, H., Cao, Y., Liang, J., Wu, J.: Enhancing Semi-Supervised Learning via Representative and Diverse Sample Selection. Advances in Neural Information Processing Systems (NeurIPS) **37**, 111199–111226 (2024)
20. Wang, W.Y., Du, W.W., Xu, D., Wang, W., Peng, W.C.: A Survey on Self-supervised Learning for Non-sequential Tabular Data. Machine Learning **114**(1), 16 (2025)
21. Wang, X., Lian, L., Yu, S.X.: Unsupervised Selective Labeling for More Effective Semi-supervised Learning. In: Proceedings of the European Conference on Computer Vision (ECCV). pp. 427–445. Springer (2022)
22. Zabërgja, G., Kamel, R., Kadra, A., Frey, C.M.M., Grabocka, J.: End-to-End Compression for Tabular Foundation Models. arXiv preprint arXiv:2602.05649 (2026)
23. Zhao, T., Wang, G., Tan, Y.S., Zhang, Q.: TabClustPFN: A Prior-Fitted Network for Tabular Data Clustering. arXiv preprint arXiv:2601.21656 (2026)