Desaprendizaje Automático Distribucional para el Olvido Incremental de Clases

Nicolás Segura Kunsagi[1][0009−0002−1413−2277], Juan Luis Suárez Díaz[1][0000−0001−8587−4345], Iván Sevillano García[1][0000−0002−5029−9106] y Francisco Herrera[1,2][0000−0002−7283−312X]

[1] Instituto Andaluz Interuniversitario en Ciencia de Datos e Inteligencia Computacional (DaSCI)
[2] Departamento de Ciencias de la Computación e Inteligencia Artificial (DECSAI) Universidad de Granada, Granada, España
{nkunsagi,jlsuarezdiaz,isevillano}@ugr.es herrera@decsai.ugr.es

Resumen El desaprendizaje automático (DA) tiene como objetivo eliminar la influencia de datos específicos en los modelos sin recurrir al costoso entrenamiento desde cero. En redes profundas, los métodos *retain-free* con intervención son de especial interés, ya que prescinden del conjunto a retener, apoyándose solo en información extraída previamente. Sin embargo, su uso incremental acumula errores, pudiendo llegar a provocar olvido catastrófico. Este trabajo propone el método Distribucional, un enfoque *retain-free* diseñado para ser resiliente ante olvidos secuenciales. Inspirado en el aprendizaje continuo, genera datos sintéticos a nivel de *embeddings* muestreando la distribución paramétrica que mejor se ajusta a cada clase. En lugar de la suposición gaussiana estándar, selecciona entre distribuciones gaussianas, de Fisher-Bingham, t-Student y laplacianas mediante tests estadísticos y el criterio BIC, garantizando una representación más fiel. Experimentos con ViT-B/16 en CIFAR-10, CIFAR-100 y MNIST demuestran que el método propuesto es el único consistente en todo el olvido secuencial, manteniendo un rendimiento competitivo en retención y reduciendo con eficacia la precisión en las clases olvidadas, destacando su robustez en CIFAR-100 frente al colapso del estado del arte.

Keywords: Desaprendizaje Automático · Olvido Incremental de Clases · Datos Sintéticos

1. Introducción

Las redes neuronales profundas modernas son entrenadas con, o tienen acceso a, volúmenes masivos de datos. Dichos datos pueden ser recopilados a partir de usuarios o mediante el rastreo indiscriminado y sin filtrado de la web, lo que puede dar lugar a diversas problemáticas. El usuario podría ejercer su derecho al olvido [20] para que esos datos no fuesen usados para entrenar. En este contexto, las técnicas orientadas a eliminar del modelo el efecto de subconjuntos específicos

de los datos de entrenamiento, denominadas desaprendizaje automático (DA) [4], adquieren una importancia fundamental. Con el fin de evitar el elevado coste computacional y ecológico que conlleva el reentrenamiento completo del modelo [3], resulta imprescindible disponer de métodos de DA eficientes.

Existen métodos de DA eficaces cuando el modelo es entrenado con la previsión de que se realizará un proceso de olvido posterior [3]; sin embargo, estos métodos no son aplicables a modelos entrenados de manera convencional. Ante este escenario, surgieron enfoques post-hoc capaces de aplicarse de manera directa sobre cualquier modelo ya entrenado. En el marco del DA, el conjunto de entrenamiento se divide en dos subconjuntos: el conjunto de olvido, cuyos efectos se desean eliminar, y el conjunto de retención, cuyo conocimiento se desea preservar. A la hora de aplicar una técnica de DA, se emplean habitualmente dos funciones de pérdida, una asociada a cada subconjunto. Asimismo, cabe distinguir dos categorías principales: los métodos *retain-free*, que no requieren acceso al conjunto de retención durante el proceso de olvido, y aquellos que sí contemplan un acceso parcial o total a dicho conjunto [2]. Dentro de los métodos *retain-free* es posible establecer una distinción adicional en función de si requieren intervención o no, entendiéndose por intervención la necesidad de emplear características del modelo extraídas a partir del conjunto de datos de entrenamiento original [23].

Tras la aplicación de un proceso de olvido, especialmente en el caso de los métodos *retain-free*, el modelo experimenta una degradación en su rendimiento sobre el conjunto de retención. En consecuencia, la realización de múltiples olvidos sucesivos incrementa la probabilidad de que se produzca un olvido catastrófico, lo que pone de manifiesto la necesidad de métodos que no solo sean eficaces en un único proceso de olvido, sino que mantengan su efectividad a lo largo de sucesivas iteraciones [22].

En el ámbito del aprendizaje continuo [21], se ha demostrado que el uso de datos procedentes de tareas anteriores resulta especialmente beneficioso para el mantenimiento del conocimiento previo [19]. Dado que no siempre se dispone de acceso a dichos datos, una alternativa consiste en el empleo de datos sintéticos generados a partir de prototipos de clase. El método *Prototype Augmentation and Self-Supervision* (PASS) [24] sigue esta filosofía: al incorporar una nueva clase, se calcula su centroide en el espacio de representaciones, y en tareas sucesivas se generan datos sintéticos de dicha clase mediante muestreo de una distribución gaussiana con media igual al centroide y matriz de covarianza igual a la identidad. Tomando como inspiración el método PASS, en el presente trabajo se propone un método de DA *retain-free* con intervención, diseñado para ser resiliente ante un gran número de olvidos secuenciales. La propuesta es el método Distribucional, el cual no solo adapta PASS, sino que lo extiende considerando distintos tipos de distribuciones paramétricas y seleccionando, para cada clase, aquella que mejor se ajusta a sus datos.

Se ha evaluado la propuesta sobre una arquitectura basada en transformers y en diversos conjuntos de datos. Los resultados obtenidos muestran que el ren-

dimiento del método propuesto es superior a otros métodos del estado del arte en el escenario de olvido secuencial de clases.

Este trabajo se estructura de la siguiente manera. En la sección 2 se indican trabajos relacionados. El método propuesto se describe en la sección 3. En la sección 4 se detalla la experimentación realizada y se analizan los resultados obtenidos. Finalmente, en la sección 5 se exponen las conclusiones del trabajo y se apuntan posibles líneas de investigación futuras.

2. Trabajos relacionados

Desaprendizaje automático. El DA busca desarrollar métodos para eliminar la influencia de un conjunto de datos sobre un modelo ya entrenado en ellos. Los métodos post hoc, aplicables sobre cualquier modelo ya entrenado, han proliferado en los últimos años bajo estrategias diversas. Los basados en manipulación de fronteras de decisión modifican los *logits* del clasificador para expulsar las muestras del conjunto de olvido de su clase original [5]. Los enfoques con intervención recogen estadísticas del modelo o los datos en una fase previa al olvido y las emplean como guía durante el mismo, prescindiendo así del acceso a los datos originales en ese momento [23]. Finalmente, algunos métodos prescinden de una función de pérdida explícita y modifican directamente las representaciones del modelo [10].

Aprendizaje continuo en desaprendizaje automático. El DA de clases se puede enmarcar dentro del aprendizaje continuo [21]. Si se olvidan k clases de forma secuencial, las tareas son Tarea 0, Tarea 1, ..., Tarea k, donde la tarea 0 es aprender originalmente con todos los datos, la tarea 1 es olvidar una clase y recordar el resto, y la tarea 2 es olvidar otra clase y recordar el resto excluyendo las ya olvidadas. Las tareas hasta la k son similares. Siguiendo esta línea, existen varios trabajos que aplican técnicas de aprendizaje continuo para olvidar una clase [8]. El trabajo de Krawczyk et al. [13] expone un escenario donde se intercalan tareas de aprendizaje de nuevas clases y el olvido de clases ya aprendidas. En esta misma dirección, se toma PASS [24], una técnica de aprendizaje incremental de clases, y se aplica al olvido incremental de clases.

Distribución en el espacio latente. PASS supone que los vectores de características o *embeddings* de una clase siguen una distribución normal, pero esto no es siempre cierto. De hecho, en modelos de procesamiento del lenguaje se sabe que los *embeddings* de conceptos relacionados no siguen una distribución normal [17], de hecho se agrupan en direcciones.

Selección de Distribuciones. Dado que los *embeddings* pueden seguir distribuciones no gaussianas hay que ampliar el tipo de distribuciones a considerar. Por un lado se pueden considerar más distribuciones que propiamente pertenecen al espacio euclídeo y las cuales están discutidas en [12]. Ahora bien, los *embeddings* pueden agruparse en direcciones, por tanto es necesario conocer distribuciones que vivan propiamente en direcciones las cuales surgen en la estadística direccional [18]. Una vez tomadas las distintas posibles distribuciones, hay que seleccionar cuál de ellas se ajusta mejor a los datos. En esta línea el trabajo

de Claeskens et al. [7] compara distintos criterios de selección. De los distintos criterios destacamos el *criterio de información bayesiano* (*bayesian information criteria*, BIC) que pretende mitigar el efecto de aumentar número de parámetros y el sobreajuste consiguiente.

3. Propuesta Distribucional

Esta sección describe el método de DA propuesto, denominado Distribucional. La sección se dividirá en tres subsecciones: la primera donde se introducirá la notación necesaria; la segunda que explica la primera parte de la propuesta, el ajuste de la mejor distribución a cada clase; y la última que explica cómo se consigue el olvido.

3.1. Preliminares

Denominaremos $\mathcal{D}$ al conjunto de datos, formado por pares $(\boldsymbol{x}, y)$, donde $\boldsymbol{x}$ es una entrada asociada a la etiqueta $y \in \{1, \ldots, C\}$, siendo C el número de clases. Como $\mathcal{D}$ es finito, es posible decir que tiene N elementos, por lo que se pueden enumerar sus elementos, $(\boldsymbol{x}_i, y_i)$, y, cada clase tendrá N_c integrantes. En DA se divide el *dataset* $\mathcal{D}$ en dos partes no solapadas: el *conjunto de retención*, $\mathcal{D}_r$, y el *conjunto de olvido*, $\mathcal{D}_f$.

Al modelo de clasificación entrenado en el conjunto de entrenamiento de $\mathcal{D}$ se notará por f_θ. Se puede separar f_θ en dos, el extractor de caracteristicas, h_{θ_2}, y el clasificador, g_{θ_1}, verificando $f_\theta = g_{\theta_1} \circ h_{\theta_2}$.

Dado un $\boldsymbol{x}_i$ se denominará como *embedding* a su imagen por h_{θ_2}, $h_{\theta_2}(\boldsymbol{x}_i) = \boldsymbol{e}_i$. Al conjunto de *embeddings* de la clase c se denotará $E_c = \{\boldsymbol{e}_i : (\boldsymbol{x}_i, y_i) \in \mathcal{D},\ \boldsymbol{e}_i = h_{\theta_2}(\boldsymbol{x}_i),\ y_i = c\}$. Además, si dichos datos se ajustan a una distribución paramétrica P_c con parámetros Λ_c, se indicará por $E_c \sim P_c(\Lambda_c)$.

3.2. Ajuste de las distribuciones

Distribucional busca simular datos sintéticos que sean representativos de las clases a recordar. Para ello se usan distribuciones paramétricas ajustadas a los *embeddings* de cada clase. Dado que no siempre una gaussiana es la distribución que mejor se ajusta a un conjunto de datos, se proponencomo candidatas a las siguientes familias de distribuciones paramétricas:

- Distribuciones gaussianas, $\mathcal{N}(\boldsymbol{\mu}, \boldsymbol{\Sigma})$ [12].
- Distribuciones de Fisher-Bingham, $\mathcal{FB}(\hat{\boldsymbol{\mu}}, \boldsymbol{A})$ [18].
- Distribuciones de von Mises-Fisher $v\mathcal{MF}(\hat{\boldsymbol{\mu}}, \kappa)$ [18].
- Distribuciones t-student, $t(v, \boldsymbol{\xi}, \boldsymbol{\Psi})$ [12].
- Distribuciones laplacianas, $\mathcal{L}(\tilde{\boldsymbol{\mu}}, \boldsymbol{B})$ [12].

Dada una clase, dentro de cada familia, la mejor distribución que se adapta a E_c es aquella por la que sus parámetros han sido estimados por máxima

verosimilitud. A dichos parámetros se les denotará por el subíndice c. También, en el caso de que algunas de las matrices $\boldsymbol{\Sigma}_c$, $\boldsymbol{A}_c$, $\boldsymbol{\Psi}_c$ o $\boldsymbol{B}_c$ sean singulares tras estimarlas, se aplica la regularización de Ledoit-Wolf [16] para obtener matrices invertibles.

Una vez estimados los parámetros de cada distribución, se selecciona la más conveniente de todas. Para ese objetivo, se sigue un flujo de trabajo en el que primero se comprueba si los datos siguen una distribución concreta mediante un test estadístico para evitar seleccionar una distribución sobreajustada; en caso de rechazar los tests anteriores, se toma la distribución que mejor se ajusta a los datos.

El primer paso del flujo es comprobar si los datos siguen una distribución gaussiana. Al vivir los *embeddings* en un espacio de alta dimensionalidad se aplica un test que comprueba si es posible que los datos sigan una normalidad multivariante. El test de Henze-Zirkler [11] sirve para este propósito. Se aplica el test de Henze-Zirkler con un $\alpha = 0{,}05$ a E_c, y en caso de no rechazar normalidad se puede suponer $E_c \sim \mathcal{N}(\boldsymbol{\mu}_c, \boldsymbol{\Sigma}_c)$. En caso contrario se pasa al segundo paso.

A continuación se comprueba si E_c sigue una distribución $\mathcal{FB}$ realizando las siguientes comprobaciones:

1. Se comprueba si E_c se distribuye aproximadamente en una esfera, para ello se calcula el coeficiente de variación de las normas de los elementos de E_c, es decir

$$CV = \frac{\sigma}{\mu}, \quad \mu = \frac{1}{N_c}\sum_{i=1}^{N_c} ||\boldsymbol{e}_i^c||\,, \quad \sigma^2 = \frac{1}{N_c}\sum_{i=1}^{N_c} \left(||\boldsymbol{e}_i^c|| - \mu\right)^2.$$

   Se supondrá que E_c se distribuye en una esfera si CV es menor a 0,1.
2. En este paso se supone que los datos que viven aproximadamente en la frontera de una esfera, aplicando el test Rayleigh [18] se puede descartar que no siguen una distribución uniforme. Se continúan las comprobaciones si se rechaza el test con un nivel de significancia 0,05. Si no se puede rechazar el test, se supone que los datos no siguen un $\mathcal{FB}$.
3. Descartada la uniformidad, se comprueba si los elementos de E_c apuntan a la misma dirección. Se calcula

$$\overline{R} = \frac{1}{N_c}\sum_{i=1}^{N_c} \frac{\boldsymbol{e}_i^c}{||\boldsymbol{e}_i^c||},$$

   si $||\overline{R}||$ es mayor a 0,7. lo que indica que los datos están agrupados en al menos un cono de 45° en torno a su dirección media en un espacio de 768 dimensiones, entonces se supone que los datos de E_c apuntan aproximadamente en la misma dirección.

Si se superan estas comprobaciones, se supondrá que E_c sigue $\mathcal{FB}$. Como E_c no se distribuye exactamente sobre la superficie de la esfera, se va a suponer que la componente radial, $||E_c|| = \{||\boldsymbol{e}_i^c|| : \boldsymbol{e}_i^c \in E_c\}$, y la angular,

$\hat{E}_c = \{\hat{\boldsymbol{e}}_i^c = \dfrac{\boldsymbol{e}_i^c}{||\boldsymbol{e}_i^c||} : \boldsymbol{e}_i^c \in E_c\}$, son independientes. Así que cuando se tenga que muestrear un dato, se muestrean su componente angular y la radial por separado.

Para $||E_c||$, se contemplan tres distribuciones candidatas: una normal, una lognormal y una gamma. Se aplica un test estadístico a cada una de ellas siguiendo el orden normal, lognormal y gamma. Si un test se acepta, se para, y si se rechaza, se continúa al siguiente. Si todos los tests son rechazados, se supondrá que $||E_c||$ sigue una lognormal. En cualquier caso, $||E_c||$ va a seguir una distribución biparamétrica $||E_c|| \sim \mathcal{P}(\lambda_c^1, \lambda_c^2)$.

Respecto a la componente angular, primero se calcula la dirección media de los datos

$$\hat{\boldsymbol{\mu}}_c = \frac{\sum_{i=1}^{N_c} \boldsymbol{e}_i^c}{\left|\left|\sum_{i=1}^{N_c} \boldsymbol{e}_i^c\right|\right|}.$$

Se proyecta $\hat{E}_c$ estereográficamente al plano tangente a $\hat{\boldsymbol{\mu}}_c$, y se calcula la matriz de covarianza de los datos proyectados $\boldsymbol{A}_c$. Se realiza un test estadístico para saber si $\boldsymbol{A}_c$ es proporcional a la matriz identidad. Si no es proporcional, los *embeddings* siguen un Fisher-Bingham general $\hat{E}_c \sim \mathcal{FB}(\hat{\boldsymbol{\mu}}_c, \boldsymbol{A}_c)$. En caso contrario, se calcula κ_c con la estimación de Banerjee et al. [1] y sigue $\hat{E}_c \sim v\mathcal{MF}(\hat{\boldsymbol{\mu}}_c, \kappa_c)$.

Ahora bien, si alguna de las tres comprobaciones para suponer $\mathcal{FB}$ no es superada, se salta directamente a comprobar cuál de todas las mejores distribuciones de cada familia es la más adecuada para E_c. Como criterio de selección se usará BIC, dada una distribución paramétrica $\mathcal{P}(\Lambda)$ con número de parámetros $k = \#\Lambda$, se define el BIC de $\mathcal{P}(\Lambda)$ sobre E_c como

$$BIC(\mathcal{P}(\Lambda); E_c) = \log(N_c)k - 2\ell(\mathcal{P}(\Lambda); E_c),$$

donde ℓ es la log-verosimilitud total de E_c bajo la distribución $\mathcal{P}(\Lambda)$. En esta comparación se excluyen las distribuciones de $\mathcal{FB}$, ya que usan una cantidad considerablemente menor de parámetros respecto al resto de distribuciones planteadas según aumenta la dimensión de los *embeddings*. Por tanto, BIC compara $\mathcal{N}(\boldsymbol{\mu}_c, \boldsymbol{\Sigma}_c)$, $t(v_c, \boldsymbol{\xi}_c, \boldsymbol{\Psi}_c)$ y $\mathcal{L}(\tilde{\boldsymbol{\mu}}_c, \boldsymbol{B}_c)$ sobre E_c. Si la distribución con menor BIC es $\mathcal{P}_c(\Lambda_c)$, se supondrá $E_c \sim \mathcal{P}_c(\Lambda_c)$.

3.3. Proceso de olvido

Para olvidar, se aplican dos funciones de pérdida, una para olvidar las clases de $\mathcal{D}_f$, L_f, y otra para prevenir el olvido de $\mathcal{D}_r$, L_r. Distributional es un método *retain-free* ya que no tiene acceso a $\mathcal{D}_r$ en tiempo de olvido, pero lo simula a nivel de *embedding*. Se muestrean N_c^s elementos para cada clase $c \notin C_f$ bajo distribución $\mathcal{P}(\Lambda_c)$

$$\boldsymbol{e}_i^c \sim \mathcal{P}(\Lambda_c),\ 1 \leq i \leq N_s^c,$$

y se define el *dataset* de datos sintéticos $\mathcal{D}_s = \{(\boldsymbol{e}_i^c, c) : \ 1 \leq i \leq N_s^c, c \notin C_f\}$.

Por un lado, L_r es la entropía cruzada sobre $\mathcal{D}_s$ y 0 sobre $\mathcal{D}_f$. Por otro lado, L_f es 0 sobre $\mathcal{D}_s$ mientras que en $\mathcal{D}_f$ es una función que favorezca el DA, en este caso se ha optado por la siguiente función, denominada *notop*:

$$L_f(f_\theta(\boldsymbol{x}_i), y_i) = \text{mín}\left(0, \underset{j \neq y_i}{\text{máx}}\{f_\theta(\boldsymbol{x}_i)^j\} - f_\theta(\boldsymbol{x}_i)^{y_i} + m\right), \quad m \geq 0.$$

4. Experimentación

En esta sección, se evalúa la efectividad del método propuesto en el olvido incremental de clases. Como arquitectura se usa ViT-B/16 [9] preentrenado en Imagenet1K y *finetuneado* en cada *dataset* por 30 épocas. Para las comparaciones se usa el modelo con la mejor época según validación. Los métodos de DA son analizados sobre CIFAR-10 [14], CIFAR-100 [14] y MNIST [15].

En esta propuesta se tiene como uno de los hiperparámetros más importantes el número de datos sintéticos creados. En las comparativas se fija a 1000 los datos sintéticos a crear y que se mantendrán los mismos durante cada paso del olvido. El segundo hiperparámetro del método Distribucional es el margen de *notop* fijado a 10. Se divide $\mathcal{D}_f$ en *train* y validación, usando el conjunto de validación para evaluar el rendimiento del método en cada época. Si el *accuracy* en validación es menor que puramente aleatorio durante dos épocas seguidas se realiza un *early stopping*, ya que entrenar más solo podría empeorar el rendimiento del modelo. También, se usarán los pesos del modelo correspondientes a la época que tenga menor *accuracy* en el conjunto de validación.

Se compara Distribucional contra diversos métodos de DA *retain-free* que pueden o no usar intervención. De los comparados solo SSD [10] requiere de intervención, mientras que el resto no lo requieren, BADTEACHER[6], BOUNDARYEXPAND [5], BOUNDARYSHRINK [5], DELETE [23] y RANDOMLABEL. A excepción de SSD, todos los métodos requieren de un finetuneo. Dicho finetuneo seguirá el mismo procedimiento que el realizado con Distribucional para mantener todas las comparaciones justas. Además para CIFAR-100 incluimos la comparación con *Retrain* que es el modelo finetuneado sin nunca haber visto los datos siendo considerado el estándar de oro.

Las métricas consideradas son, por un lado, el *accuracy* en retención, es decir, el porcentaje de clases del conjunto de retención bien clasificadas y, por otro, el *accuracy* en olvido, o porcentaje de acierto en las clases a olvidar de test. También se muestra la evolución de las probabilidades máximas, reportando la probabilidad de la clase más probable, tanto en olvido como en retención. Un método funciona bien si tiene un alto *accuracy* y probabilidades máximas en retención, y un bajo *accuracy* en olvido. En cuanto a las probabilidades en el olvido, lo deseable es tener un rendimiento cercano a un modelo entrenado desde cero.

Se realizan 10 ejecuciones distintas en cada *dataset*. En cada ejecución de CIFAR-10 y MNIST se seleccionan 5 clases, y luego el modelo olvida cada clase

secuencialmente bajo cada método en el orden en que las clases fueron escogidas. En CIFAR-100, en cada ejecución se tomaron 20 clases.

En cuanto a las distribuciones seleccionadas por el flujo de trabajo descrito en la sección 3, en CIFAR-100 la práctica totalidad de las clases se ajustan a $v\mathcal{MF}$, mientras que en CIFAR-10 y MNIST predomina la $\mathcal{FB}$. Este comportamiento es esperable al trabajar sobre una arquitectura *transformer*, cuyos *embeddings* tienden a agruparse en direcciones. Que $v\mathcal{MF}$ no escale a $\mathcal{FB}$ en CIFAR-100 se debe probablemente a que, al contar cada clase con menos elementos, el test sobre $\boldsymbol{A}_c$ carece de potencia suficiente para rechazar la hipótesis de proporcionalidad a la identidad, es decir, para detectar la anisotropía angular presente.

Las figuras 1 y 2 recogen el *accuracy* y las probabilidades máximas de retención y de olvido en CIFAR-10 y MNIST. En ambos *datasets*, la mayoría de los métodos olvidan de forma efectiva sin apenas penalizar la retención, salvo SSD y BOUNDARYEXPAND, cuyo *accuracy* de olvido se mantiene elevado durante toda la secuencia. BADTEACHER destaca porque, aunque reduce inicialmente el *accuracy* de olvido, sus probabilidades máximas se aplanan con los pasos y dicho *accuracy* vuelve a subir; esto no ocurre en RANDOMLABEL, cuya etiqueta aleatoria se restringe a las clases de retención. Distribucional se mantiene a la altura de los métodos con comportamiento deseable en ambos *datasets*.

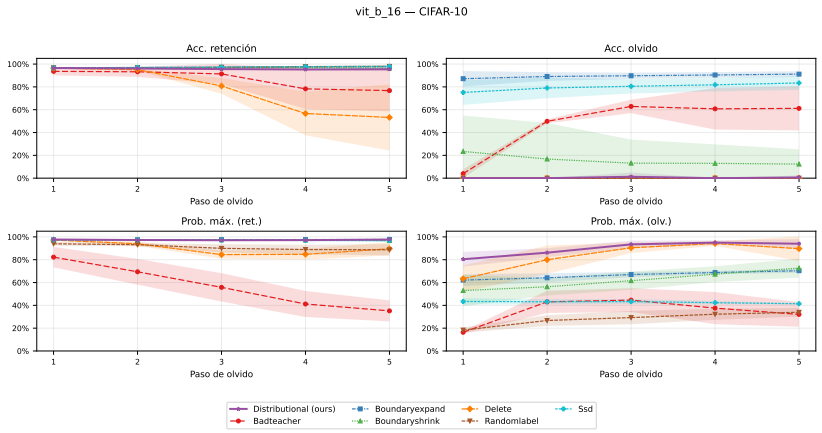


Figura 1: Resultados del olvido incremental de clases en CIFAR-10 con ViT-B/16. Cada subgráfico muestra la media y la desviación típica sobre 10 ejecuciones. El método Distribucional se representa con línea continua para facilitar su identificación.

La figura 3 y la tabla 1 recogen la comparativa en CIFAR-100, donde las diferencias entre métodos se acentúan respecto a CIFAR-10 y MNIST. SSD y BOUNDARYEXPAND mantienen el comportamiento deficiente ya observado: su *accuracy* de olvido no solo se mantiene alto, sino que crece con los pasos, hasta

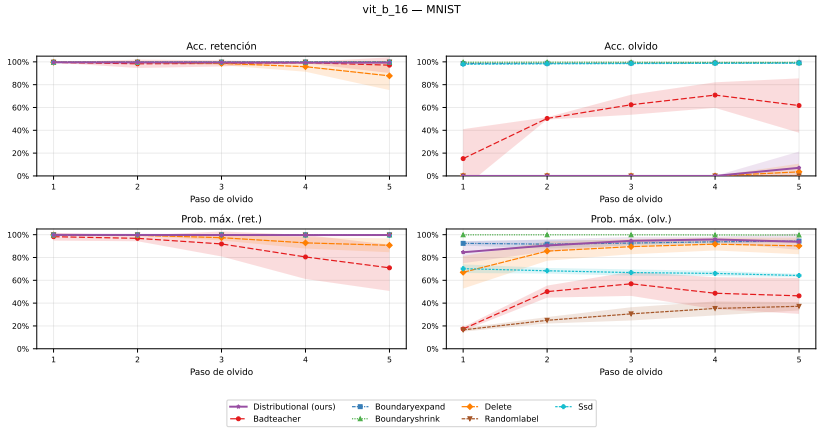


Figura 2: Resultados del olvido incremental de clases en MNIST con ViT-B/16. Cada subgráfico muestra la media y la desviación típica sobre 10 ejecuciones. El método Distribucional se representa con línea continua para facilitar su identificación.

0,713 y 0,752, respectivamente, en el paso 20. BADTEACHER, que en CIFAR-10 y MNIST conseguía inicialmente un olvido razonable, aquí es incapaz de reducir su *accuracy* de olvido por debajo de 0,799 en ningún paso, llegando a 0,838 en el paso 20. BOUNDARYSHRINK y DELETE sí logran bajar su *accuracy* de olvido, pero a costa de degradar por completo la retención, con caídas hasta 0,344 y 0,058, respectivamente, en el paso 20 frente al 0,863 de *Retrain*, lo que evidencia un olvido catastrófico más que un olvido selectivo. De entre los métodos comparados, solo RANDOMLABEL y Distribucional evitan ambos problemas, aunque difieren en la velocidad y el coste del olvido: RANDOMLABEL solo baja de un 25 % de *accuracy* de olvido a partir del paso 15 y mantiene la retención casi idéntica a *Retrain*, mientras que Distribucional se mantiene por debajo del 2 % de *accuracy* de olvido durante toda la secuencia, a cambio de una retención más moderada. Puesto que el objetivo es un olvido total en cada paso y no solo al final de la secuencia, Distribucional resulta el método más consistente con dicho objetivo entre los comparados.

La robustez del método resulta especialmente evidente en CIFAR-100, donde la mayoría de los métodos comparados sufren olvido catastrófico o son incapaces de mantener un olvido efectivo a medida que aumenta el número de clases desaprendidas. Este comportamiento pone de manifiesto que los datos sintéticos generados a partir de distribuciones ajustadas preservan de forma más fiable el conocimiento de las clases a retener, lo que se traduce en una mayor resiliencia ante el olvido acumulativo.

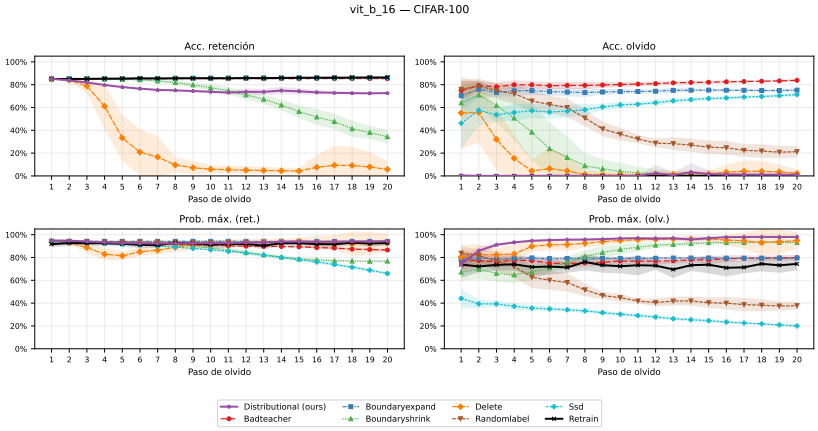


Figura 3: Resultados del olvido incremental de clases en CIFAR-100 con ViT-B/16. Cada subgráfico muestra la media y la desviación típica sobre 10 ejecuciones. El método Distribucional y *Retrain* se representan con línea continua para facilitar su identificación.

Tabla 1: Resultados de las métricas *accuracy* retención / *accuracy* olvido con 3 cifras significativas en CIFAR-100 en los pasos de olvido 5, 10, 15 y 20

	Paso 5	Paso 10	Paso 15	Paso 20
Retrain	0,853 / 0,000	0,856 / 0,000	0,860 / 0,000	0,863 / 0,000
SSD	0,854 / 0,572	0,858 / 0,623	0,860 / 0,679	0,863 / 0,713
BADTEACHER	0,852 / 0,799	0,853 / 0,801	0,852 / 0,821	0,853 / 0,838
BEXPAND	0,852 / 0,748	0,856 / 0,739	0,858 / 0,752	0,862 / 0,752
BSHRINK	0,847 / 0,384	0,774 / 0,039	0,564 / 0,008	0,344 / 0,004
DELETE	0,333 / 0,044	0,059 / 0,008	0,043 / 0,018	0,058 / 0,023
RANDOMLABEL	0,851 / 0,657	0,856 / 0,366	0,859 / 0,252	0,862 / 0,213
Distribucional	0,779 / 0,000	0,738 / 0,000	0,742 / 0,016	0,726 / 0,010

5. Conclusiones

En este trabajo se ha presentado el método Distribucional, un método de DA *retain-free* con intervención diseñado para el escenario de olvido incremental de clases. La propuesta extiende el método PASS al seleccionar, para cada clase, la distribución paramétrica que mejor se ajusta a sus *embeddings*. Para ello, se elabora flujo de trabajo basado en tests estadísticos y el criterio BIC, generando así datos sintéticos más representativos que los obtenidos bajo la suposición gaussiana isotrópica estándar. Los resultados experimentales sobre CIFAR-10, CIFAR-100 y MNIST con ViT-B/16 muestran que el método propuesto es el que tiene mejor comportamiento a lo largo de todos los pasos de olvido secuencial, manteniendo un rendimiento competitivo en retención y reduciendo eficazmente el *accuracy* en las clases olvidadas.

Entre las limitaciones del método cabe señalar la necesidad de la intervención inicial. Esto supone que los datos originales deben estar disponibles en ese momento, lo que puede no ser siempre factible en aplicaciones reales. Otra gran limitación es la propia necesidad de clases para aplicar el método, lo que lo hace inaplicable a modelos de recomendación.

Como trabajo futuro, se plantea extender la evaluación a un mayor número de *datasets* y arquitecturas, estudiar alternativas a la función de pérdida *notop*, así como el uso de conjuntos sintéticos de mayor tamaño y técnicas de inversión de *embeddings* para entrenar capas más profundas del modelo durante el proceso de olvido. Otro camino a explorar es una posible extensión a tareas distintas a clasificación, como la detección de objetos o la segmentación semántica, donde el concepto de clase sigue presente pero el esquema de entrenamiento difiere sustancialmente. Asimismo, resulta relevante estudiar la adaptación del método a escenarios con datos no etiquetados o parcialmente etiquetados, en los que la estimación de la distribución por clase debe inferirse mediante técnicas de agrupamiento o aprendizaje semisupervisado, ampliando así el espectro de aplicaciones prácticas del método. Por último, es especialmente interesante explorar el potencial de auto-olvido de la propuesta: dado que los parámetros distribucionales se extraen en la fase de intervención y no se requiere acceso al conjunto de datos original durante el olvido propiamente dicho, el método podría aplicarse de forma iterativa sin necesidad de almacenar datos reales; estudiar hasta qué punto esta propiedad se mantiene a lo largo de muchos pasos, y en qué medida la intervención inicial puede relajarse, lo que constituye una dirección prometedora.

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 PID2023-150070NB-I00, financiado por MICIU/AEI/10.13039/501100011033 y por FEDER, UE.

Declaración de intereses. Los autores declaramos que no tenemos ningún conflicto de intereses.

Referencias

1. Banerjee, A., Dhillon, I.S., Ghosh, J., Sra, S., Ridgeway, G.: Clustering on the Unit Hypersphere using von Mises-Fisher Distributions. Journal of Machine Learning Research **6**(9) (2005)
2. Bonato, J., Cotogni, M., Sabetta, L.: Is retain set all you need in machine unlearning? restoring performance of unlearned models with out-of-distribution images. In: European Conference on Computer Vision. pp. 1–19. Springer (2024)
3. Bourtoule, L., Chandrasekaran, V., Choquette-Choo, C.A., Jia, H., Travers, A., Zhang, B., Lie, D., Papernot, N.: Machine unlearning. In: 2021 IEEE symposium on security and privacy (SP). pp. 141–159. IEEE (2021)
4. Cao, Y., Yang, J.: Towards making systems forget with machine unlearning. In: 2015 IEEE symposium on security and privacy. pp. 463–480. IEEE (2015)
5. Chen, M., Gao, W., Liu, G., Peng, K., Wang, C.: Boundary unlearning: Rapid forgetting of deep networks via shifting the decision boundary. In: Proceedings

of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. pp. 7766–7775 (2023)
6. Chundawat, V.S., Tarun, A.K., Mandal, M., Kankanhalli, M.: Can bad teaching induce forgetting? unlearning in deep networks using an incompetent teacher. Proceedings of the AAAI Conference on Artificial Intelligence **37**(6), 7210–7217 (2023)
7. Claeskens, G., Hjort, N.L.: Model selection and model averaging. Cambridge books (2008)
8. Devalapally, A., Valluri, G.: A simple machine unlearning approach using elastic weight consolidation. In: International Conference on Recent Trends in AI Enabled Technologies. pp. 1–10. Springer (2023)
9. Dosovitskiy, A., Beyer, L., Kolesnikov, A., Weissenborn, D., Zhai, X., Unterthiner, T., Dehghani, M., Minderer, M., Heigold, G., Gelly, S., et al.: An image is worth 16x16 words: Transformers for image recognition at scale. arXiv preprint arXiv:2010.11929 (2020)
10. Foster, J., Schoepf, S., Brintrup, A.: Loss-free machine unlearning. In: The Second Tiny Papers Track at ICLR 2024, Tiny Papers @ ICLR 2024, Vienna, Austria, May 11, 2024 (2024)
11. Henze, N., Zirkler, B.: A class of invariant consistent tests for multivariate normality. Communications in statistics-Theory and Methods **19**(10), 3595–3617 (1990)
12. Kotz, S., Balakrishnan, N., Johnson, N.L.: Continuous multivariate distributions, Volume 1: Models and applications, vol. 334. John wiley & sons (2019)
13. Krawczyk, A., Gepperth, A.: Continual unlearning through memory suppression. ESANN Proceedings (2025)
14. Krizhevsky, A., Hinton, G., et al.: Learning multiple layers of features from tiny images (2009)
15. LeCun, Y., Bottou, L., Bengio, Y., Haffner, P.: Gradient-based learning applied to document recognition. Proceedings of the IEEE **86**(11), 2278–2324 (2002)
16. Ledoit, O., Wolf, M.: A well-conditioned estimator for large-dimensional covariance matrices. Journal of multivariate analysis **88**(2), 365–411 (2004)
17. Li, B., Zhou, H., He, J., Wang, M., Yang, Y., Li, L.: On the sentence embeddings from pre-trained language models. In: Proceedings of the 2020 conference on empirical methods in natural language processing (EMNLP). pp. 9119–9130 (2020)
18. Mardia, K.V., Jupp, P.E.: Directional statistics. John Wiley & Sons (2009)
19. Rolnick, D., Ahuja, A., Schwarz, J., Lillicrap, T., Wayne, G.: Experience replay for continual learning. Advances in neural information processing systems **32** (2019)
20. Voigt, P., Von dem Bussche, A.: The EU general data protection regulation (GDPR). A practical guide, 1st ed., Cham: Springer International Publishing **10**(3152676), 10–5555 (2017)
21. Wang, L., Zhang, X., Su, H., Zhu, J.: A comprehensive survey of continual learning: Theory, method and application. IEEE transactions on pattern analysis and machine intelligence **46**(8), 5362–5383 (2024)
22. Zhao, H., Ni, B., Fan, J., Wang, Y., Chen, Y., Meng, G., Zhang, Z.: Continual forgetting for pre-trained vision models. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. pp. 28631–28642 (2024)
23. Zhou, Y., Zheng, D., Mo, Q., Lu, R., Lin, K.Y., Zheng, W.S.: Decoupled distillation to erase: A general unlearning method for any class-centric tasks. In: Proceedings of the Computer Vision and Pattern Recognition Conference. pp. 20350–20359 (2025)
24. Zhu, F., Zhang, X.Y., Wang, C., Yin, F., Liu, C.L.: Prototype augmentation and self-supervision for incremental learning. In: Proceedings of the IEEE/CVF conference on computer vision and pattern recognition. pp. 5871–5880 (2021)