I302 - Aprendizaje Automático y Aprendizaje Profundo

Trabajo Práctico 4: Aprendizaje no Supervisado

Nazareno Gonella

31 de mayo de 2025

Ingeniería en Inteligencia Artificial

1. Clustering de Datos

Resumen

En esta parte, se analizaron tres métodos de clústering de aprendizaje no supervisado, siendo K-means, Gaussian Mixture Models, y DBScan. En K-means se identificó $K = 15$ como número óptimo de clústeres mediante el método del codo, con un distance squared error de 308,94. En GMM, inicializando los centroides con los resultantes del K-means, y $K = 15$, se alcanzó un log-likelihood de $-8715{,}46$. Para DBSCAN, se estimó un rango óptimo de ϵ entre 0,05 y 0,10 con $minPoints = 10$ utilizando el método de la distancia k-ésima, logrando identificar 15 clústers distintos.

1.1. Métodos

1.1.1. K-Means

Busca particionar un conjunto de datos en K clústeres minimizando la distancia euclídea dentro de cada clúster. Se inicia con K centroides aleatorios en el espacio $\mathbb{R}^d$. En cada iteración, se asigna cada muestra x_i al clúster cuyo centroide es el más cercano:

$$z_i = \arg\min_k \|x_i - \mu_k\|^2 \tag{1}$$

Y luego se recalculan los centroides utilizando el promedio de las muestras recién asignadas:

$$\mu_k = \frac{1}{N_k} \sum_{i:z_i=k} x_i \tag{2}$$

donde N_k es la cantidad de muestras en el clúster k.

1.1.2. Método del Codo

Técnica heurística utilizada para determinar el número óptimo de clústeres K en algoritmos de agrupamiento. Consiste en graficar la variación de la función de error al aplicar el algoritmo con diferentes valores de K. Se selecciona el valor a partir del cual las mejoras empiezan a disminuir notablemente.

1.1.3. Gaussian Mixture Models

Asume que los datos se generan a partir de una combinación de K distribuciones normales multivariadas. Cada clúster k está representado por una distribución gaussiana con media μ_k y matriz de covarianza Σ_k, y se le asigna un peso π_k tal que $\sum_{k=1}^{K} \pi_k = 1$. La función de densidad del modelo se define como:

$$p(x) = \sum_{k=1}^{K} \pi_k \, \mathcal{N}(x \mid \mu_k, \Sigma_k) \tag{3}$$

Los parámetros π_k, μ_k y Σ_k se estiman mediante el algoritmo de Expectation-Maximization (1.1.4).

1.1.4. Expectation-Maximization

El algoritmo Expectation-Maximization (EM) estima los parámetros de modelos con variables latentes mediante una optimización iterativa de la verosimilitud. En cada iteración t, alterna entre dos pasos:

- **Paso E (Expectation):** se calcula la responsabilidad $r_{ik}^{(t)}$, que es la probabilidad posterior de que la muestra x_i provenga del componente k bajo los parámetros actuales $\theta^{(t)}$:

$$r_{ik}^{(t)} = \frac{\pi_k^{(t)}\, p(x_i \mid \mu_k^{(t)}, \Sigma_k^{(t)})}{\sum_{k'} \pi_{k'}^{(t)}\, p(x_i \mid \mu_{k'}^{(t)}, \Sigma_{k'}^{(t)})}$$

- **Paso M (Maximization):** se actualizan los parámetros del modelo:

$$\mu_k^{(t+1)} = \frac{\sum_i r_{ik}^{(t)}\, x_i}{r_k^{(t)}}, \quad \Sigma_k^{(t+1)} = \frac{\sum_i r_{ik}^{(t)} (x_i - \mu_k^{(t+1)})(x_i - \mu_k^{(t+1)})^\top}{r_k^{(t)}}, \quad \pi_k^{(t+1)} = \frac{r_k^{(t)}}{N}$$

1.1.5. DBSCAN

DBSCAN detecta regiones densas separándolas del ruido, con dos parámetros: ε (radio) y $minPts$ (mínimos puntos para núcleo).

Un punto p es:

- **Núcleo**: si dentro del radio ε alrededor de p contiene al menos $minPts$ puntos.
- **Alcanzable por densidad**: si está en rango de ε de un núcleo.
- **Ruido**: si no es núcleo ni alcanzable.

1.1.6. Método de la Distancia k-ésima

Técnica heurística utilizada para estimar el parámetro ϵ en DBSCAN. Consiste en calcular para cada punto la distancia a su k-ésima vecino más cercano (k = min_points) y graficar estas distancias ordenadas de menor a mayor. Se selecciona el valor de ϵ en el punto donde la pendiente de la curva cambia abruptamente, similar al método del codo.

1.1.7. Métricas

- **Distance Squared Error (DSE)**:

$$\text{DSE} = \sum_{i=1}^{N} \left\| \mathbf{x}_i - \boldsymbol{\mu}_{k_i} \right\|^2$$

donde N es la cantidad de muestras, $\mathbf{x}_i$ es el vector de características de la muestra i, y $\boldsymbol{\mu}_{k_i}$ es el centroide más cercano a $\mathbf{x}_i$.

- **Log-Likelihood (LL)** para GMM:

$$\text{LL} = \sum_{i=1}^{N} \log \left(\sum_{k=1}^{K} \pi_k \mathcal{N}(\mathbf{x}_i \mid \boldsymbol{\mu}_k, \Sigma_k) + \epsilon \right)$$

donde π_k son los coeficientes de mezcla, y $\mathcal{N}(\cdot)$ es la función de densidad de la distribución normal multivariada con media $\boldsymbol{\mu}_k$ y covarianza Σ_k

1.1.8. Datos

- `clustering:` 4999 muestras, cada una siendo un arreglo de 2 números enteros (A, B).

1.2. Desarrollo

Se utiliza el conjunto de datos `clustering.csv`, que cuenta con 4999 muestras de 2 componentes.

Para K-means y GMM, se ajustó varias veces cada modelo, conservándose aquel con mejores resultados en sus respectivas funciones de pérdida (distance squared error y log likelihood). Esto se realiza para contrarrestar la naturaleza estocástica de ambos modelos.

Primero se realizó K-means, inicializando cada centroide en la posicion de una muestra aleatoria, para luego evaluar valores de K en el rango $[1, 40]$. Se aplicó el método del codo (Figura 1) y se halló que el valor óptimo es $K = 15$.

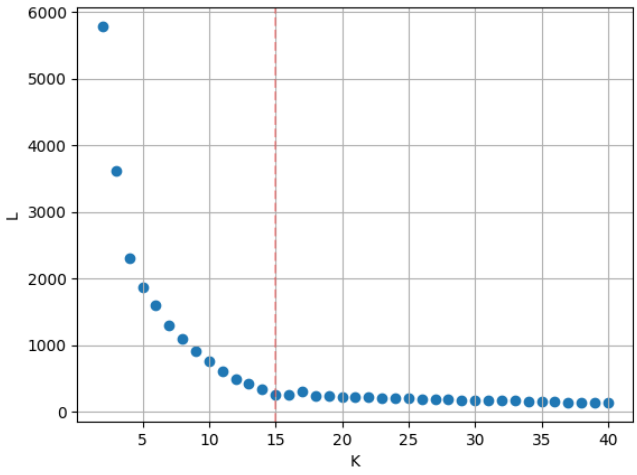


Figura 1: Método del Codo

El modelo GMM se inicializó con los centroides obtenidos por K-means para μ_k, y con una covarianza global compartida por todos los clústeres, calculada sobre el conjunto de datos. Con $K = 15$, la optimización del log-likelihood alcanzó un valor de $-8715{,}46$ para esta configuración.

Para DBSCAN, se exploraron combinaciones de los parámetros ϵ y $minPoints$. Para $minPoints = 10$, se utilizó el método de la distancia k-ésima (Figura 2) para estimar un rango adecuado de ϵ, que se encontró aproximadamente entre 0,05 y 0,10. La configuración ($minPoints = 10$, $epsilon = 0{,}10$) resultó en la detección de 15 clústeres.

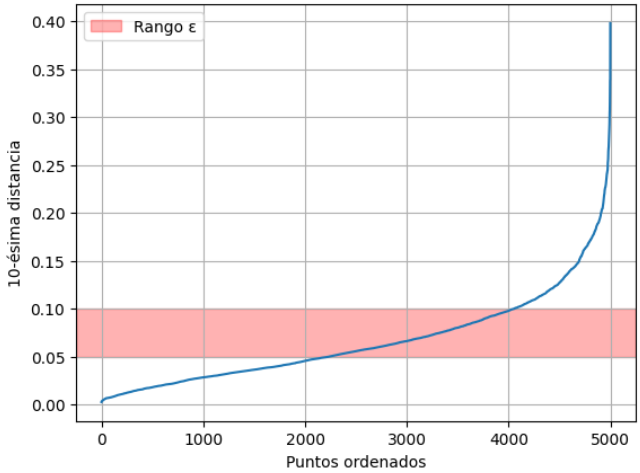
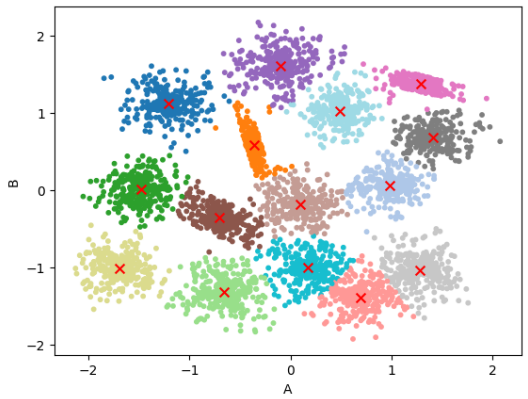
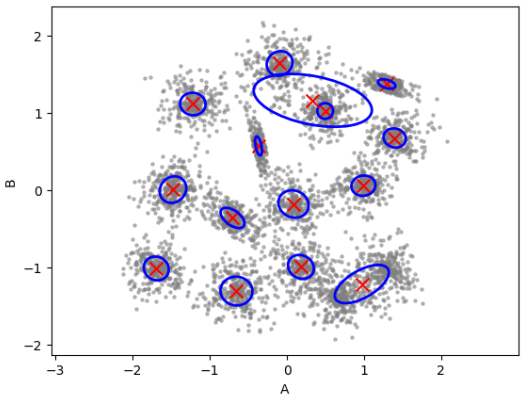


Figura 2: Método de la distancia k-ésima

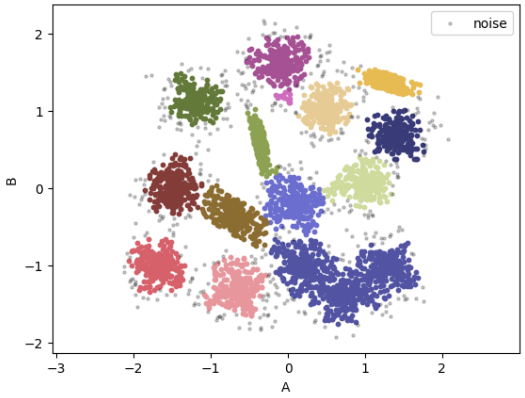
1.3. Resultados



(a) K-means: K = 15



(b) GMM: K = 15



(c) fig

Figura 3: DBScan: $minPoints = 10$, $\epsilon = 0{,}10$, 15 clústers

2. Reducción de Dimensionalidad

Resumen

Se implementó Análisis de Componentes Principales (PCA) sobre el dataset MNIST, compuesto por imágenes de las cifras del 0 al 9 de 28x28 píxeles, representados como vectores de 784 dimensiones. Se evaluó el error cuadrático medio (MSE) de reconstrucción en función del número de componentes principales a eliminar, identificando $k = 625$ como la cantidad óptima, con un valor de MSE = 207,71.

2.1. Métodos

2.1.1. Análisis de Componentes Principales (PCA)

Método de reducción de dimensionalidad que transforma un conjunto de variables correlacionadas en un conjunto menor de variables no correlacionadas llamadas componentes principales. Estas componentes capturan la mayor varianza posible de los datos.

Se calculan los autovectores y autovalores de la matriz de covarianza de los datos, para luego proyectar los datos originales sobre los autovectores asociados a los mayores autovalores. Se descartan los autovectores correspondientes a los menores k autovalores.

2.1.2. Métricas

- **Mean Squared Error (MSE)**:

$$\text{MSE} = \frac{1}{N}\sum_{i=1}^{N}(y_i - \hat{y}_i)^2$$

donde N es la cantidad de muestras, y_i el valor real y $\hat{y}_i$ el valor predicho.

2.1.3. Datos

- `MNIST_dataset:` 70000 muestras. Cada muestra representa una imagen con de 28x28 píxeles, vectorizada en un arreglo de 784 números enteros en el rango [0, 255]. Cada imagen también cuandta con su respectivo label, un valor entero en el rango [0, 10].

2.2. Desarrollo

Se separan las etiquetas del conjunto de datos `MNIST_dataset`, quedando Y_labels y las imágenes X. Se consigue la covarianza de X centrado, y se calculan los autovalores y autovectores de la matriz resultante.

En esta implementación se eliminan los autovectores asociados a los k autovalores mas pequeños, quedándose con las componentes principales que mejor representan la variabilidad de los datos.

Como se observa en la figura 4, se determina que el mejor valor de k es el valor promedio de los errores cuadráticos medios entre las imágenes originales y sus reconstrucciones. De esta manera se elige un valor de k previo a un aumento abrupto del error.

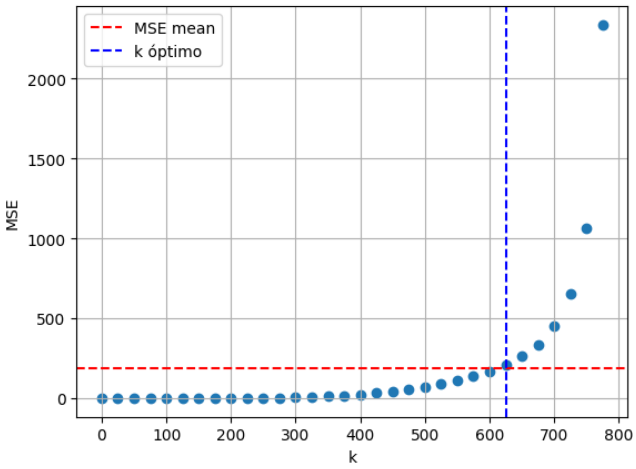


Figura 4: k óptimo a través del valor promedio del MSE ($k = 625$)

2.3. Resultados

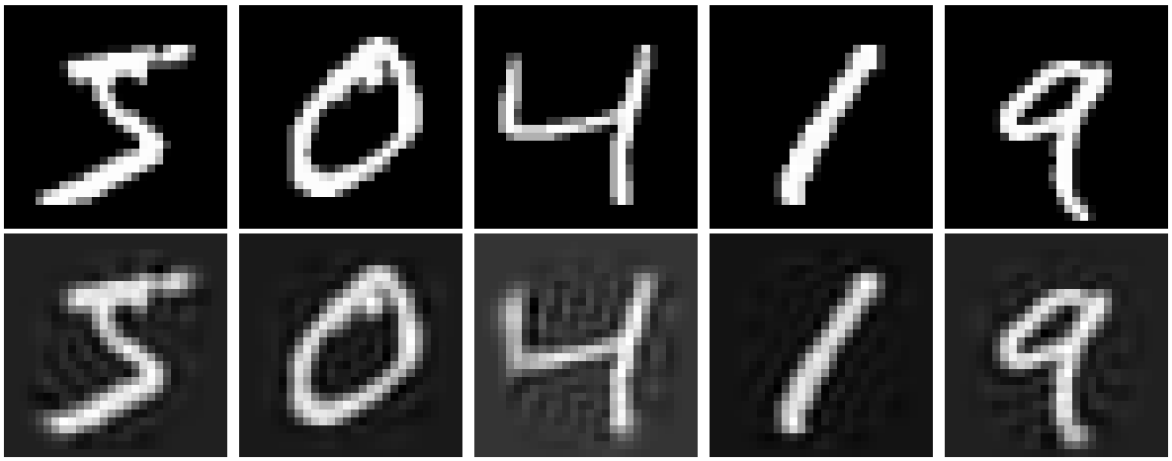


Figura 5: Comparación entre imágenes (Índices de 0 a 4). Arriba: Imágenes originales ($k = 0$). Abajo: Imágenes con dimensionalidad reducida ($k = 625$)

Este valor de k permite conservar gran parte de la imagen original, aún habiendo descartado casi un 80 % de las componentes principales menos significativas.