Universidad de SanAndrés

I302 - Aprendizaje Automático y Aprendizaje Profundo

2do Semestre 2025

Trabajo Práctico 4

Fecha de entrega: Lunes 17 de Noviembre, 23:59 hs.

Formato de entrega: Los archivos desarrollados deberán entregarse en un archivo comprimido (.zip) a través del Campus Virtual, utilizando el siguiente formato de nombre: *Apellido_ Nombre_ TP4.zip*. Se aceptará únicamente un archivo por estudiante y debe contener por lo menos los siguientes elementos:

```
Apellido_Nombre_TP4.zip/
|- data/
|- Apellido_Nombre_Informe_TP4.pdf
|- Apellido_Nombre_Notebook_TP4.ipynb
```

- **Informe:** Debe incluir todos los aspectos teóricos, decisiones metodológicas, visualizaciones, análisis y conclusiones. El objetivo es que el informe contenga toda la explicación principal del trabajo. Se puede hacer referencia al notebook con frases como "Ver sección X del notebook para la implementación". El informe debe entregarse utilizando el archivo `template_informe.tex` provisto y no debe exceder las 10 páginas.
- **Notebook:** Debe contener el código utilizado, experimentos, análisis exploratorio, gráficos y el proceso completo de procesamiento y modelado. Sirve como respaldo técnico del informe y debe estar ordenado y bien documentado. Se recomienda modularizar el código en archivos `.py` cuando sea posible.

Trabajo Práctico 4: Aprendizaje No-Supervisado

El objetivo de este trabajo es desarrollar y evaluar metodos de aprendizaje no supervisado. **No se permite usar librerías de machine learning como scikit-learn o PyTorch para implementar los metodos solicitados, a menos que sea pedido explícitamente en el enunciado del ejercicio.**

1. **Inspección de los datos** El dataset *caras.csv* contiene 800 imágenes de 68×68 pixeles en escala de grises, junto con sus etiquetas de clase. Para realizar un análisis exploratorio de estos datos, se proponen los siguientes puntos:

   a) Implementar una función que pueda graficar un número arbitrario de imágenes del dataset. Mostrar un ejemplo de un gráfico con 15 imágenes seleccionadas de manera aleatoria. Esta función va a ser necesaria a lo largo de todo el trabajo, por lo que se deberá ir adaptando de acuerdo a las necesidades que vayan surgiendo.

   b) Realizar un analisis exploratorio enfocado en la distribución de clases en los datos. Para tener una mejor noción de la variabilidad de las muestras, graficar muestras agrupadas por clase, para al menos 5 clases.

   c) Realizar un split de datos estratificado para conformar un conjunto de entrenamiento y otro de evaluación, en donde los datos de evaluación sean el 20 % del total.

2. **Reducción de dimensionalidad**

   a) Implementar una función para estandarizar los datos, y otra para realizar Principal Component Analysis (PCA). Estandarizar y aplicar PCA sobre los datos de entrenamiento. Con la transformacion aprendida previamente reducir la dimensionalidad de los datos de evaluacion.

   b) Realizar un gráfico que muestre la varianza explicada acumulada en función del número de componentes. Conservar el número de componentes que aseguren explicar el 90 % de la varianza de los datos. Utilizando la función aprendida en el punto 1.a), realizar un gráfico comparativo entre las imágenes originales y la reconstrucción partiendo del número de componentes determinado anteriormente.

   c) Entrenar un modelo de autoencoder deterministico (AE) utilizando la librería PyTorch para armar y entrenar las redes neuronales involucradas (la red de encoder y la de decoder). Para hacer una comparacion justa, la dimension latente debe ser la misma que la obtenida en el PCA. Recuerde dividir el conjunto de datos en dos subconjuntos: entrenamiento y validación, los mismos utilizados para el PCA. El subconjunto de entrenamiento se empleará para entrenar el AE, mientras que el de validación servirá para ajustar los hiperparámetros (regularizacion, arquitectura o lo que considere) y evaluar el error de reconstrucción.

   d) Una vez desarrollado el AE, compare la calidad de las imágenes reconstruidas con las obtenidas mediante PCA en el inciso anterior, utilizando 10 imagenes tomadas aleatoriamente del conjunto de validación del AE.

   e) Para los siguientes puntos, en lugar de trabajar con las imágenes originales vamos a trabajar con su versión de menor dimensionalidad en el espacio latente del PCA y AE obtenida en el inciso anterior para acelerar el procesamiento. Aplique la transformación para todos los datos, tanto de entrenamiento como de evaluación usando la transformacion aprendida con los datos de entrenamiento.

3. **Clustering**

   a) Desarrollar una funcion que implemente el algoritmo k-Means. Probar con valores de K dentro del rango de $[5, 20]$.

   b) Desarrollar una funcion que implemente el algoritmo GMM. Probar con valores de K dentro del rango de $[5, 20]$.

   c) Analizar el desempeño para distintos valores de K. Utilizar el método de *ganancias decrecientes*, junto con el método de *Silhouette score*. Graficar tanto la ganancia marginal como el valor de Silhouette score en función de K. ¿Qué se puede concluir sobre K observando estas curvas? ¿Cómo se relaciona el numero de K encontrado con el número de clases presentes en los datos? Justificar.

   d) Del mejor valor de K resultante del análisis anterior graficar las muestras de cada cluster en 2D utilizando la función de reduccion dimensional aprendida en el punto 1. ¿Qué puede decir de la calidad de los clusters de acuerdo a la cantidad de muestras que agrupa y la homogeneidad de clases en cada cluster? Realizar los gráficos/tablas/análisis que crea conveniente para fundamentar su respuesta.