AUTOENCODERS DETERMINÍSTICOS

Asignatura – Redes De Neuronas

Descripción breve

El objetivo de este proyecto es implementar un autoencoder profundo determinístico sobre los conjuntos de datos MNIST y Fashion-MNIST (FMNIST)

Adriana García, Sara Marianova y Sara Suárez

Tabla de contenido

1. Introducción

El objetivo de este proyecto es implementar un autoencoder profundo determinístico sobre los conjuntos de datos MNIST y Fashion-MNIST, explorando diferentes arquitecturas, tamaños de espacio latente y técnicas de regularización. Se busca analizar la representación comprimida, la capacidad de reconstrucción y generalización de los modelos.

2. Metodología

2.1 Métrica PSNR

Se usa la métrica PSNR para medir la calidad de la reconstrucción de las imágenes. Cuanto más alto es el valor de esta métrica, la imagen está mejor reconstruida.

2.2 Grid Search

Se realizó una búsqueda sistemática con los siguientes valores.

- **latent_dim**: {15, 30, 50, 100, 300, 600} para probar desde mucha compresión hasta cuellos de botella amplios.
- **LR**: {1e-3, 1e-4} porque son valores estables.
- **λL1**: {0.0, 1e-4, 1e-3} para comparar sin regularización y con dos niveles de penalización del latente.
- **Dropout**: {0.0, 0.1, 0.3} para medir su efecto de regularización suave y moderada.
- Se entrenó 30 epochs como equilibrio entre coste y convergencia.

2.3 Modelos y experimentos probados

- **Convolutional Autoencoder**: versiones de 3 y 5 capas aplicadas a ambos datasets.
- **Denoising Autoencoder:** se entrena con ruido gaussiano para aprender a reconstruir imágenes limpias, usando las mejores variantes CNN-3L y CNN-5L, evaluadas con PSNR y early stopping.
- **Visualización de embeddings**: se extrajo el vector latente z y se aplicaron t-SNE y PCA para observar la separabilidad de clases y la estructura del espacio latente.
- **MLP Autoencoder**: red totalmente conectada entrenada con MSE + L1 sobre z y comparada con las versiones CNN en eficiencia y rendimiento**.**
- **Generalización cruzada**: Se entrena el modelo en un conjunto y se evalúa en el otro para medir la generalización en las tres arquitecturas (3 capas y 5 capas de la CNN y MLP). Se observa el PSNR medio y desviación en origen y destino, y se usa el mejor modelo preentrenado para cada arquitectura.
- **Uso generativo del decoder**: muestreo desde distribuciones $z \sim \mathcal{N}(0, I)$ y $\mathcal{N}(\mu, \Sigma_{\text{diag}})$ para generar imágenes sintéticas, comparando arquitecturas y datasets.

3. Resultados obtenidos

3.1 Comparativa MNIST con 3 capas vs 5 capas

En MNIST, la CNN de 3 capas supera a la de 5 por ~3 dB (31,97 vs 28,74), ofreciendo reconstrucciones más nítidas. Ambas mejoran con mayor latente, pero la de 5 capas se resiente más cuando el cuello de botella es pequeño. Con 30–40 épocas, 3 capas converge mejor con

LR=1e-3, mientras que 5 capas requieren LR=1e-4 o más épocas. La L1 solo aporta mejoras modestas y el dropout perjudica en ambos casos, especialmente en la red más profunda.

3.2 Comparativa FMNIST con 3 capas vs 5 capas

En FMNIST, la CNN de 3 capas supera a la de 5 por ~4 dB (28,57 vs 24,57) porque la más profunda suaviza en exceso y pierde texturas finas. Ambas ganan con mayor latente, pero la de 5 capas es muy frágil con cuellos de botella pequeños. Con 30 épocas, 3 capas rinden mejor con LR=1e-3, mientras que 5 capas requieren LR=1e-4 o más épocas. La L1 solo aporta mejoras leves y el dropout perjudica en ambas, especialmente en 5 capas.

3.3 Comparativa global de MNIST vs FMNIST con CNN autoencoder

MNIST es más fácil de reconstruir que FMNIST: con 3 capas logra 31,97 dB vs 28,57 dB y con 5 capas 28,74 dB vs 24,57 dB, siempre a favor de MNIST. La razón es que los dígitos tienen trazos limpios, mientras que la ropa incluye texturas finas y bordes difusos que penalizan más el PSNR. En ambos casos, mayor latente ⇒ mayor PSNR, pero FMNIST sufre más con latentes pequeñas. El LR óptimo depende de la profundidad (1e-3 en 3 capas, 1e-4 en 5) y la L1 solo aporta mejoras leves; el dropout perjudica sistemáticamente, sobre todo en FMNIST.

3.4 Comparativa de MNIST vs FMNIST en MLP

Con MLP, MNIST rinde mejor que FMNIST (26.07 vs 23.32 dB con latent=600 y LR=1e-3) porque los dígitos son más simples que las prendas. Aumentar la dimensión latente siempre ayuda, pero MNIST lo aprovecha más y FMNIST satura antes. Con 30 épocas, LR=1e-3 supera claramente a 1e-4; L1 solo añade mejoras pequeñas y el dropout perjudica (ya con 0.1 cae varios dB). En resumen: para MLP, usa latente grande y LR=1e-3; para cerrar la brecha en FMNIST, mejor pasar a CNN o entrenar más épocas.

3.5 Comparativa Global: MLP vs 3 Capas vs 5 Capas

La CNN de 3 capas logra el mejor PSNR en ambos datasets (≈31.97 dB en MNIST, ≈28.57 dB en FMNIST), por encima de la CNN de 5 capas (≈28.74/≈24.57 dB) y del MLP (≈26.07/≈23.32 dB). La de 5 capas suaviza en exceso y no converge tan bien con el mismo presupuesto; el MLP, sin sesgo espacial, queda atrás. Todas mejoran con latente grande; con 30–40 épocas, LR=1e-3 supera a 1e-4, y la L1 solo aporta ajustes menores. Evita dropout y, para maximizar PSNR, usa CNN 3 capas con latente amplio y LR=1e-3.

3.6 Denoising Autoencoder

Al comparar los autoencoders de 3 capas (3L) y 5 capas (5L) con MLP y CNN en MNIST y Fashion-MNIST, se observa que el modelo de 3L en CNN presenta resultados más eficientes y robustos, logrando reconstrucciones más nítidas frente al modelo de 5 capas de CNN, ya que obtiene mejores resultados de PSNR en casi todos los niveles de ruido. El PSNR disminuye de manera esperable al aumentar el ruido, pero el 3L de CNN mantiene una curva más estable y conserva mejor los detalles finos de las imágenes mientras que el 5L de CNN tiende a suavizar las reconstrucciones, perdiendo nitidez y necesitando más tiempo de entrenamiento para converger. En MLP se obtiene que con σ bajo, el PSNR mejora ligeramente y alcanza su máximo alrededor de σ≈0.1–0.2, a partir de ahí desciende con rapidez al aumentar el ruido. Esto refleja que el denoising autoencoder con MLP, entrenado con σ_train=0.3, generaliza bien para ruidos moderados pero sufre cuando σ se aleja de ese valor. MNIST mantiene PSNR superiores a

FMNIST porque sus dígitos son más simples de reconstruir que las texturas de ropa. Para robustez, conviene ajustar σ_train o emplear entrenamiento con múltiples niveles de ruido.

3.7 Embeddings

Se extrajeron embeddings latentes de los autoencoders de 3 y 5 capas de CNN para MNIST y FMNIST, y se proyectaron a 2D usando t-SNE y PCA. Utilizando t-SNE, se observa que las clases se separan claramente en todos los modelos, mostrando que los embeddings capturan información discriminativa. Por contra, con PCA las clases presentan gran solapamiento y baja varianza explicada (≈10–17%), indicando que la estructura de clases no se refleja en las primeras dos componentes principales. Por último, no se observan diferencias significativas entre las arquitecturas de 3 y 5 capas en CNN, ya que ambas aprenden embeddings útiles para diferenciar clases, aunque PCA no consigue representarlo. En MLP el t-SNE separa mejor las clases tanto en MNIST como en FMNIST, siendo un poco mejor en MNIST. Por el contrario, PCA no logra realizar una buena separación.

3.8 Comparativa de mejores y peores reconstrucciones de imágenes

Se compararon las mejores y peores reconstrucciones por modelo usando su PSNR individual. En MNIST, la CNN de 3 capas logró el mayor PSNR y las imágenes más nítidas; la CNN de 5 capas quedó por detrás con más suavizado, y el MLP mostró más artefactos al carecer de inductivo espacial. En FMNIST se repite el patrón: la CNN 3L mantiene mejor forma y textura, la 5L pierde detalle y el MLP rinde peor. El DAE mejora la robustez frente a ruido gaussiano alrededor de σ≈0.3 y mantiene PSNR cercanos al modelo base, aunque introduce más suavizado en los peores casos. En conjunto, la CNN de 3 capas ofrece el mejor equilibrio entre fidelidad, nitidez y generalización; la 5 capas y el MLP son menos eficientes, y el DAE gana robustez a costa de algo de textura.

3.9 Entrenamiento cruzado

Se observa una clara asimetría: entrenar en FMNIST generaliza bien a MNIST, mientras que entrenar en MNIST apenas se transfiere a FMNIST. Con la CNN de 3 capas, entrenar en FMNIST y evaluar en MNIST alcanza 32.31±1.95 dB en test, frente a 18.43±2.48 dB cuando se entrena en MNIST y se prueba en FMNIST. Con la CNN de 5 capas, los resultados son 24.91±1.95 dB frente a 15.52±2.47 dB. Con MLP, se obtiene 18.15±1.85 dB frente a 12.00±3.07 dB. En conjunto, FMNIST aporta rasgos más completos que abarcan los dígitos de MNIST, mientras que lo aprendido en dígitos no alcanza la complejidad de la ropa, y aumentar la profundidad o usar MLP agrava el desajuste de dominio.

3.10 Uso del decoder como generador

Se usó el decoder de la CNN de 3 capas como generador tomando los pesos entrenados en MNIST y FMNIST. Se muestrearon vectores latentes $z \sim N(0,1)$ y se inyectaron directamente para producir imágenes sintéticas de 28×28. Las muestras resultantes mostraron formas coherentes con las clases y PSNR promedio cercanos a 30 dB, señal de un espacio latente continuo y útil donde pequeñas variaciones en z producen cambios consistentes. Aunque la calidad visual está por debajo de la de VAE o GAN, el experimento confirma que el decoder puede actuar como generador aproximado y que el autoencoder modela la estructura subyacente de los datos.

4. Conclusiones

En resumen, la CNN de 3 capas es la que mejor funciona: equilibra bien capacidad y generalización y logra los PSNR más altos. La de 5 capas suele suavizar en exceso con el mismo tiempo de entrenamiento y el MLP queda por detrás al no explotar la estructura espacial. MNIST se reconstruye mejor que FMNIST. Subir la dimensión latente ayuda, el dropout empeora y la L1 solo aporta mejoras pequeñas; con estas épocas, LR=1e-3 va mejor en 3 capas y LR=1e-4 es más estable en 5 capas. El Denoising AE añade robustez al ruido con un ligero suavizado, los latentes separan clases en t-SNE, entrenar en FMNIST generaliza bien a MNIST y el decoder puede generar imágenes plausibles desde gausianas latentes.