

Tarea N°2: Redes neuronales

Valentina Paz Campos Olguín

I. RESUMEN

Este proyecto está orientado a la clasificación automática de imágenes usando los conjuntos de datos QuickDraw-10 y QuickDraw-Animals. Un modelo MLP (Multi-Layer Perceptron) fue implementado y evaluado, aplicando PCA para la reducción de dimensionalidad y mejorar el rendimiento y reducir la complejidad del modelo. Distintas arquitecturas MLP fueron testeadas a través de un GridSearch variando la cantidad de capas ocultas, cantidad de neuronas por capa, funciones de activación y funciones de pérdida, con múltiples ejecuciones de entrenamiento por configuración para encontrar la mejor combinación de hiperparámetros.

Los resultados mostraron que combinar PCA con MLPs configurados correctamente, especialmente usando activaciones no lineales como Tanh [2], alcanzaron una buena clasificación en ambos conjuntos de datos. Este estudio demuestra un avance en el conocimiento del aprendizaje supervisado y el ajuste de redes neuronales, mientras se establece una etapa para futuro trabajo en modelos más profundos y optimización automática de los hiperparámetros.

II. INTRODUCCIÓN

La clasificación de imágenes es una tarea fundamental en la visión de computador y aprendizaje automático, con aplicaciones desde el reconocimiento de objetos a etiquetado automatizado y más. Los conjuntos públicos como QuickDraw-10 y QuickDraw-Animals proveen *benchmarks* para desarrollar y probar algoritmos de clasificación en categorías de imágenes que pueden ser desafiantes.

Este informe apunta a implementar un modelo MLP para clasificar imágenes desde estos conjuntos de datos. Para mejorar la eficiencia y reducir la complejidad computacional, el PCA está aplicado como una técnica de reducción de dimensionalidad antes del entrenamiento. Múltiples configuraciones de los modelos con variación de arquitecturas e hiperparámetros son explorados para evaluar su impacto en el rendimiento de clasificación. Las salidas contribuyen a un entendimiento a las consideraciones prácticas en diseño y ajuste de redes neuronales para tareas de clasificación de imágenes.

III. DESARROLLO

III-A. Materiales

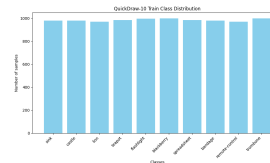
El experimento fue ejecutado en una máquina local con las siguientes especificaciones:

Tabla I: Implementos utilizados

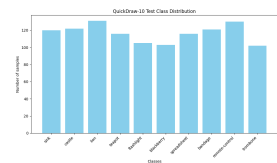
Materiales	Especificación
Modelo	Lenovo Yoga Slim 7
Procesador	AMD Ryzen 5 4500u
Gráfica	AMD Radeon
RAM	8GB
Sistema operativo	Linux
Distribución	Kali
Arquitectura	x86-64
Versión	6.12.25-amd64

III-B. Preprocesamiento

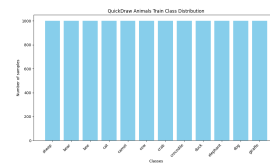
El conjunto de datos de QuickDraw consta de 10 clases, con 9834 imágenes para el entrenamiento y 1166 para la evaluación, mientras que QuickDraw-Animals conta de 12 clases, donde se tienen 12000 imágenes de entrenamiento y 2399 para la evaluación. Además, se empleó la librería “Scikit-learn” que ofrece una gran variedad de algoritmos de ML [1].



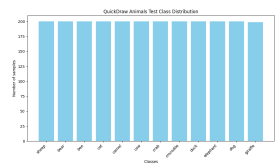
(a) Conjunto de entrenamiento QuickDraw-10



(b) Conjunto de evaluación QuickDraw-10



(a) Conjunto de entrenamiento QuickDraw-Animals



(b) Conjunto de evaluación QuickDraw-Animals

Los histogramas muestran una distribución balanceada entre las clases en ambos conjuntos de datos, con aproximadamente 1000 imágenes por clase en los conjuntos de entrenamiento, 100 imágenes por clase en el conjunto de evaluación en QuickDraw-10 y 200 imágenes por clase en el conjunto de evaluación de QuickDraw-Animals. Eso asegura una representación equitativa para cada clase y contribuye a una evaluación confiable y sin sesgo de los modelos.

Se aplicó reducción de dimensionalidad utilizando PCA, reduciendo cada imagen a 256 componentes, así pudiendo facilitar el procesamiento al reducir la complejidad computacional de las imágenes originales, las cuales contienen 65,536 características.

III-C. Diseño del modelo e implementación

La clasificación de modelos implementados son MLPs con arquitecturas variadas. Las diferentes configuraciones fueron testeadas, variando el número de capas ocultas (entre 2 a 4 capas), el número de neuronas por capa (64, 128, 256 y 512), funciones de activación (tales como ReLU y Tanh), y funciones de pérdida (categorical cross-entropy y MSE). Esto permitió la exploración del impacto en estos hiperparámetros en el rendimiento de clasificación.

Cada MLP fue construido usando capas *fully connected*, teniendo la capa de salida con una cantidad de neuronas igual al número de clases en cada conjunto de datos, es decir, 10 neuronas para el dataset de QuickDraw-10 y 12 neuronas para QuickDraw-Animals. Los modelos fueron compilados con el optimizador SGD, y su entrenamiento tuvo 20 épocas por ejecución, ya que en las ejecuciones del modelo se muestra que continuar con más épocas no hace mejorar la precisión de forma significativa. Cada configuración fue entrenada y evaluada en 5 ejecuciones independientes con diferentes pesos iniciales.

III-D. Estructura de código

Los experimentos utilizaron conjuntos de datos preprocesados con características reducidas con PCA como *inputs* de los modelos MLP. Tanto QuickDraw-10 como QuickDraw-Animals fueron usados para generalizar el modelo en diferentes dominios de imágenes.

Cada configuración de modelo fue entrenado usando *batches* de 32 ejemplos, con un rendimiento evaluado usando métricas como exactitud, precisión, sensibilidad, F1-score, y matrices de confusión. Estas métricas fueron calculadas después de cada ejecución de entrenamiento y fueron guardadas en distintos archivos csv para evitar la pérdida de esta información.

IV. RESULTADOS EXPERIMENTALES

IV-A. Vista general del experimento

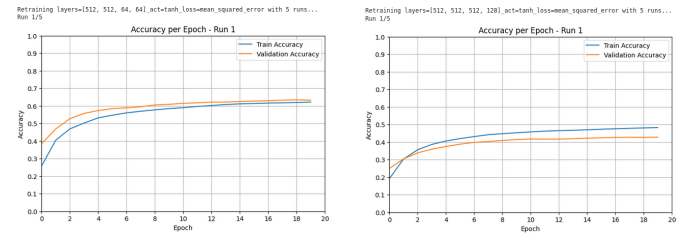
Esta sección presenta los resultados obtenidos de evaluar el top 3 de modelos MLP en 2 datasets: QuickDraw-10 y QuickDraw-Animals. El rendimiento fue medido usando principalmente la precisión de clasificación, tanto de forma general como por clase, además de matrices de confusión para analizar el comportamiento de cada modelo a detalle. Adicionalmente, los tiempos de ejecución y el ajuste de hiperparámetros son discutidos.

IV-B. Justificación número de épocas

Para determinar la cantidad adecuada de épocas para el entrenamiento de los modelos MLP, se analizaron las curvas de precisión tanto en los datos de entrenamiento como de validación a lo largo de las épocas. En las Figuras 3a y 3b se observa que, para los modelos seleccionados, la precisión se estabiliza aproximadamente a partir de la época 15, con incrementos mínimos posteriores hasta la época 20. Esto indica que entrenar más allá de 20 épocas no aporta mejoras significativas en el desempeño y podría aumentar

innecesariamente el tiempo de cómputo y el riesgo de sobreajuste.

Por esta razón, se estableció un límite de 20 épocas para el entrenamiento, buscando un balance adecuado entre eficiencia computacional y calidad del modelo.



(a) Precisión por época para modelo MLP en QuickDraw-10

(b) Precisión por época para modelo MLP en QuickDraw-Animals

Figura 3: Precisión por época de los modelos MLP en ambos datasets

IV-C. Ajuste de hiperparámetros

Aquí se describe el proceso de búsqueda de llos hiperparámetros: cuáles parámetros fueron variados (capas, neuronas, función de activación y de pérdida), el criterio para seleccionar los modelos y más.

A continuación, se detalla cuales fueron los 3 mejores modelos para cada conjunto de datos según la métrica global.

Tabla II: Modelos en QuickDraw-10

Modelo	Capas	Activación	Pérdida
1	[512, 512, 128, 64]	Tanh	MSE
2	[512, 512, 64, 64]	Tanh	MSE
3	[512, 512, 256, 128]	Tanh	MSE

Tabla III: Modelos en QuickDraw-Animals

Modelo	Capas	Activación	Pérdida
1	[512, 512, 512, 128]	Tanh	MSE
2	[512, 512, 512, 64]	Tanh	MSE
3	[512, 512, 512, 512]	Tanh	MSE

IV-D. Métricas de modelos

Analizando la precisión global de los mejores 3 modelos de cada conjunto de datos, logra revelar algunas tendencias importantes que se deben revisar.

Primeramente, a pesar de que los modelos más grandes y con arquitecturas más profundas tienden a tener una ligeramente mejor precisión, las mejoras son usualmente marginales. Por ejemplo, en QD10, el modelo 1 con las capas [512, 512, 128, 64] alcanza la precisión más alta (64.3 %), pero el modelo 3 con una arquitectura más compleja ([512, 512, 256, 128]) rinde un poco peor (63.8 %). Esto sugiere que aumentar la complejidad del modelo podría, a cierto punto, empezar a dejar de tener sentido ya que comienza a disminuir el rendimiento de este.

En segundo lugar, todos los modelos usan la función de activación tanh y una función de pérdida de MSE, los cuales indican que las diferencias de precisión observadas se relacionan más con las variaciones de arquitectura más que los comportamientos de activación.

En tercer lugar, el dataset de QuickDraw-Animals muestra una precisión significativamente menor (43.6 % máximo), el cual se alinea con la gran dificultad de clasificar visualmente clases más complejas. Esto destaca cómo las características del dataset influyen en alcanzar una precisión adecuada a pesar de las arquitecturas similares de los modelos.

Finalmente, considerando las pequeñas diferencias en la precisión junto a los tiempos de entrenamiento y el costo computacional, los modelos más simples con menos parámetros podrían ser preferidos en escenarios prácticos donde la eficiencia es crítica.

Los tiempos de entrenamiento para los mejores 3 modelos MLP son presentados en las Tablas IV y V. Todos los modelos utilizaron la función de activación tanh y MSE como métrica de pérdida.

En el conjunto de datos de QD10, los modelos difieren principalmente en sus configuraciones de capas, siendo el modelo 3 el que tiene capas intermedias de mayor tamaño, y el modelo 2 teniendo las capas de menor tamaño. El tiempo de entrenamiento refleja esta complejidad: El modelo 3 requiere de más tiempo para entrenar, mientras que el modelo 2 entrena ligeramente más rápido debido a la menor cantidad de neuronas en algunas capas.

Similarmente, para QDAnimals, todos los modelos tienen arquitecturas más profundas con 4 capas ocultas con 512 neuronas cada una, siendo el modelo 3 el que tiene una arquitectura más extensa al tener todas sus capas con 512 neuronas. Este aumento de complejidad en el modelo se asocia a duraciones de entrenamiento más largas, como se observa en los tiempos promedios de entrenamiento del modelo 3 comparado a los modelos 1 y 2.

Cabe destacar que hay una clara correlación entre la complejidad del modelo (medido a través de su cantidad de capas y sus tamaños) y el tiempo de entrenamiento. Este intercambio debe ser balanceado con el *accuracy* para seleccionar el despliegue del modelo óptimo.

Tabla IV: Métricas de modelos en QuickDraw-10

Modelo	Accuracy	Tiempo [s]
1	0.6426	21.1
2	0.6392	20.0
3	0.6383	22.5

Tabla V: Métricas de modelos en QuickDraw-Animals

Modelo	Accuracy	Tiempo [s]
1	0.4380	30.56
2	0.4369	28.45
3	0.4362	32.14

Las tablas muestran la mediana de las precisiones para 3 diferentes modelos MLP evaluados.

Para QuickDraw-10, el Modelo 2 alcanzó la mayor precisión mediana de aproximadamente 0.639. Los Modelos 0 y 1, con arquitecturas ligeramente diferentes, muestran precisiones similares pero un poco más bajas, alrededor de 0.632. Esto sugiere que aumentar el número de neuronas en las capas más profundas puede contribuir positivamente a la precisión en este conjunto de datos.

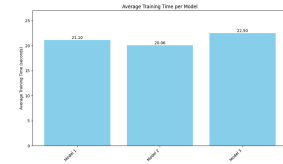
En contraste, para QuickDraw-Animals, los valores de precisión mediana son generalmente más bajos, siendo el Modelo 2 ([512, 512, 512, 512]) el que mejor desempeño mostró con aproximadamente 0.427. Los Modelos 0 y 1 tienen precisiones marginalmente más bajas, cerca de 0.42 y 0.41, respectivamente. La precisión más baja en general podría reflejar la mayor complejidad o variabilidad en los dibujos de animales en comparación con las categorías más simples de QuickDraw-10.

Tabla VI: Mediana de precisión en QuickDraw-10

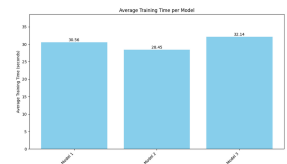
Modelo	Capas	Mediana Precisión
2	[512, 512, 256, 128]	0.6388
1	[512, 512, 64, 64]	0.6322
0	[512, 512, 128, 64]	0.6323

Tabla VII: Mediana de precisión en QuickDraw-Animals

Modelo	Capas	Mediana Precisión
2	[512, 512, 512, 512]	0.4266
0	[512, 512, 512, 64]	0.4234
1	[512, 512, 512, 128]	0.4183



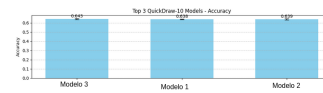
(a) Tiempos promedio QD10



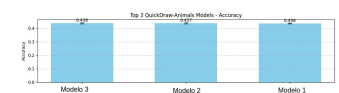
(b) Tiempos promedio Animals

IV-E. Precisión de los modelos

Las figuras 5a y 5b muestran la precisión general de los mejores 3 modelos MLP en el conjunto de datos de QuickDraw-10 y QuickDraw-Animals, respectivamente. Los valores de precisión de los modelos de QD10 varían aproximadamente entre 63.8 % y 64.3 %, mientras que para QDAnimals la precisión es más baja, teniendo un rango entre los valores 43.6 % y 43.8 %. Estos resultados indican un nivel moderado en el rendimiento de la clasificación, siendo consistente con la complejidad y variabilidad inherente de los *datasets*.



(a) Precisión de top 3 modelos QuickDraw-10



(b) Precisión de top 3 modelos QuickDraw-Animals

Figura 5: Comparación de precisión de los top 3 modelos en los datasets QuickDraw-10 y QuickDraw-Animals

IV-F. Precisión de cada modelo por clase

Las figuras 6 y 7 presentan la precisión alcanzada para cada modelo en las clases individuales. Este análisis revela

variabilidad en la dificultad de clasificación entre las clases. Por ejemplo, en QuickDraw-10, las clases como “remote-control” y “sink” alcanzaron una mayor precisión, mientras que en clases como “flashlight” y “trombone” mostraron un menor rendimiento. Similarmente, en QuickDraw-Animals, las clases como “giraffe” y “crocodile” tuvieron mejor precisión en comparación con clases más desafiantes como “cat” y “duck”.

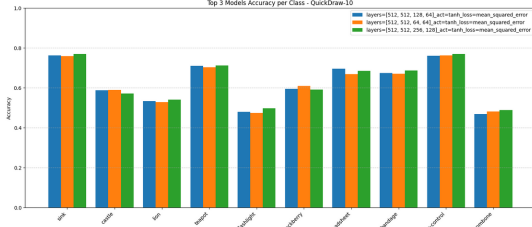


Figura 6: Precisión por clase de modelos QuickDraw-10

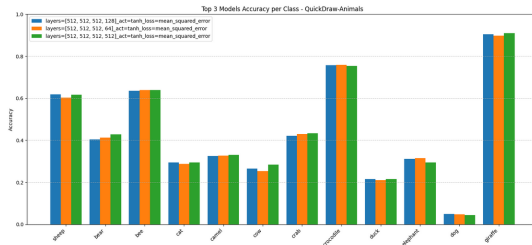
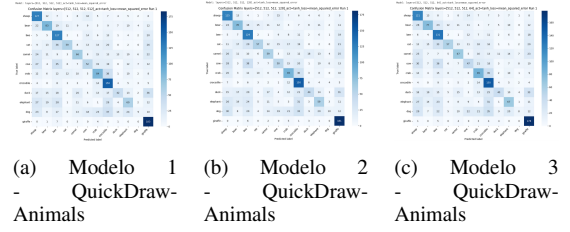
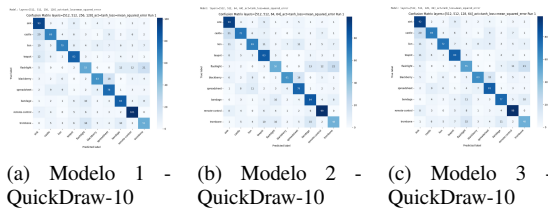


Figura 7: Precisión por clase de modelos QuickDraw-Animals

IV-G. Matrices de confusión

Para entender el comportamiento de clasificación de cada modelo, las matrices de confusión son presentadas para cada top 3 modelos por conjunto de datos. Estas matrices destacan qué tan seguido los ejemplos de cada verdadera clase son predecidas correctamente o son mal clasificadas en otras clases.

Las matrices de confusión para los modelos QD10 indican robustez consistente en clases como “sink” y “remote-control”, con cierta confusión observada entre las clases como “lion” y “teapot”. Para QDAnimals, ciertas clases como “giraffe” y “crocodile” fueron bien reconocidas, mientras que en las clases como “cat”, “bear” y “duck” tienden a ser confundidas de forma más frecuente con otras, indicando áreas de potencial mejora.



V. CONCLUSIONES

En este trabajo, se evaluó el rendimiento de los mejores 3 modelos MLP en los conjuntos de datos de QuickDraw-10 y QuickDraw-Animals. Los resultados mostraron que mientras más profundos y complejas son las arquitecturas tienden a alcanzar una ligera mejor precisión general, las mejoras son en general marginales. Esto indica un punto de reducción donde aumentar la complejidad del modelo ya no demuestra una ganancia significativa y hasta puede empeorar el rendimiento.

Todos los modelos usaron la función de activación tanh y función de pérdida MSE, así que las diferencias en precisión son principalmente atribuidas a las variaciones de arquitectura más que los cambios en los objetivos de entrenamiento. Adicionalmente, el conjunto de datos de QuickDraw-Animals demostró ser más desafiante, con una precisión notablemente menor comparado a QuickDraw-10. Esto destaca la fuerte influencia en la complejidad del dataset para un rendimiento alcanzable.

El tiempo de entrenamiento se correlacionó positivamente con la complejidad del modelo, esto significa que mientras más grande es el modelo, más tiempo toma en entrenarse. Asimismo, en escenarios prácticos donde los recursos computacionales o latencia son limitados, los modelos más simples que ofrecen una precisión comparable podrían preferirse más.

Para un trabajo futuro, explorar arquitecturas más avanzadas como redes neuronales convolucionales (CNNs) o modelos basados en transformers podrían mejorar el rendimiento de la clasificación a través de la captura visual de las características. Experimentar con más funciones de pérdida, funciones de activación, y técnicas de *data augmentation* podrían también ayudar a mejorar la robustez del modelo. Además, incorporar un ajuste de hiperparámetros aún más exhaustiva podría ayudar a balancear la complejidad del modelo con su habilidad de generalización.

En conclusión, este estudio demuestra las compensaciones entre la complejidad del modelo, precisión, y tiempos de entrenamiento cuando se aplican MLPs a la clasificación de *sketches*. Los resultados proveen una base para una futura investigación, apuntando a mejorar el rendimiento en estos datasets mientras se mantiene la eficiencia computacional.

REFERENCIAS

- [1] Pedregosa, F., Varoquaux, G., Gramfort, A., Michel, V., Thirion, B., Grisel, O., Blondel, M., Prettenhofer, P., Weiss, R., Dubourg, V., & otros. (2011). Scikit-learn: Machine learning in Python. *Journal of Machine Learning Research*, 12, 2825-2830. Recuperado de <https://www.jmlr.org/papers/volume12/pedregosa11a/pedregosa11a.pdf>
- [2] Glorot, X., & Bengio, Y. (2010). Understanding the difficulty of training deep feedforward neural networks. *Proceedings of the Thirteenth International Conference on Artificial Intelligence and Statistics*, 249-256.