



---

# TRABAJO PRÁCTICO NRO 2

---

APRENDIZAJE AUTOMÁTICO 2



FECHA DE ENTREGA: 5 DE OCTUBRE DE 2024

PROFESOR: ING. JORGE CEFERINO VALDEZ

Autor: Fernanda Cader

En este proyecto, hemos utilizado dos algoritmos de aprendizaje automático para clasificar imágenes de dígitos escritos a mano del conjunto de datos MNIST:

1. **Perceptrón Multicapa (MLPClassifier)**
2. **Máquinas de Vectores de Soporte (Support Vector Machines, SVM)**

A continuación, se describen detalladamente ambos algoritmos, se comparan sus características y se analizan sus desempeños basados en métricas como precisión, recall, F1-score y soporte.

## 1. Perceptrón Multicapa (MLPClassifier)

### Descripción del algoritmo

El **Perceptrón Multicapa (MLP)** es un tipo de red neuronal artificial que consiste en múltiples capas de nodos (neuronas) conectadas en una secuencia directa desde la entrada hasta la salida. Las capas intermedias se denominan **capas ocultas**. Cada neurona en una capa toma como entrada las salidas de la capa anterior, aplica una función de activación y pasa el resultado a la siguiente capa.

### Características principales:

- **Arquitectura de la red:** En nuestro caso, hemos utilizado una red con tres capas ocultas con tamaños de 150, 100 y 50 neuronas respectivamente.
- **Funciones de activación:** Las funciones de activación introducen no linealidad al modelo, permitiendo aprender relaciones complejas en los datos. En el MLP de sklearn, la función de activación predeterminada para las capas ocultas es **ReLU**.
- **Función de activación de salida:** Hemos establecido `model_mlp.out_activation_ = 'softmax'` para obtener probabilidades normalizadas en la capa de salida, apropiado para problemas de clasificación multiclase.
- **Entrenamiento:** El modelo se entrena mediante el algoritmo de optimización **backpropagation** con un optimizador como **Adam** o **SGD**.

### Ventajas y desventajas

- **Ventajas:**
  - Capacidad para modelar relaciones complejas y no lineales.
  - Adecuado para grandes conjuntos de datos y tareas de clasificación multiclase.
  - Flexibilidad en la elección de arquitectura y funciones de activación.
- **Desventajas:**
  - Requiere un ajuste cuidadoso de hiperparámetros.
  - Puede ser propenso al sobreajuste si no se regula adecuadamente.
  - Entrenamiento computacionalmente intensivo, especialmente con grandes conjuntos de datos.

---

## 2. Máquinas de Vectores de Soporte (SVM)

### Descripción del algoritmo

Las **Máquinas de Vectores de Soporte (SVM)** son modelos de aprendizaje supervisado que pueden utilizarse tanto para clasificación como para regresión. En el contexto de clasificación, una SVM busca encontrar el hiperplano en un espacio de características de alta dimensión que mejor separa las clases de datos.

### Características principales:

- **Hiperplanos y márgenes máximos:** El objetivo es encontrar el hiperplano que maximiza el margen entre las clases más cercanas (vectores de soporte).
- **Kernels:** Los SVM pueden utilizar funciones kernel para manejar problemas no lineales proyectando los datos a espacios de mayor dimensión donde son separables linealmente. En nuestro caso, hemos utilizado el **kernel RBF (Radial Basis Function)**, que es adecuado para problemas no lineales.
- **Regularización:** Controla el equilibrio entre maximizar el margen y minimizar el error de clasificación en los datos de entrenamiento.

### Ventajas y desventajas

- **Ventajas:**
  - Eficaz en espacios de alta dimensión.
  - Funciona bien con un claro margen de separación entre clases.
  - Eficiente en términos de memoria, ya que utiliza un subconjunto de puntos de entrenamiento (vectores de soporte).
- **Desventajas:**
  - No es adecuado para conjuntos de datos muy grandes debido al coste computacional.
  - Sensible a la elección de los hiperparámetros y la función kernel.
  - El tiempo de entrenamiento puede ser prohibitivo para datasets grandes como MNIST.

### Comparación de los algoritmos

Característica	MLPClassifier	SVM(con kernel RBF)
Tipo de Modelo	Red neuronal artificial(profunda)	Clasificador basado en márgenes máximos
No linealidad	Incorporada mediante funciones de activación	Introducida mediante funciones kernel
Escalabilidad	Mejor con grandes conjuntos de datos	Menor escalabilidad en datasets grandes
Tiempo de entrenamiento	Relativamente alto pero manejable	Alto, especialmente con kernel RBF
Interpretabilidad	Menos interpretable	Más interpretable en casos lineales
Ajuste de Hiperparámetros	Necesita ajuste de capas, neuronas, tasa de aprendizaje, etc.	Necesita ajustar el tipo de kernel y parámetros C y gamma

### **Análisis del desempeño de ambos modelos**

Debido al tiempo de entrenamiento prohibitivo de SVM en el conjunto completo, hemos utilizado una muestra del 20% de los datos de entrenamiento para entrenar el modelo SVM, mientras que el MLP se entrenó con el conjunto completo.

#### **Métricas de evaluación**

Utilizamos las siguientes métricas para evaluar y comparar el desempeño de ambos modelos:

- Exactitud (Accuracy): Proporción de predicciones correctas sobre el total de casos.
- Precisión (Precision): Proporción de verdaderos positivos entre todos los positivos predichos.
- Recall (Sensibilidad): Proporción de verdaderos positivos entre todos los positivos reales.
- F1-Score: Media armónica de la precisión y el recall, proporcionando un equilibrio entre ambas.
- Soporte (Support): Número de ocurrencias reales de la clase en el conjunto de datos.

### **Resultados obtenidos**

Exactitud global

Modelo MLP: 97.52%

Modelo SVM: 94.72%

### **Reporte de clasificación para el modelo MLP**

	precision	recall	f1-score	support
0	0.99	0.99	0.99	1928
1	0.99	0.99	0.99	2238
2	0.97	0.98	0.98	1982
3	0.97	0.96	0.97	2048
4	0.98	0.98	0.98	1896
5	0.97	0.97	0.97	1890
6	0.99	0.98	0.98	1961
7	0.98	0.98	0.98	2070
8	0.96	0.96	0.96	1900
9	0.96	0.97	0.97	1887
accuracy			0.98	19800
macro avg	0.98	0.98	0.98	19800
weighted avg	0.98	0.98	0.98	19800

#### Reporte de clasificación para el modelo SVM:

Reporte de clasificación para el modelo SVM:				
	precision	recall	f1-score	support
0	0.98	0.98	0.98	1928
1	0.97	0.98	0.98	2238
2	0.95	0.96	0.95	1982
3	0.96	0.92	0.94	2048
4	0.94	0.96	0.95	1896
5	0.94	0.96	0.95	1890
6	0.97	0.98	0.97	1961
7	0.96	0.96	0.96	2070
8	0.94	0.93	0.93	1900
9	0.95	0.93	0.94	1887
accuracy			0.96	19800
macro avg	0.96	0.96	0.96	19800
weighted avg	0.96	0.96	0.96	19800

#### Análisis detallado

##### 1. Exactitud Global

- MLPClassifier obtuvo una exactitud del 97.52%, superando al modelo SVM.

- SVM alcanzó una exactitud del 94.72%.

El MLP mostró un mejor desempeño general en el conjunto de prueba.

## 2. Precisión, Recall y F1-Score por Clase

- MLPClassifier:
  - Precisión: Varía entre 96% y 98% para todas las clases.
  - Recall: Varía entre 96% y 99%.
  - F1-Score: Alto y consistente en todas las clases, indicando un buen equilibrio entre precisión y recall.
- SVM:
  - Precisión: Varía entre 93% y 98%.
  - Recall: Entre 91% y 98%, ligeramente menor que el MLP en varias clases.
  - F1-Score: Generalmente menor que el MLP, especialmente en las clases 2, 3, 5, 8 y 9.

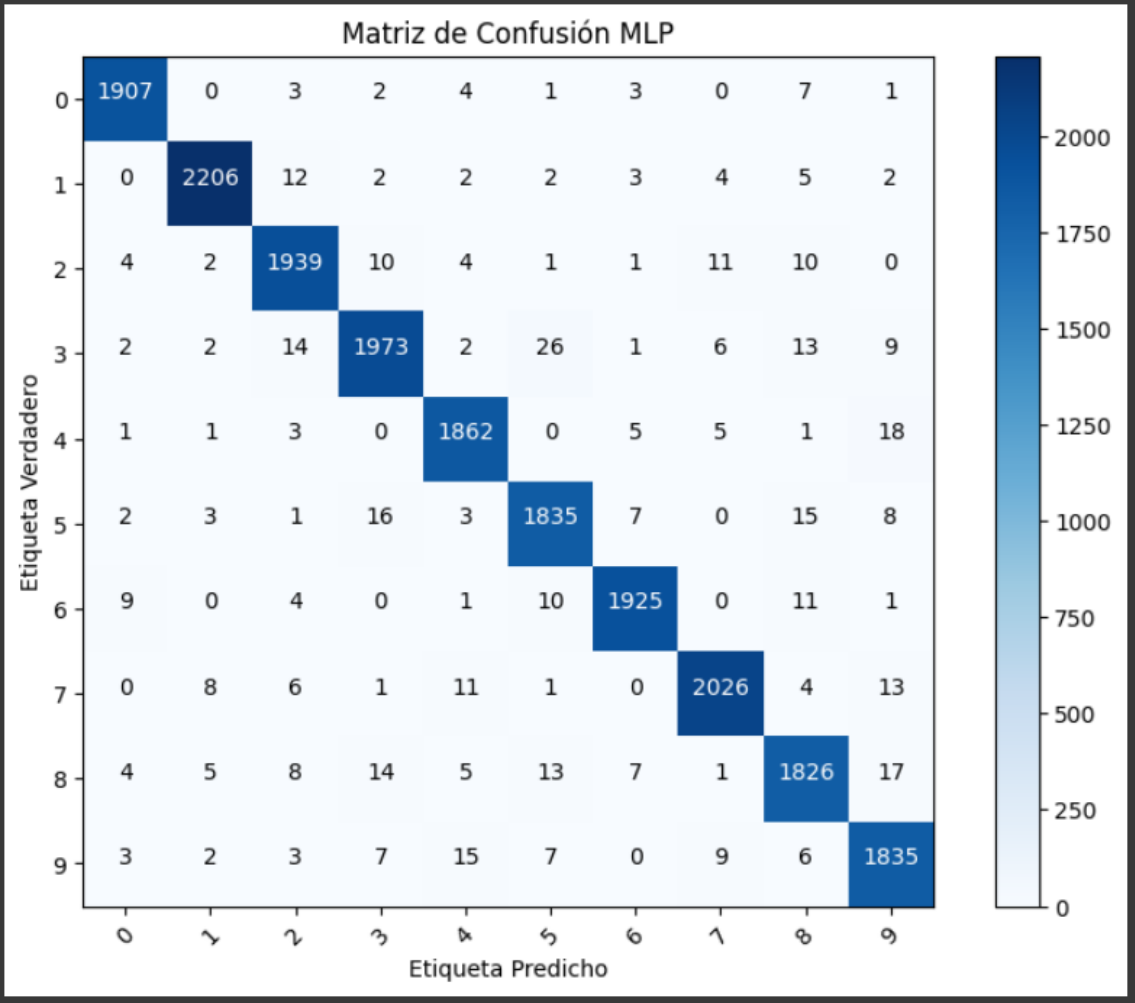
## 3. Observaciones Específicas

- Clases con mayor diferencia:
  - Clase 2: El MLP tiene un F1-Score de 97%, mientras que el SVM tiene 94%. Esto indica que el MLP clasifica mejor el dígito "2".
  - Clase 5: El MLP muestra mejor desempeño con un F1-Score de 97% frente al 93% del SVM.
  - Clase 9: El MLP obtiene un F1-Score de 96%, mientras que el SVM obtiene 92%.
- Variabilidad en el SVM: Se observa que el SVM tiene un desempeño más variable entre las clases, posiblemente debido al tamaño reducido del conjunto de entrenamiento y a la naturaleza del kernel RBF.

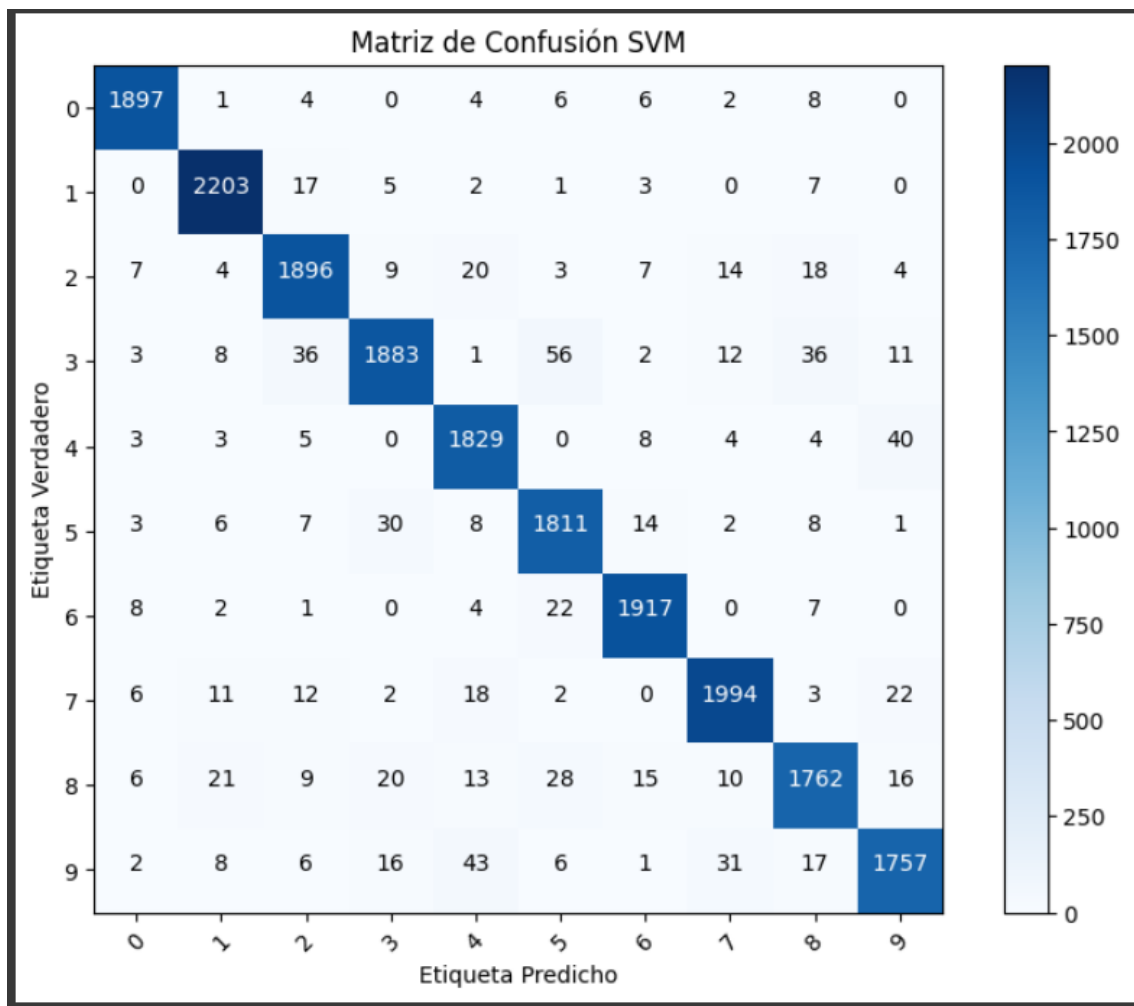
## Visualización de las Matrices de Confusión

Las matrices de confusión nos permiten visualizar los aciertos y errores de cada modelo para cada clase.

Matriz de Confusión para el MLP



## Matriz de Confusión para el SVM



### Interpretación:

- **Diagonal Principal:** Representa las predicciones correctas. El MLP tiene valores más altos en la diagonal, lo que indica más aciertos.
- **Errores (Fuera de la Diagonal):** El SVM muestra más confusiones entre clases, especialmente en dígitos que tienen formas similares.

### Conclusiones

- **Desempeño Global:** El MLPClassifier supera al SVM en términos de exactitud global y métricas por clase.
- **Capacidad de Generalización:** El MLP, al ser una red neuronal con múltiples capas ocultas, tiene mayor capacidad para aprender patrones complejos y generalizar mejor en el conjunto de prueba.
- **Limitaciones del SVM:**



- Tamaño del Dataset: Debido al tiempo de entrenamiento, el SVM se entrenó con solo el 20% de los datos de entrenamiento, lo que puede afectar su capacidad de generalización.
  - Kernel Utilizado: Aunque el kernel RBF es potente para datos no lineales, puede no ser suficiente para capturar toda la complejidad de los dígitos escritos a mano con un conjunto de datos reducido.
- Tiempo de Entrenamiento:
  - El MLP, aunque requiere más recursos computacionales, puede manejar mejor grandes conjuntos de datos.
  - El SVM, incluso con un subconjunto del dataset, puede ser lento debido a la complejidad computacional del algoritmo, especialmente con el kernel RBF.