

# Análisis comparativo de Redes Neuronales (MLP vs SRNN vs LSTM vs GRU)

## Descripción detallada de la actividad

En esta actividad, el estudiante desarrollará un estudio experimental completo orientado a la comparación del desempeño de distintos modelos de redes neuronales aplicados a la predicción de series temporales financieras, específicamente precios históricos de acciones.

El propósito principal es analizar de manera rigurosa las diferencias entre:

- Redes neuronales artificiales simples (MLP – *Multilayer Perceptron*),
- Redes neuronales simples (SRNN – *Red neuronal recurrente simple*)
- Redes neuronales recurrentes del tipo **LSTM** (*Long Short-Term Memory*),
- Redes neuronales recurrentes del tipo **GRU** (*Gated Recurrent Unit*),

evaluando su capacidad para modelar dependencias temporales, estabilidad durante el entrenamiento, precisión predictiva y sensibilidad a la optimización de hiperparámetros.

## Alcance técnico de la actividad

El estudiante deberá realizar las siguientes etapas:

### 1. Adquisición de datos reales

Seleccionar y descargar un conjunto de datos reales de precios históricos de acciones desde una fuente pública confiable (por ejemplo: Yahoo Finance, Alpha Vantage, Kaggle u otra API financiera), documentando claramente:

- Empresa o activo financiero seleccionado (ticker),
- Intervalo temporal,
- Frecuencia de muestreo (diaria, semanal, etc.),
- Variables utilizadas (Open, High, Low, Close, Volume, etc.),
- Fuente de los datos.

### 2. Preprocesamiento y estructuración

Aplicar técnicas adecuadas para preparar los datos para modelos de aprendizaje profundo, incluyendo:

- Limpieza de datos y manejo de valores faltantes,
- Normalización o estandarización,
- Construcción de ventanas temporales (*sliding windows*),
- División en conjuntos de entrenamiento, validación y prueba,
- Análisis exploratorio básico de la serie temporal.

### 3. Diseño e implementación de modelos

Construir e implementar tres arquitecturas independientes:

1. Red neuronal feedforward (MLP),
2. Red neuronal Recurrente Simple
3. Red neuronal LSTM,
4. Red neuronal GRU,

definiendo explícitamente:

- Número de capas,
- Número de neuronas por capa,
- Funciones de activación,
- Estrategias de regularización (Dropout, Early stopping),
- Función de pérdida y optimizador.

### 4. Optimización de hiperparámetros

Aplicar un procedimiento sistemático de optimización para cada modelo, considerando parámetros como:

- Learning rate,
- Batch size,
- Número de épocas,
- Tamaño de la ventana temporal,
- Número de neuronas,
- Profundidad del modelo,

empleando técnicas como búsqueda en rejilla (*Grid Search*), búsqueda aleatoria (*Random Search*), justificando la metodología seleccionada.

### 5. Evaluación experimental

Evaluar el desempeño de los modelos utilizando métricas cuantitativas apropiadas para regresión en series temporales, tales como:

- MAE (Mean Absolute Error),
- RMSE (Root Mean Square Error),
- MAPE (Mean Absolute Percentage Error),

- $R^2$  (coeficiente de determinación).

Adicionalmente, se recomienda:

- Analizar curvas de entrenamiento y validación,
- Estudiar posibles fenómenos de sobreajuste o subajuste,
- Comparar tiempos de entrenamiento y complejidad computacional.

## **6. Análisis comparativo y discusión**

Realizar un análisis crítico de los resultados obtenidos, abordando:

- Diferencias de desempeño entre MLP, SRNN, LSTM y GRU,
- Impacto de la optimización de hiperparámetros,
- Estabilidad de los modelos,
- Capacidad de generalización,
- Ventajas y limitaciones prácticas de cada arquitectura en el contexto financiero.

## **7. Entregables**

El estudiante deberá entregar:

- Código fuente completamente funcional y documentado,
- Informe técnico estructurado,
- Tablas comparativas de métricas,
- Gráficas de predicción vs valores reales,
- Descripción del proceso experimental,
- Conclusiones técnicas fundamentadas.

## **Competencias desarrolladas**

Esta actividad fortalece:

- Modelado predictivo con Deep Learning,
- Análisis de series temporales,
- Diseño experimental en Machine Learning,
- Optimización de modelos,
- Evaluación cuantitativa,
- Pensamiento crítico y comunicación técnica.

## **Criterios de evaluación:**

### **1. Obtención y descripción de la base de datos (10 pts)**

Excelente: Dataset confiable, documentado y justificado.

Bueno: Dataset correcto con descripción básica.

Aceptable: Dataset válido con descripción incompleta.

Insuficiente: Dataset poco claro.

Nulo: No presenta dataset.

## **2. Preprocesamiento y preparación de datos (15 pts)**

Excelente: Limpieza, normalización, ventanas temporales y partición correcta.

Bueno: Preprocesamiento adecuado.

Aceptable: Básico.

Insuficiente: Deficiente.

Nulo: No realizado.

## **3. Implementación de modelos (20 pts)**

Excelente: Implementa MLP, LSTM y GRU correctamente.

Bueno: Implementación correcta con mínima justificación.

Aceptable: Solo dos modelos.

Insuficiente: Un modelo.

Nulo: Ninguno.

## **4. Optimización de hiperparámetros (15 pts)**

Excelente: Uso de métodos sistemáticos (Grid, Random).

Bueno: Optimización parcial.

Aceptable: Ajuste manual limitado.

Insuficiente: Muy superficial.

Nulo: No realizado.

## **5. Evaluación con métricas (15 pts)**

Excelente: Varias métricas (RMSE, MAE, MAPE,  $R^2$ ).

Bueno: Métricas principales.

Aceptable: Una métrica.

Insuficiente: Incorrectas.

Nulo: No evaluado.

## **6. Análisis comparativo (15 pts)**

Excelente: Interpretación profunda y crítica.

Bueno: Comparación clara.

Aceptable: Superficial.

Insuficiente: Débil.

Nulo: No presenta.

## **7. Presentación técnica y reproducibilidad (10 pts)**

Excelente: Código organizado y reproducible.

Bueno: Claro.

Aceptable: Básico.

Insuficiente: Desordenado.

Nulo: No entregado.

**Escala de calificación:**

90–100 Excelente

80–89 Muy bueno

70–79 Bueno

60–69 Aceptable

<60 Insuficiente