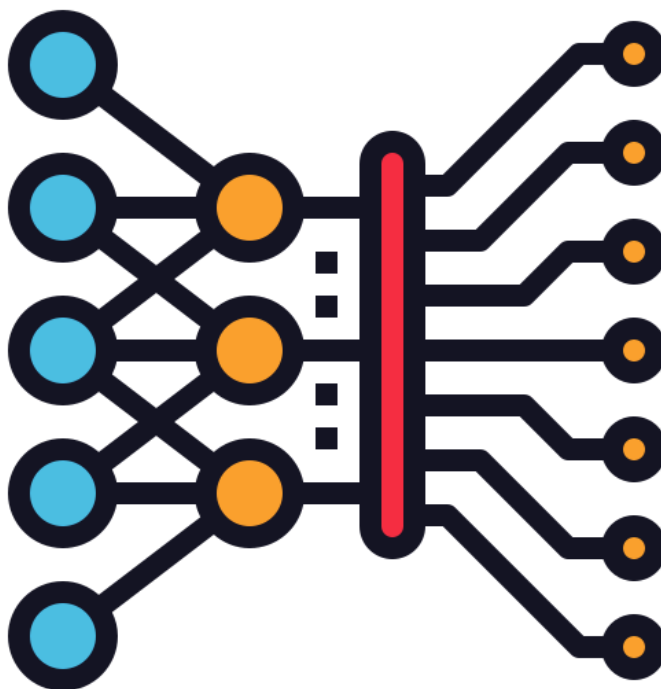




UNR Universidad
Nacional de Rosario

Trabajo práctico Aprendizaje Automático correcciones

Tecnicatura Universitaria en Inteligencia Artificial



Docentes

- Joel Spak
- Agustín Almada
- Bruno Cocitto

Alumnos

- Peroni Antonio
- Raffaeli Taiel



Modelo utilizado

Se utilizó el modelo de clasificación de redes neuronales con los hiperparametros óptimos obtenidos con optuna.

Arquitectura de la red

La red neuronal consta de dos capas ocultas, una de 8 neuronas y otra de 33 neuronas , y una capa de salida con una neurona.

- capas ocultas:
 - Funcion de activacion ReLu:
ReLU es una función de activación que se aplica a la salida de cada neurona para introducir no linealidades en el modelo su función es $f(x)=\max(0,x)$. Esto significa que si la entrada x es positiva, la salida es x ; si es negativa, la salida es cero.
- Capa de salida:
 - Función de activación sigmoidea:
La capa de salida tiene 1 neurona con una activación sigmoide, comúnmente utilizada en problemas de clasificación binaria.

Compilación del modelo

Función de pérdida

En la compilación del modelo se utiliza entropía binaria cruzada, es común para problemas de clasificación binaria. Mide la discrepancia entre las probabilidades predichas y las etiquetas reales.

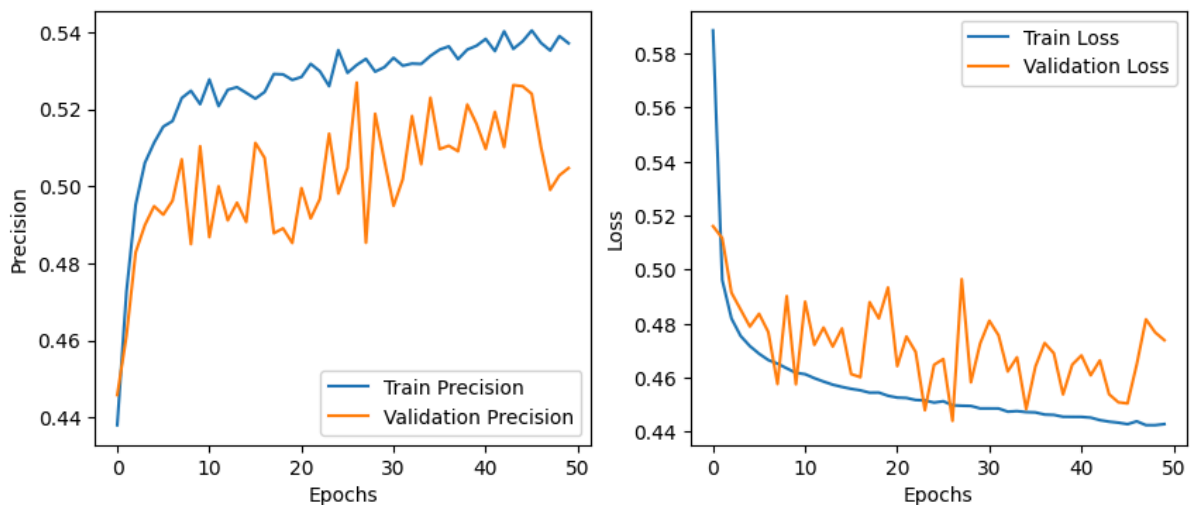
Función de optimización

Como función de optimización se utiliza la función adam que combina la idea del momento en el Gradiente Descendente con la adaptación de la tasa de aprendizaje en RMSProp. Utiliza dos momentos, uno para el gradiente y otro para el cuadrado del gradiente, adaptando la tasa de aprendizaje para cada parámetro haciéndolo un algoritmo adaptativo y robusto.

Resultados obtenidos

Analizamos la evolución de la pérdida a lo largo de las épocas tanto en el conjunto de entrenamiento como en el de validación para evaluar si nuestro modelo está experimentando sobreajuste.

Además, seguimos la misma metodología para medir la métrica de precisión y observar cómo varía entre los conjuntos de entrenamiento y validación .



El modelo tiene buena capacidad para generalizar datos no vistos ya que entre el entrenamiento y validación no hay una diferencia significativa.

Métricas de clasificación de clases con umbral estándar.

Se obtuvieron con el umbral estándar (mayor o igual al 50% probabilidades de pertenecer a una clase) las siguientes métricas de clasificación:

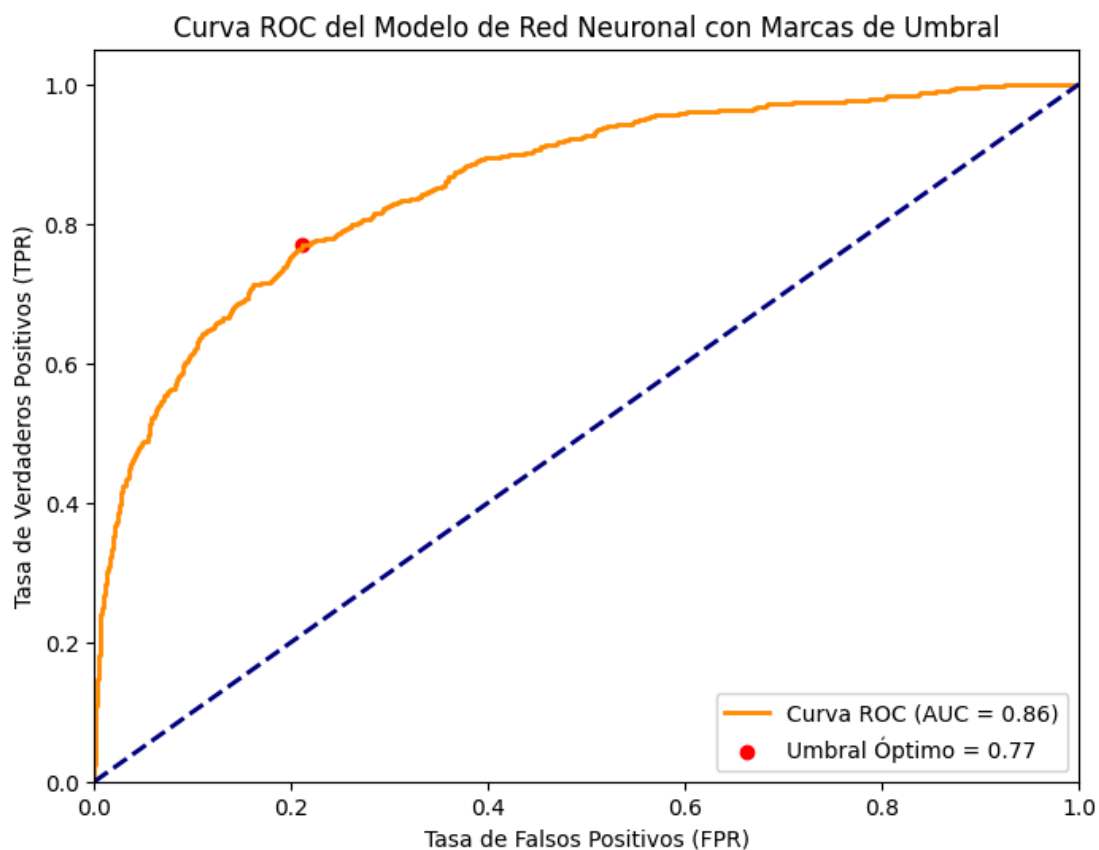
	precision	recall	f1-score	support
0	0.92	0.78	0.84	2351
1	0.50	0.78	0.61	689
accuracy		0.78	0.78	3040

Habiendo considerado el problema planteado, llegamos a la conclusión de que la predicción en la que hay que tener menor cantidad de falsos positivos y mayor cantidad de reales positivos es en la clase 0 (Día que no llueve). Ya que suponemos

que la organización de eventos al aire libre sufre más consecuencias cuando el pronóstico de que no va a llover es falso.

Con el umbral estándar, el modelo capta el 78% de los días en los que no llueve y el 92% de esas predicciones son verdaderos positivos. Esto puede mejorar significativamente ya que podemos encontrar un umbral óptimo que mejore estas métricas y ayude a nuestro modelo a llegar al objetivo de predecir con una mayor cantidad de reales positivos de la clase 0.

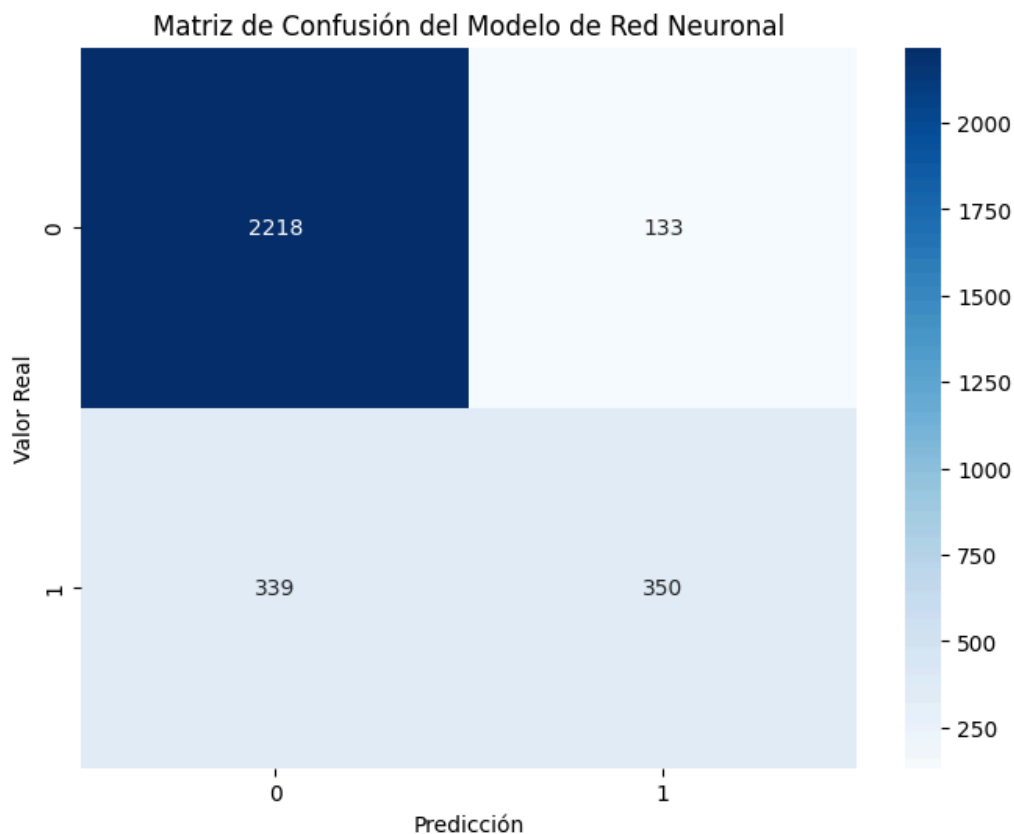
Métricas de clasificación de clases con umbral óptimo.



Luego de realizar el gráfico de curva ROC, el umbral más óptimo encontrado es de 77%, lo cual significa que las instancias de datos que tengan menos probabilidades serán consideradas de la clase 1 (días que llueve).

	precision	recall	f1-score	support	accuracy
0	0.87	0.94	0.90	2351	0.84
1	0.72	0.51	0.60	689	

En este caso con el umbral óptimo obtenido, mejora mucho el recall de la clase 0 llegando a captar el 94% de los días en los que no llueve, siendo el 87% reales positivos. Si se continúa aumentando el umbral aumenta el recall pero disminuye la precisión de la clase 0, esto no es conveniente ya que nuestra intención es tratar de predecir correctamente la mayor cantidad de reales positivos.



Por último realizamos un gráfico de matriz de confusión para así poder visualizar de forma más sencilla las clasificaciones de nuestro modelo. Consideramos que es óptimo el resultado luego de aplicar el nuevo umbral para la clasificación de clases.

Aplicación en streamlit

