Caso de Estudio 01

Predicción de Éxito en Campañas de Telemercadeo Bancario con Redes Neuronales Profundas

Contexto

Una institución financiera ha llevado a cabo múltiples campañas de telemercadeo directo para ofrecer depósitos a plazo a sus clientes. El objetivo es predecir si un cliente suscribirá ('sí') o no ('no') un depósito a plazo. Predecir con alta precisión el resultado de la llamada puede optimizar significativamente los esfuerzos de la campaña, permitiendo al banco enfocar sus recursos en los clientes con mayor probabilidad de conversión y, al mismo tiempo, minimizar la fatiga del cliente.
Su misión es diseñar, entrenar y evaluar un modelo de red neuronal profunda (Deep Neural Network - DNN) capaz de realizar esta predicción. Este caso de estudio te guiará a través del proceso completo, desde la exploración de datos hasta la interpretación y comunicación de resultados a un nivel profesional.

Descripción de los datos

El conjunto de datos contiene información sobre campañas de marketing directo de una institución bancaria portuguesa. Las campañas se basaron en llamadas telefónicas. A menudo, se requería más de un contacto con el mismo cliente para saber si este se suscribiría al producto (depósito a plazo).
El conjunto de datos contiene 45211 entradas y 16 variables de entrada (más la variable de salida), que se pueden clasificar en:

a) **Datos del cliente**: edad, trabajo, estado civil, educación y si tiene préstamos.

b) **Datos del último contacto**: tipo de comunicación, duración de la llamada y día de la semana.

c) **Otros datos de la campaña**: número de contactos realizados durante la campaña actual y el resultado de la campaña de marketing anterior.

d) **Variable de salida**: si el cliente ha suscrito o no un depósito a plazo.

Instrucciones y Criterios de Evaluación

Este caso de estudio se basa en el problema de optimizar las campañas de telemercadeo de una institución financiera. El objetivo es analizar, diseñar, entrenar y evaluar un modelo de red neuronal profunda capaz de predecir la probabilidad de que un cliente acepte la oferta.
A continuación, se presenta una serie de **preguntas orientadoras** diseñadas para guiar su análisis y justificar sus decisiones a lo largo del proyecto. Si bien las respuestas a estas preguntas son fundamentales para demostrar su comprensión conceptual, la calificación final del caso de estudio se evaluará sobre la base del **mejor modelo clasificador** que logre desarrollar, justificar y presentar. Se valorará tanto el rendimiento cuantitativo del modelo (medido con métricas apropiadas) como la calidad de la argumentación que respalde sus decisiones de arquitectura y preprocesamiento.

0.1 Comprensión del Problema y Análisis de Datos (EDA)

a) **Objetivo de Negocio:** ¿Por qué es valioso para el banco utilizar un modelo predictivo en lugar de contactar a todos los clientes? Explique en términos de ROI (Retorno de Inversión) y experiencia del cliente.

b) **Desbalance de Clases:** El análisis exploratorio revela un fuerte desbalance en la variable objetivo (`y`). Cuantifique esta proporción. ¿Qué implicaciones tiene este desbalance para el entrenamiento y la evaluación del modelo?

c) **Métrica de Exactitud (*Accuracy*):** ¿Por qué la exactitud no es una métrica de evaluación adecuada para este problema? Proponga un escenario hipotético donde un modelo inútil podría alcanzar una alta exactitud.

d) **Visualización de Datos:** A partir de los gráficos de tasa de conversión por `job` y `month`, ¿qué dos perfiles de cliente o momentos del año parecen ser los más prometedores para la campaña de marketing?

1 Preprocesamiento y Preparación de Datos

a) **Codificación Categórica:** El notebook utiliza `OneHotEncoder` para las variables predictoras categóricas. ¿Por qué se prefiere esta técnica sobre `LabelEncoder`? ¿Qué problema potencial introduce `LabelEncoder` en este contexto?

b) **Escalado de Características:** Explique la justificación de usar `StandardScaler` en las variables numéricas. ¿Cómo ayuda este paso al proceso de convergencia del optimizador de la red neuronal (ej. descenso de gradiente)?

c) **Datos Faltantes:** La variable `poutcome` tiene más de un 80% de valores faltantes. ¿Sería una buena estrategia eliminar estas filas? Justifique su respuesta y proponga cómo debe ser tratado este ”valor faltante”.

d) **Fuga de Datos (*Data Leakage*):** La variable `duration` (duración de la llamada) es un predictor muy potente. ¿Por qué su inclusión en un modelo que busca predecir el éxito *antes* de realizar la llamada es un ejemplo de fuga de datos?

2 Diseño y Arquitectura del Modelo

a) **Función de Activación en Capas Ocultas:** El modelo base utiliza la función de activación ReLU (*Rectified Linear Unit*). Describa matemáticamente esta función y explique una de sus principales ventajas sobre funciones más antiguas como la sigmoide en capas ocultas.

b) **Capa de Salida:** La capa de salida utiliza un solo neurón con activación *sigmoid*. Explique por qué esta configuración es la ideal para un problema de clasificación binaria. ¿Qué representa el valor de salida de este neurón?

c) **Complejidad del Modelo:** ¿Qué se esperaría que ocurriera con las curvas de pérdida de entrenamiento y validación si se utilizara una red excesivamente simple (ej. una capa oculta con 8 neuronas)? ¿Y con una red excesivamente compleja (ej. 5 capas con 256 neuronas cada una) sin regularización? Relacione su respuesta con los conceptos de *underfitting* y *overfitting*.

d) **Regularización con Dropout:** ¿Cuál es el propósito de las capas de `Dropout`? Explique conceptualmente cómo funciona y por qué ayuda a prevenir el sobreajuste.

e) **Cálculo de Parámetros:** Dado el ‘model.summary()‘ del notebook, explique cómo se calcula el número de parámetros entrenables para la segunda capa oculta (la de 64 neuronas). Muestre el cálculo.

3 Entrenamiento y Optimización

a) **Función de Pérdida:** ¿Por qué se utiliza `binary_crossentropy` como función de pérdida? ¿Cómo se relaciona esta función con la salida probabilística de la capa sigmoide?

b) **Optimizador:** El modelo utiliza el optimizador Adam. ¿Cuál es el rol de un optimizador en el proceso de entrenamiento? Mencione otro optimizador comúnmente utilizado en Deep Learning.

c) **Tamaño del Lote (*Batch Size*):** ¿Qué representa el hiperparámetro `batch_size`? ¿Qué efecto tendría en el entrenamiento usar un tamaño de lote muy pequeño (ej. 32) versus uno muy grande (ej. 1024)?

d) **Pesos de Clase (*Class Weight*):** El notebook calcula y utiliza pesos de clase para manejar el desbalance. Explique cómo esta técnica ayuda al modelo a prestar más atención a la clase minoritaria durante el entrenamiento.

e) **Parada Temprana (*Early Stopping*):** ¿Qué problema de entrenamiento ayuda a mitigar el callback `EarlyStopping`? ¿Qué significan los parámetros `monitor` y `patience`?

4 Evaluación y Justificación de Negocio

a) **Matriz de Confusión:** Analice la matriz de confusión del modelo final. En el contexto del banco, ¿cuál sería el costo de un Falso Positivo (FP) y cuál el de un Falso Negativo (FN)? ¿Qué error considera más perjudicial para el negocio?

b) **Precisión vs. Recall:** Explique el *trade-off* entre precisión y recall para la clase ”Sí”. ¿En qué escenario de negocio el banco preferiría un modelo con mayor precisión? ¿Y uno con mayor recall?

c) **Curva ROC y AUC:** ¿Qué representa la curva ROC y qué mide el valor de AUC (Área Bajo la Curva)? ¿Por qué un AUC de 0.92 es significativamente mejor que un AUC de 0.5?

d) **Umbral de Decisión:** El modelo utiliza por defecto un umbral de 0.5 para clasificar. Si el objetivo es capturar al 75% de los clientes que dirían ”Sí” (Recall = 0.75), ¿debería subir o bajar el umbral? ¿Qué consecuencia esperaría que tenga esta acción sobre la precisión del modelo?