

Reporte de Resultados

1) Decisiones de procesamiento de datos

Antes de entrenar los modelos se tomaron varias decisiones basadas en la estructura del dataset Adult y con el objetivo de evitar fuga de información entre particiones. En cuanto a variables, se emplearon todas las reportadas en *adult.names*, incluyendo numéricas y categóricas, y la variable objetivo *income* se mapeó a 0 para $\leq 50K$ y 1 para $> 50K$. Respecto de los valores faltantes, en *work class*, *occupation* y *native-country* se imputó la categoría *Unknown* para no descartar filas y permitir que el modelo aprenda la señal asociada a la ausencia de información. Para la codificación y el escalamiento, las variables categóricas se transformaron con One-Hot Encoding configurado con *handle_unknown='ignore'* para prevenir errores ante categorías no vistas en validación o prueba, mientras que las numéricas se estandarizaron con *StandardScaler*. Tanto el OHE como el *scaler* se ajustaron exclusivamente sobre el conjunto de entrenamiento para evitar *data leakage*.

El particionado utilizó *adult.data* como entrenamiento y dividió *adult.test* en partes de validación y prueba final con una proporción 50/50 y estratificación por *income* para asegurar estabilidad en las proporciones de clase. En esta entrega no se añadieron características sintéticas; la prioridad fue mantener un flujo reproducible y claro con imputación mínima, codificación OHE para las categóricas y normalización de las numéricas.

2) Hiperparámetros del mejor MLP y comparación sin/con regularización

Probamos varias configuraciones cambiando el tamaño y la profundidad de la red. Para elegir la mejor, no miramos solo cómo aprendía en entrenamiento, sino cómo rendía en el conjunto de validación; además, usamos Early Stopping (*patience=10*) para detener el entrenamiento cuando dejarlo correr más ya no aportaba mejoras en validación. El modelo elegido fue un MLP con tres capas ocultas (256, 128 y 64 neuronas), apagado aleatorio del 50% (*dropout*), penalización ligera de pesos (*weight decay*), tasa de aprendizaje 0.001, tamaño de lote 128 y una pérdida binaria con ponderación para atender el desbalance entre clases.

Al comparar la versión sin regularización (sin apagado aleatorio ni penalización) con la versión regularizada, vimos que la primera rinde muy bien en entrenamiento pero se estanca o empeora en validación, un indicador claro de sobreajuste. En cambio, con regularización y Early Stopping (*patience=10*), la curva de validación se mantiene estable y el modelo generaliza mejor, logrando más aciertos en los casos $> 50K$ y mejor separación entre clases. En resumen, ponerle límites al modelo (apagado aleatorio + penalización + detener a tiempo) sí marca la diferencia; si lo que buscamos es capturar más positivos, esta configuración es la que conviene.

3) Comparación del mejor MLP frente a Regresión Logística (línea base) e interpretación

Evaluación del mejor MLP

Split	Accuracy	Precisión	Recall	F1	ROC-AUC
Train	0.8274	0.5946	0.8901	0.7129	0.9277
Val	0.8104	0.5654	0.8539	0.6803	0.9091
Test	0.8111	0.5655	0.8638	0.6835	0.9143

Comparativa final: LogReg vs MLP

Modelo	Accuracy	Precision	Recall	F1	ROC-AUC
LogReg	0.8571	0.7375	0.6136	0.6699	0.9081
MLP	0.8111	0.5655	0.8638	0.6835	0.9143

El mejor MLP evaluado obtuvo, en *train*, *val* y *test*, desempeños consistentes: *accuracy* 0,8274, 0,8104 y 0,8111; *precision* 0,5946, 0,5654 y 0,5655; *recall* 0,8901, 0,8539 y 0,8638; *F1* 0,7129, 0,6803 y 0,6835; y ROC-AUC 0,9277, 0,9091 y 0,9143, respectivamente. En la comparativa final entre modelos, la Regresión Logística alcanzó *accuracy* 0,8571, *precision* 0,7375, *recall* 0,6136, *F1* 0,6699 y ROC-AUC 0,9081; por su parte, el MLP logró *accuracy* 0,8111, *precision* 0,5655, *recall* 0,8638, *F1* 0,6835 y ROC-AUC 0,9143.

La interpretación es la siguiente. El MLP consigue un *recall* superior (0,864) y un ROC-AUC más alto (0,914), lo que indica que detecta más casos de ingresos mayores 50K y separa mejor las clases en términos probabilísticos; es especialmente útil cuando el costo de perder un positivo (falso negativo) es alto. La Regresión Logística presenta mayor *accuracy* (0,857) y mejor *precision* (0,738), por lo que comete menos falsos positivos, algo ventajoso si estos resultan más costosos; sin embargo, deja escapar más positivos reales debido a su menor *recall*. El *F1* del MLP (0,683) supera ligeramente al de la Logística (0,670), coherente con un aumento notable del *recall* incluso sacrificando cierta *precision*. Operativamente, no conviene usar un umbral fijo de 0,5; es preferible ajustar el umbral con la curva *Precision–Recall* y una función de costos que refleje el balance entre falsos negativos y falsos positivos para el problema. El MLP, además, permite desplazar el punto de operación para elevar la *precision* manteniendo un *recall* alto si así se requiere.

Notas finales

En este trabajo construimos un flujo claro y reproducible: pre procesamos el Adult sin fuga de información (imputación a *Unknown*, OHE con escalado solo en *train*), y comparamos modelos cuidando el desbalance con *pos_weight*. La regularización (dropout + weight decay + early stopping) fue clave para que el MLP generaliza mejor: evitar el sobreajuste y elevó *recall* y ROC-AUC. En números, el MLP detectó más casos >50K (*recall* 0.86 y AUC 0.91), mientras que la Regresión Logística tuvo mejor *accuracy* y *precisión*, es decir, cometió menos falsos positivos pero dejó pasar más positivos reales.

Con eso, la elección del modelo depende del costo de equivocarse: si es peor perder un positivo (FN), conviene el MLP; si es más costoso marcar falsos positivos (FP), la Logística funciona mejor como base. En cualquier caso, no es buena idea usar umbral 0.5 “por defecto”: proponemos ajustar el umbral con la curva Precision-Recall y una función de costos, e incluso calibrar probabilidades para decisiones más estables. Como siguientes pasos, vale la pena probar ingeniería de características simple, una búsqueda de hiperparámetros más fina y análisis de interpretabilidad, para revisar sesgos por subgrupos y hacer un despliegue más robusto.