

MAESTRÍA EN INTELIGENCIA DE NEGOCIOS Y CIENCIA DE DATOS

MODALIDAD: ONLINE 2024
INTELIGENCIA DE NEGOCIO

PROYECTO 6: SISTEMA DE OPTIMIZACIÓN DE PRECIOS DINÁMICO

INFORME TÉCNICO PREDICCIÓN DE DEMANDA DE LAPTOPS



Desarrollar un modelo de red neuronal para estimar la demanda de laptops usando características técnicas y de precio.

Grupo 5

- Mery Lorena Romero Gonzales.
- Jenniffer Mercedes Lainez Rosado.
- Karina Marisela Cali Cadena.
- Jose Manuel Navarro Oyarvide.
- Rubén Fernando Guamán Armijos.
- Andy Hans Rodriguez Lino.

Docente

Gladys Maria Villegas Rugel

10 de agosto 2025

Este proyecto tiene como objetivo el desarrollo y evaluación de un modelo de Red Neuronal Artificial (RNA) para predecir la demanda de laptops en función de características técnicas y de precio. La motivación principal es identificar cómo factores como la marca, el procesador, la memoria RAM, el almacenamiento y el precio influyen en el volumen de ventas esperado.

Preparación de Datos

El dataset consta de 1303 registros y 28 variables predictoras. Entre las más relevantes se incluyen:

- Atributos de hardware: pulgadas de pantalla, resolución, procesador, memoria RAM, almacenamiento, tarjeta gráfica.
- Atributos de mercado: marca, tipo de producto, sistema operativo.
- Variable objetivo: Demanda (unidades estimadas de venta).

Arquitectura y Entrenamiento

La arquitectura base de la red fue:

```
= NeuralNetwork(layers=[X.shape[1], 128, 64, 32, 1], activation='tanh')  
predictions = nn.predict(X_scaled)  
predictions_original = scaler_Y.inverse_transform(predictions)
```

- Activación tanh: seleccionada por estabilidad frente a ReLU (que provocó NaN en el loss).
- Learning rate bajo (0.0001) para prevenir explosiones de gradiente.
- Entrenamiento en 1000 épocas, con seguimiento de la pérdida por intervalos de 100 épocas.

Preprocesamiento

Antes del entrenamiento se aplicaron las siguientes transformaciones:

- Conversión de texto con unidades (GB, kg, GHz) a valores numéricos.
- Codificación de variables categóricas, OneHotEncoding o extracción de términos clave.
- Escalado de:
 - X con StandardScaler (media 0, desviación estándar 1).
 - Y con MinMaxScaler (rango [0, 1]) para evitar inestabilidad en el entrenamiento.
- Eliminación de valores NaN y inf en todas las columnas.

Resultados Experimentales

Se realizaron pruebas con diferentes configuraciones:

Evolución del Loss – Escalado correcto

Epoch 0	Loss: 1.5640
Epoch 100	Loss: 1.4230
Epoch 200	Loss: 1.2973
Epoch 300	Loss: 1.1851
Epoch 400	Loss: 1.0848
Epoch 500	Loss: 0.9951
Epoch 600	Loss: 0.9147
Epoch 700	Loss: 0.8427
Epoch 800	Loss: 0.7781
Epoch 900	Loss: 0.7201

El loss decreció progresivamente hasta estabilizarse.

Caso sin escalar la variable objetivo

Epoch 0	Loss: 207913.1992
Epoch 100	Loss: 1386.0208
Epoch 200	Loss: 752.6918
Epoch 300	Loss: 675.2305
Epoch 400	Loss: 641.8407
Epoch 500	Loss: 622.9860
Epoch 600	Loss: 609.6185
Epoch 700	Loss: 599.6755
Epoch 800	Loss: 592.6674
Epoch 900	Loss: 587.1037

El valor inicial fue extremadamente alto y decreció lentamente, confirmando la importancia de escalar Y.

Caso con variables sin normalizar y activación no óptima

Epoch 0	Loss: 207597.3566
Epoch 100	Loss: 52247.9551
Epoch 200	Loss: 42911.0043
Epoch 300	Loss: 41242.3612
Epoch 400	Loss: 40948.6182
Epoch 500	Loss: 40871.1889
Epoch 600	Loss: 40822.6937
Epoch 700	Loss: 40808.2626
Epoch 800	Loss: 40780.6653
Epoch 900	Loss: 40753.2420

El **loss** permaneció alto, sin mejora significativa, demostrando la ineficiencia del modelo con datos no preprocesados.

Comparación de Arquitecturas

Se evaluaron tres configuraciones de red neuronal, manteniendo mismas entradas/salidas y solo variando número de capas y neuronas:

Configuración	Capas ocultas	Loss Final
A	[64, 32]	0.85
B (base)	[128, 64, 32]	0.72
C	[256, 128, 64, 32]	0.70

Aumentar la complejidad mejora el ajuste, pero puede generar más costo computacional sin una ganancia significativa. La configuración B ofrece buen equilibrio. Se concluye, que a mayor capacidad, el error baja, pero el coste computacional aumenta.

Funciones de Activación

Activación	Loss Final	Observación
ReLU	NaN	Inestabilidad
tanh	0.72	Mejor equilibrio
Sigmoid	0.81	Saturación de gradiente

tanh es más adecuado para este problema de regresión, especialmente cuando las entradas y salidas están escaladas.

Hiperparámetros

LR	Epochs	Loss Final	Observación
0.01	1000	NaN	Explosión de gradiente
0.001	1000	0.75	Convergencia lenta
0.0001	1000	0.72	Óptima

Un learning rate bajo es crítico para evitar errores numéricos y permitir convergencia gradual y precisa, es decir, una tasa de aprendizaje elevada provocó que la pérdida aumentara drásticamente y no se estabilizara.

Mejor equilibrio con LR = 0.0001 y 1500 épocas.

Comparación con Baseline

Modelo	RMSE (escala original)
Regresión Lineal	264.5
RNA óptima (tanh)	192.1

La red neuronal captura relaciones no lineales que un modelo lineal no puede, por lo que resulta más efectivo para este problema.

Conclusión

- El preprocesamiento especialmente el escalado de y fue determinante para la estabilidad del modelo.
- tanh fue la activación más robusta para este dataset.
- La arquitectura [128, 64, 32] logró buen equilibrio entre precisión y coste computacional.
- El modelo superó al baseline lineal en un 27% de reducción del RMSE.
- Casos sin normalización o con activaciones inadecuadas evidenciaron pérdidas excesivas y falta de convergencia.

Esto demuestra la importancia de preparar correctamente los datos y ajustar los hiperparámetros de forma iterativa.

Por otra parte, el modelo construido permitió identificar que variables como la marca del equipo, el tipo de procesador, la capacidad de memoria RAM, el tipo y tamaño de almacenamiento, así como el precio del dispositivo, influyen significativamente en el volumen de ventas esperado de laptops. Estas variables técnicas presentan patrones de correlación que fueron correctamente captados por la red neuronal. Esto valida el enfoque del modelo como una herramienta útil para proyectar demanda con base en características de producto.

Recomendación

- Seguir ajustando hiperparámetros.
- Incluir técnicas de regularización.
- Ampliar el dataset para mayor robustez.

Link

https://colab.research.google.com/drive/1vQZFHUW_dusq36QBJi-k5BpYSihxjFfZ