

Informe Laboratorio #2

Describe el objetivo de la red y por qué considera que la implementación de una red neuronal puede obtener el resultado deseado

El objetivo de esta implementación de red neuronal es resolver un problema de regresión. Consiste en predecir los valores numéricos correspondientes a las variables 'Precio de producción', 'Precio de producción global' y 'Precio de producción nacional' de las películas en función de las características 'movie_averageRating', 'movie_numOfVotes' y 'aprobación_Index'. Las variables objetivas y su naturaleza continua, así como la necesidad de predecir valores numéricos en lugar de clasificarlos en categorías discretas, fueron las razones de la elección del método de regresión.

Gracias a que una red neuronal puede modelar relaciones no lineales y patrones complejos en los datos respalda su uso para utilizarla en este laboratorio. También pueden capturar interacciones y dependencias entre características que modelos lineales no podrían. Esto hace a las redes neuronales ideales para tareas complejas como esta.

Las redes neuronales también tienen la capacidad de aprender estructuras jerárquicas mientras lee la información propagándose a través de múltiples capas. Esto brinda gran beneficio al momento de trabajar con datos de múltiples características, como en este caso, donde las características de una película pueden influir de manera compleja en su presupuesto de producción y ganancias.

Describe la composición y los resultados obtenidos por red neuronal

Red neuronal	Red Neuronal 1: Arquitectura con Batch Normalization y Softmax (Clasificación)	Red Neuronal 2: Arquitectura con Leaky ReLU y Linear (Regresión)	Red Neuronal 3: Arquitectura con ReLU y Linear (Regresión)
Descripción	Diseñada para resolver un problema de clasificación	Diseñada para resolver un problema de clasificación	Diseñada para un problema de regresión

Capas	<ol style="list-style-type: none"> 1. Capa de Entrada (Dense): 128 neuronas, activación ReLU. 2. Capa de Normalización por lotes (Batch Normalization). 3. Capa Oculta (Dense): 256 neuronas, activación ReLU. 4. Capa de Normalización por lotes (Batch Normalization). 5. Capa Oculta (Dense): 256 neuronas, activación ReLU. 6. Capa de Normalización por lotes (Batch Normalization). 7. Capa Oculta (Dense): 128 neuronas, activación ReLU. 8. Capa de Normalización por lotes (Batch Normalization). 9. Capa Oculta (Dense): 64 neuronas, activación ReLU. 	<ol style="list-style-type: none"> 1. Capa de Entrada (Dense): 128 neuronas, activación Linear. 2. Capa de Activación (LeakyReLU) con un factor alpha de 1.0. 3. Capa de Normalización por lotes (Batch Normalization). 4. Capa Oculta (Dense): 256 neuronas, activación Linear. 5. Capa de Activación (LeakyReLU) con un factor alpha de 1.0. 6. Capa de Normalización por lotes (Batch Normalization). 7. Capa Oculta (Dense): 256 neuronas, activación Linear. 8. Capa de Activación (LeakyReLU) con un factor alpha de 1.0. 9. Capa de Normalización por lotes 	<ol style="list-style-type: none"> 1. Capa de Entrada (Dense): 128 neuronas, activación Linear. 2. Capa de Activación (ReLU). 3. Capa de Normalización por lotes (Batch Normalization). 4. Capa Oculta (Dense): 256 neuronas, activación Linear. 5. Capa de Activación (ReLU). 6. Capa de Normalización por lotes (Batch Normalization). 7. Capa Oculta (Dense): 256 neuronas, activación Linear. 8. Capa de Activación (ReLU). 9. Capa de Normalización por lotes (Batch Normalization). 10. Capa Oculta (Dense): 128
-------	---	---	--

	<p>10. Capa de Normalización por lotes (Batch Normalization).</p> <p>11. Capa de Salida (Dense): 3 neuronas, activación Softmax.</p>	<p>(Batch Normalization).</p> <p>10. Capa Oculta (Dense): 128 neuronas, activación Linear.</p> <p>11. Capa de Activación (LeakyReLU) con un factor alpha de 1.0.</p> <p>12. Capa de Normalización por lotes (Batch Normalization).</p> <p>13. Capa Oculta (Dense): 64 neuronas, activación Linear.</p> <p>14. Capa de Activación (LeakyReLU) con un factor alpha de 1.0.</p> <p>15. Capa de Normalización por lotes (Batch Normalization).</p> <p>16. Capa de Salida (Dense): 3 neuronas, activación Linear.</p>	<p>neuronas, activación Linear.</p> <p>11. Capa de Activación (ReLU).</p> <p>12. Capa de Normalización por lotes (Batch Normalization).</p> <p>13. Capa Oculta (Dense): 64 neuronas, activación Linear.</p> <p>14. Capa de Activación (ReLU).</p> <p>15. Capa de Normalización por lotes (Batch Normalization).</p> <p>16. Capa de Salida (Dense): 3 neuronas, activación Linear.</p>
--	--	--	---

Resultados

- Resultados de la Red Neuronal 1:

28/28 - 0s - loss: 22097226633314304.0000 - accuracy: 0.6895 - 36ms/epoch - 1ms/step

Loss: 2.2097226633314304e+16

Accuracy: 0.689497709274292

28/28 [=====] - 0s 831us/step

RMSE (Red Neuronal 1): 148651355.0370161

R² (Red Neuronal 1): -0.5256261082247992

- Resultados de la Red Neuronal 2:

28/28 - 0s - loss: 67779448.0000 - accuracy: 0.4258 - 35ms/epoch - 1ms/step

Loss: 67779448.0

Accuracy: 0.42579910159111023

28/28 [=====] - 0s 808us/step

RMSE (Red Neuronal 2): 148651352.0772564

R² (Red Neuronal 2): -0.5256259839914829

- Resultados de la Red Neuronal 3:

28/28 - 0s - loss: 67778400.0000 - accuracy: 0.6895 - 36ms/epoch - 1ms/step

Loss: 67778400.0

Accuracy: 0.689497709274292

28/28 [=====] - 0s 827us/step

RMSE (Red Neuronal 3): 148650820.32231092

R² (Red Neuronal 3): -0.5256012807938001

Se usaron varios métodos para ver la eficiencia y exactitud de cada red neuronal. Loss indica que tan lejos están las predicciones del valor real. Los valores están muy altos para todos, lo que dice que las predicciones están muy alejadas de los valores reales. Accuracy no es muy relevante, y puede ser ignorada debido a que generalmente se usa en problemas de clasificación. RMSE mide cuánto varían las predicciones en relación con los

valores reales. Entre menos valor, mejor. R^2 representa que tan bien se ajustan las predicciones a valores reales. Entre más cerca de 1, mejor.

En base a los resultados, los modelos no lograron un buen ajuste a los datos. Las tres redes neuronales tienen resultados muy similares en los RMSE y R^2 . Todos los modelos tienen valores de R^2 muy cercanos a -0.53 y valores de RMSE alrededor de 148,650,000. Los valores negativos de R^2 y los altos valores de RMSE y pérdida indican que las predicciones están muy alejadas de los valores reales. Esto puede ser por varias razones, como la complejidad de la red, la cantidad de datos, la selección de características y los hiper parámetros.

Discuta la diferencia de rendimiento y conceptuales en la composición y resultados obtenidos en cada red neuronal

Red 1: Tiene una estructura con varias capas densas y activaciones ReLU en las capas ocultas. Se usa una función de pérdida de error cuadrático medio (MSE) para optimización. A pesar de ser una red profunda, los resultados muestran un alto valor de pérdida y bajo R^2 , lo que indica un mal ajuste a los datos. Es posible que esta red esté sufriendo de sobreajuste debido a su profundidad.

Red 2: Esta red usa activaciones Leaky ReLU en vez de ReLU. Se usa la función de pérdida Pseudo-Huber para la optimización. Igual que la Red 1, muestra resultados similares con alto valor de pérdida y bajo R^2 . A pesar del cambio en las activaciones y la función de pérdida, no hay mejoras.

Red 3: Usa activaciones ReLU, pero agrega Batch Normalization después de cada capa densa. Se usa la función de pérdida Pseudo-Huber para la optimización. A pesar de agregar Batch Normalization, los resultados siguen siendo igual de deficientes que las otras redes, lo que muestra que esta técnica no ha logrado mejorar el rendimiento.

Conceptualmente, las diferencias entre redes no tuvieron un impacto en el rendimiento. Tal vez la complejidad de los datos y dificultad para que las redes capturen patrones adecuados haya sido la razón. O también puede ser que la arquitectura de redes no haya sido la más adecuada para este problema.

Seleccione la red neuronal óptima y justifique su respuesta

Es difícil elegir una red óptima porque las 3 arrojaron resultados similares en términos de pérdida, precisión, RMSE y R^2 . Sin embargo, considerando las características y las diferencias conceptuales entre las redes, posiblemente sea la Red 2 la más prometedora.

Red Neuronal 2:

- Esta red usa activaciones Leaky ReLU en lugar de ReLU.
- Implementa la función de pérdida Pseudo-Huber, que puede ser más firme frente a valores atípicos y errores grandes.
- Aunque los resultados no son tan diferentes de las otras redes, la elección de activaciones diferentes y una función de pérdida menos sensible al error cuadrático podrían tener un impacto positivo en la generalización del modelo.

Sin embargo, es importante mencionar que los resultados actuales no son buenos. Se deben explorar y cambiar aún más los ajustes en las redes neuronales, hiper parámetros y técnicas de preprocesamiento, para lograr un resultado satisfactorio.