

Checkpoint 4 - Grupo 05

Introducción

Para este checkpoint entrenamos 4 modelos de redes neuronales optimizando distintos hiperparametros para ir obteniendo mejores resultados y puntajes. Empezamos haciendo modificaciones en ambos datasets (testing y training) para poder implementar redes neuronales. Concatenamos el año, el mes y el día en única variable del dataset. Convertimos el arrival date a date time para luego pasarlo a un valor numérico y a un int calculando una diferencia de días desde cierta fecha que usamos como referencia. Convertimos child de float a int y juntamos children y babies en una misma columna.

Para poder manejar los valores NaN de Agent y Company los convertimos al valor -1. Luego, preparamos los datos mediante One-Hot Encoding para representar características categóricas de manera binaria.

Usamos la técnica de preprocesamiento StandardScaler para estandarizar las variables.

Construcción del modelo

A continuación detallamos los siguientes puntos para el “mejor” modelo obtenido:

- ¿Cuál fue la arquitectura escogida?
 - Capa entrada
 - neuronas de entrada: 190 neuronas de entrada
 - Función de activación: 'relu'
 - Tipo de conexión: la capa de entrada se conecta directamente a la primera capa oculta (dense)
 - Capas ocultas
 - neuronas por capa: 64
 - Función de activación: la función de activación en la capa oculta puede ser 'relu', 'tanh', 'sigmoid', 'swish', 'leaky_relu', 'elu'
 - Tipo de conexión: dense
 - Capa de salida
 - cantidad de neuronas: 1
 - función de activación: 'swish'
- ¿Qué hiperparametros se optimizaron?

```
'learning_rate': [0.001, 0.01, 0.1, 0.2], 'units': [64, 128, 256],  
'activation': ['relu', 'tanh', 'sigmoid', 'swish', 'leaky_relu',  
'elu'], 'epochs': [50, 100, 200, 300], 'batch_size': [32, 64, 128,  
256]
```

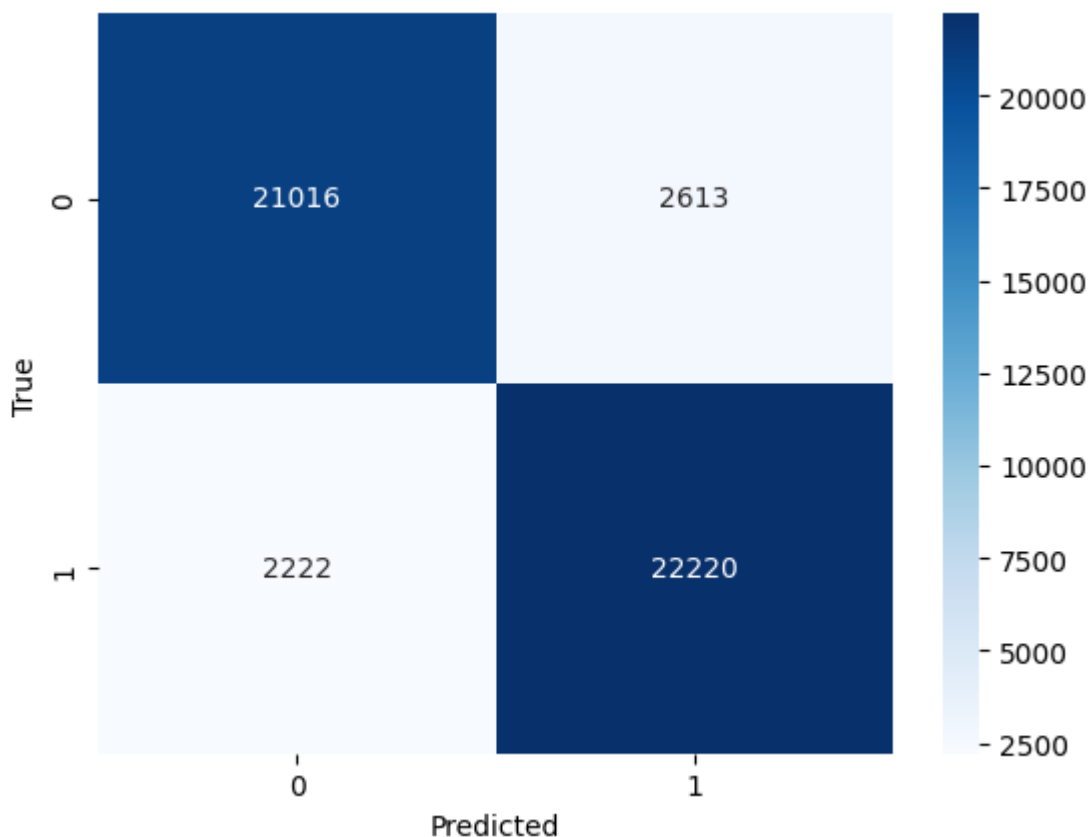
- ¿Qué optimizador se utilizó?
Se utilizó el optimizador SGD (Gradiente Descendente Estocástico) con 'learning_rate': 0.02
- ¿Se utilizó alguna técnica de regularización? ¿Cuál?
No se utilizaron técnicas de regularización.
- ¿Cuántos ciclos de entrenamiento utilizó?
En el mejor modelo encontrado, se utilizaron 200 Epochs para el entrenamiento.

Cuadro de Resultados

Modelo	F1-Test	Presicion Test	Recall Test	Accuracy	Kaggle
modelo_1	0.804511	0.826649	0.783528	0.806390	0.80161
modelo_2	0.890896	0.871013	0.911709	0.886459	0.8271
modelo_3	0.893303	0.893431	0.893175	0.891514	0.84145
modelo_4	0.901877	0.894777	0.909090	0.899419	0.83928

Se puede observar que el modelo_4 es el más óptimo.

Matriz de Confusion



En la matriz se puede observar que hay 22220 casos en los que el modelo predijo que se cancelaba y efectivamente era así. 2222 casos en los que predijo que no se cancelaba pero en realidad eran reservas canceladas. 21016 casos en los que el modelo dijo que no se iba a cancelar y era así. Y 2613 casos que se predijo que se iban a cancelar y no fue así.

Se observa que la mayoría de las reservas fueron predichas correctamente.

Tareas Realizadas

Integrante	Tarea
Iara Jolodovsky	Modelo 2, 3 y 4, reporte
Martín Abramovich	Modelo 1 y 2, reporte
Tomás Vainstein	Modelo 1 y 4, reporte