Explicación de solución punto 4

Por: Sebastian Carmona Estrada

Fecha: 06/02/2024

Prueba: **PRUEBA DE MODELACIÓN ANALÍTICA**

"1. Cargue el archivo train.csv y Construya un modelo que capaz de realizar predicciones de FRAUDE. 2. Enviar un archivo test_evaluado.csv con todas las columnas en el mismo orden que se encuentran en test.csv y adicionalmente la columna FRAUDE poblada con el valor predicho por su modelo. Cualquier valor"

Solución:

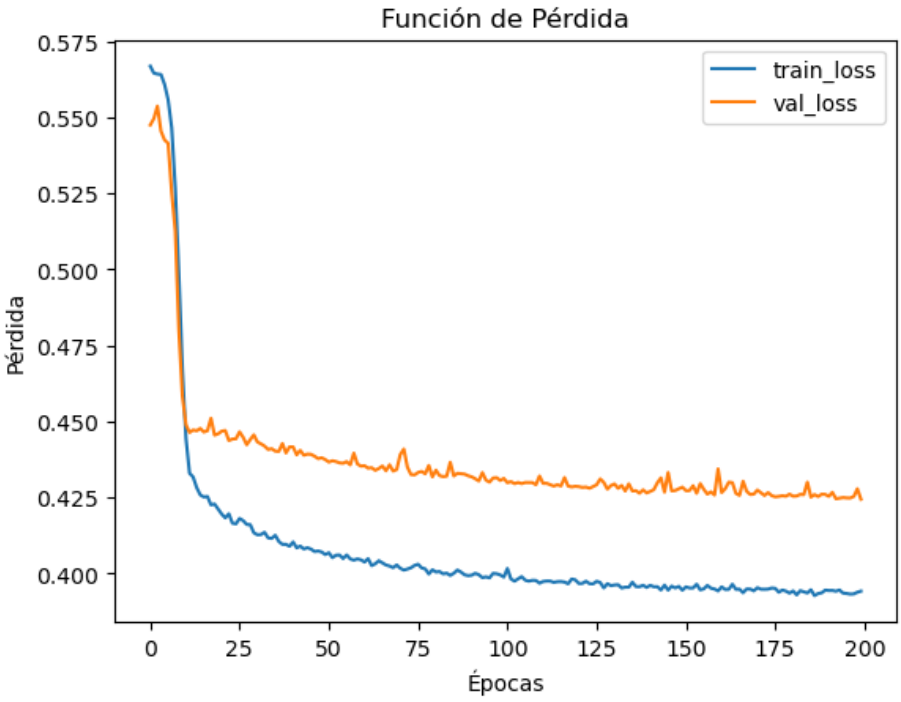
El modelo construido es una red neuronal con la siguiente estructura:

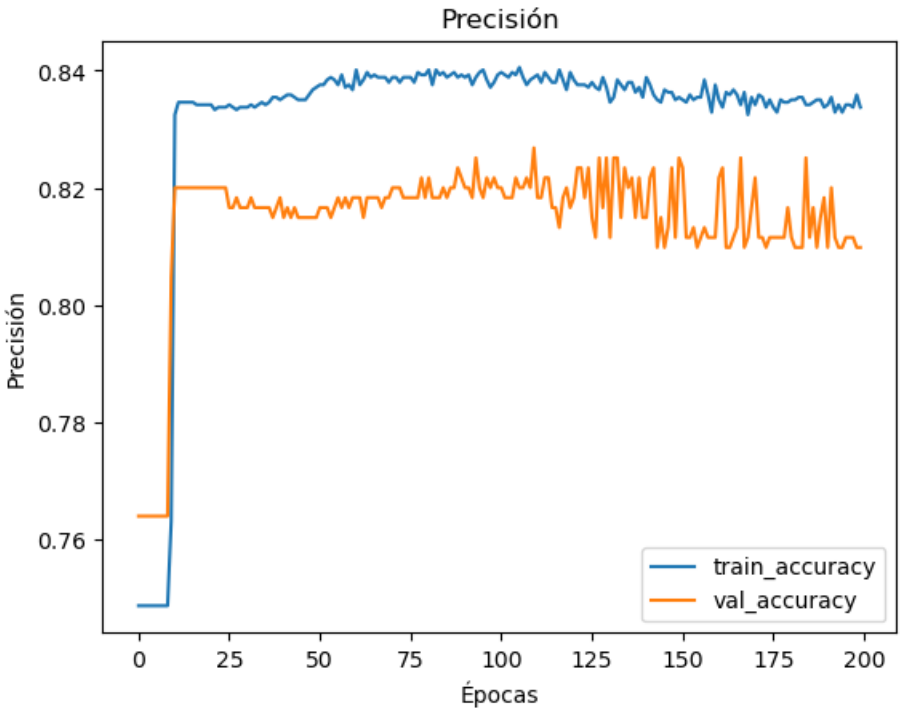
```
model = tf.keras.Sequential([
    tf.keras.layers.Dense(1, activation='sigmoid',
input_shape=(ds_train.shape[1] - 1,)),
    tf.keras.layers.Dense(32, activation='sigmoid'),
    tf.keras.layers.Dense(32, activation='sigmoid'),
    tf.keras.layers.Dense(1, activation='sigmoid')
    ])


    model.compile(optimizer='adam',
                loss='binary_crossentropy',
                metrics=['accuracy'])
```

Se experimentó con varias configuraciones de capas ocultas, cantidad de neuronas y función de activación. Sin embargo, debido a la poca cantidad de datos que se tiene (2941 observaciones), el modelo entrena correctamente a partir de una capa oculta con 5 neuronas. El optimizador elegido corresponde a un "adam", el cual es igual un descenso del gradiente estocástico, pero con la diferencia de que se incluye "momentum", lo cual puede acelerar el proceso de aprendizaje.

Como se trata de un problema de clasificación binaria, se utilizó una función de pérdida crossentropy', la cual es la adecuada para este tipo de problemas. La métrica utilizada para analizar la precisión del modelo corresponde a una métrica de precisión. Se podrían utilizar otras métricas si se desea realizar un análisis más profundo.

La curva de aprendizaje y error obtenidas se muestra a continuación:





Las variables utilizadas para entrenar el modelo son las siguientes:

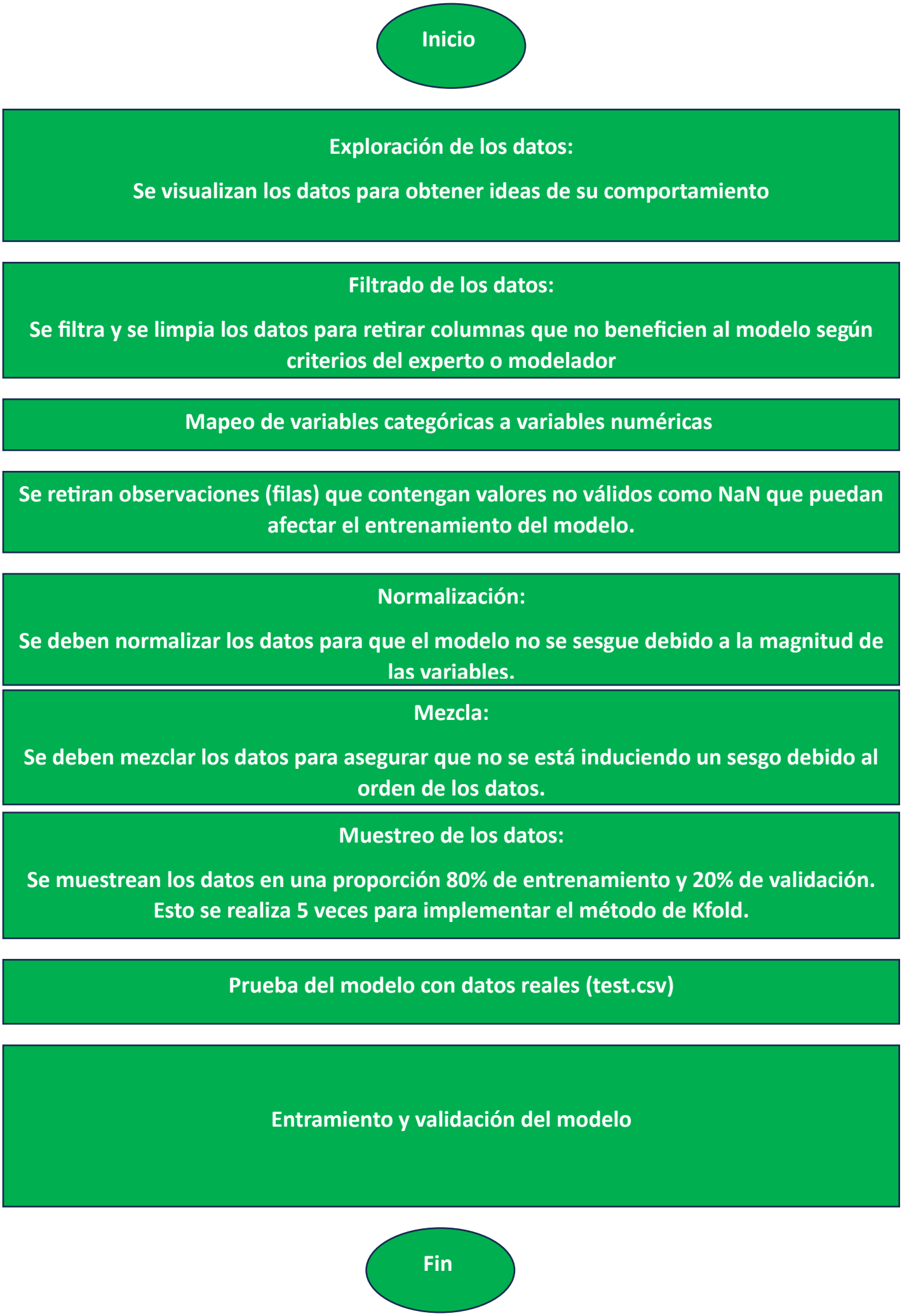
1. VALOR,
2. Canal1,
3. COD_PAIS,
4. CANAL,
5. DIASEM,
6. DIAMES,
7. FECHA_VIN,
8. OFICINA_VIN,
9. SEXO,
10. SEGMENTO,
11. EDAD,
12. INGRESOS,
13. EGRESOS,
14. NROPAISES,
15. NROCIUDADES,
16. Dist_HOY,
17. Dist_sum_NAL

Se debe tener en cuenta que es importante realizar un análisis de informatividad de las variables para entender el nivel de informatividad de cada una de estás para el modelo. Por ejemplo, la información de la variable CANAL y Canal1, es muy similar. Por lo anterior, esta variable podría sesgar el modelo ya que el modelo recibe dos veces la misma información. A pesar de lo anterior, no hubo diferencia alguna al retirar alguna de estas dos variables. Por otro lado, es importante tener en cuenta que utilizar variables temporales puede sesgar fácilmente el modelo.

El modelo obtuvo una precisión promedio a partir de la técnica de KFolds de 83 %.

Preprocesado:

Una etapa fundamental del entrenamiento de redes neuronales es el preprocesado de los datos. El preprocesado implementado para este modelo corresponde al siguiente diagrama de flujo:


Inicio
Exploración de los datos:
Se visualizan los datos para obtener ideas de su comportamiento
Filtrado de los datos:
Se filtra y se limpia los datos para retirar columnas que no beneficien al modelo según criterios del experto o modelador
Mapeo de variables categóricas a variables numéricas
Se retiran observaciones (filas) que contengan valores no válidos como NaN que puedan afectar el entrenamiento del modelo.
Normalización:
Se deben normalizar los datos para que el modelo no se sesgue debido a la magnitud de las variables.
Mezcla:
Se deben mezclar los datos para asegurar que no se está induciendo un sesgo debido al orden de los datos.
Muestreo de los datos:
Se muestrean los datos en una proporción 80% de entrenamiento y 20% de validación. Esto se realiza 5 veces para implementar el método de Kfold.
Prueba del modelo con datos reales (test.csv)
Entramiento y validación del modelo
Fin