

## Implementación del modelo en H2O

El uso de H2O viene dado principalmente para realizar una comparación entre los resultados obtenidos en esta plataforma y en PyTorch.

Por otro lado, uno de los objetivos de su utilización es que aquellos estudiantes que no se vean preparados para implementar o desarrollar código relativo a las redes neuronales, puedan también hacer uso de una red para aprender modificando los parámetros de ésta observando los resultados que conlleva.

Para empezar a hacer uso de H2O Flow, lo primero es haber realizado su instalación y haber accedido a la interfaz web en <http://localhost:54321>.

La interfaz gráfica proporcionada por H2O Flow consta de un menú superior con el que se pueden realizar distintas acciones, como importar archivos o crear modelos, que son las que atañen en este caso.



*Figura 1. Barra de menú de H2O Flow*

Debajo de este menú, aparece un proyecto por defecto con una barra de tareas para realizar acciones sobre este y un asistente con las acciones más recomendadas e importantes.

Por último, a la derecha de la página web, aparece una sección de ayuda en la que se puede aprender sobre el uso de H2O.

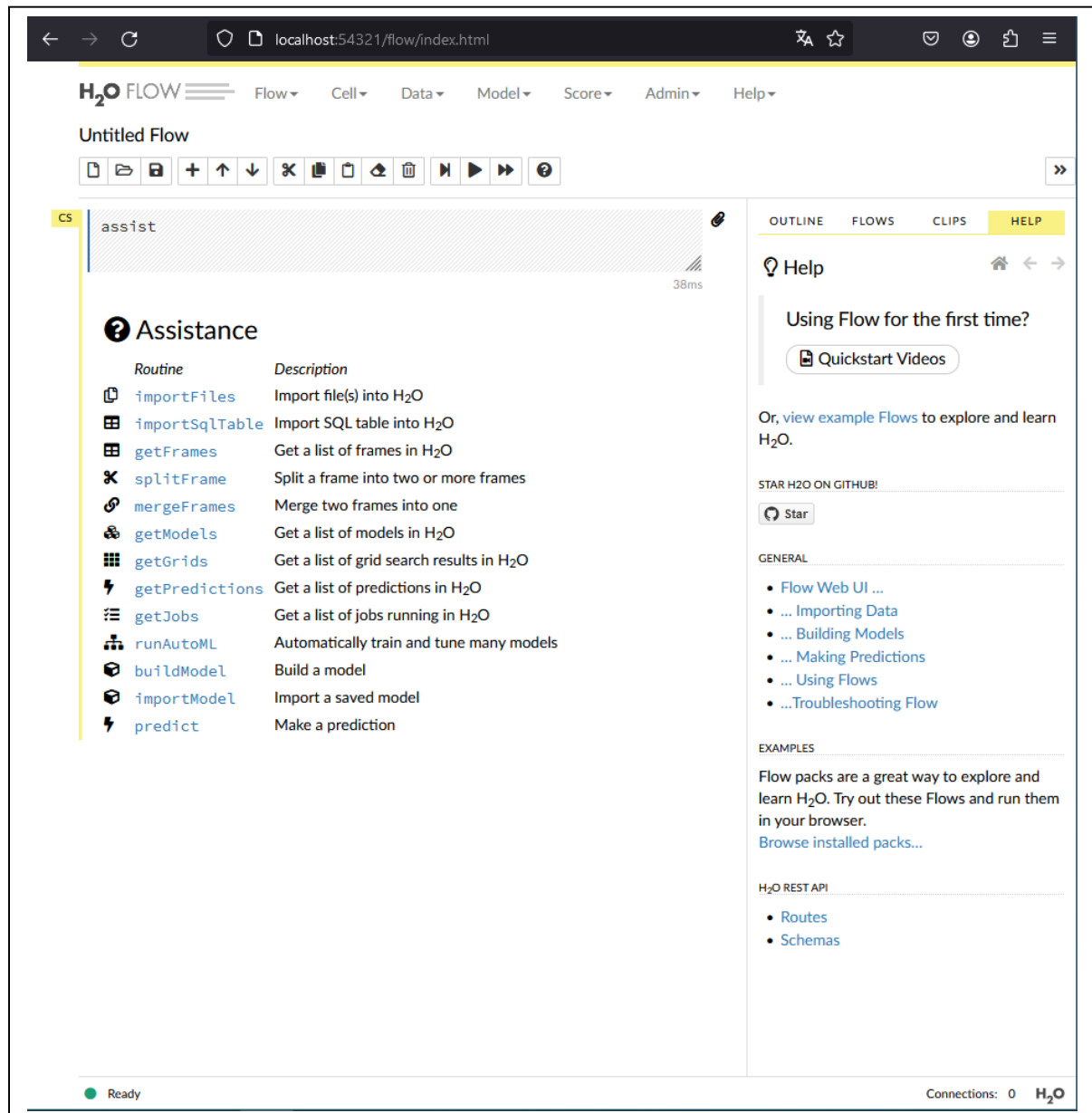


Figura 2. Interfaz gráfica H2O Flow

Para el desarrollo de la red que se busca, lo primero que se ha hecho ha sido importar los archivos con los conjuntos de datos, es decir, los diferentes dataset. Y después se ha hecho “Parse” de los ficheros, donde se ha indicado el tipo de fichero que es, en este caso “csv”, y el separador de columnas dentro del fichero, “;”.

También, se ha indicado el nombre de las columnas y el tipo de dato de éstas. Esto se ha hecho debido a que la estructura del fichero es la siguiente:

```
1row;2row;3row;4row;5row;6row;7row;8row;9row;10row;11row;12row;13row;14row;15row;
0;1;1;1;0;1;0;0;0;1;1;0;0;0;1;1;1;1;1;1;0;0;0;1;1;0;0;0;1;a
1;1;1;1;0;1;0;0;0;1;1;0;0;1;1;1;1;0;1;0;1;0;0;0;1;1;0;0;0;1;b
0;1;1;1;0;1;0;0;0;1;1;0;0;0;0;1;0;0;0;0;1;0;0;0;1;0;1;1;1;0;c
1;1;1;1;0;1;0;0;0;1;1;0;0;0;1;1;0;0;0;1;1;0;0;0;1;1;1;1;1;0;d
1;1;1;1;1;1;0;0;0;0;1;0;0;0;0;1;1;1;1;0;1;0;0;0;0;1;0;0;0;0;e
1;1;1;1;1;1;0;0;0;0;1;0;0;0;0;1;1;1;1;0;1;0;0;0;0;0;0;0;0;0;f
0;1;1;1;0;1;0;0;0;1;1;0;0;0;0;1;0;1;1;1;1;0;0;0;1;1;0;0;0;0;g
1;1;0;0;1;1;0;0;0;1;1;0;0;0;1;1;1;1;1;1;0;0;0;1;1;0;0;0;1;h
```

Figura 3. Estructura de los datasets utilizados en H2O

Además, se seleccionado la opción de que la primera línea del fichero contenga los nombres de las columnas.

**Setup Parse**

PARSE CONFIGURATION

Sources nfs:\C:\Users\Bowly\Desktop\datasets\letra\letra5k2.csv
ID letra5k2.hex
Parser 

CSV

Separator 

:: '059'

Escape Character 0
Column Headers 

Auto

☒ First row contains column names

☐ First row contains data

Options 

☐ Enable single quotes as a field quotation character

☒ Delete on done

EDIT COLUMN NAMES AND TYPES

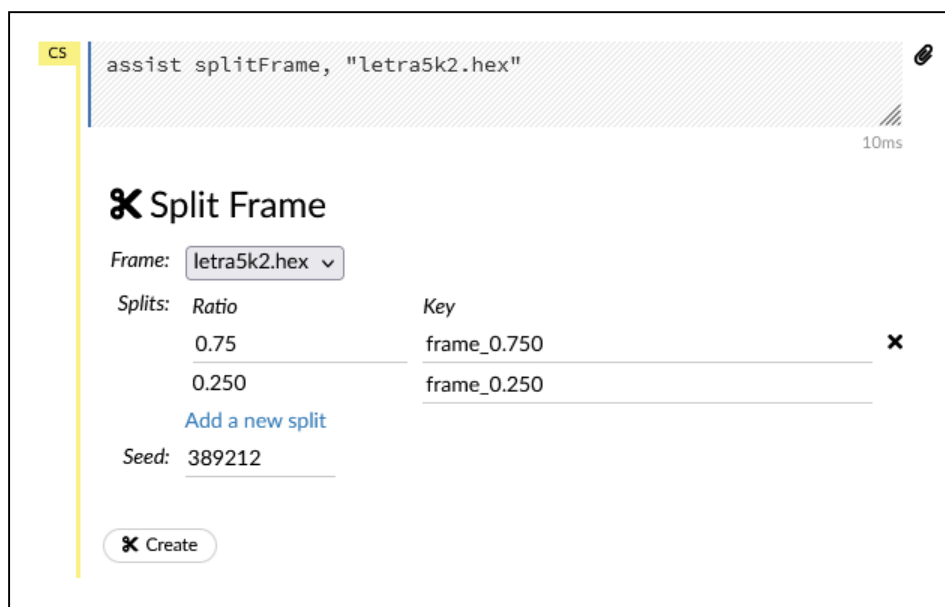
Search by column name...

1	1row	<div>Numeric</div>	0 1 0 1 1 1 0 1 0
2	2row	<div>Numeric</div>	1 1 1 1 1 1 1 1 1
3	3row	<div>Numeric</div>	1 1 1 1 1 1 1 0 1
4	4row	<div>Numeric</div>	1 1 1 1 1 1 1 0 1
5	5row	<div>Numeric</div>	0 0 0 0 1 1 0 1 0
6	6row	<div>Numeric</div>	1 1 1 1 1 1 1 1 0
7	7row	<div>Numeric</div>	0 0 0 0 0 0 0 0 0
8	8row	<div>Numeric</div>	0 0 0 0 0 0 0 0 1
9	9row	<div>Numeric</div>	0 0 0 0 0 0 0 0 0
10	10row	<div>Numeric</div>	1 1 1 1 0 0 1 1 0
11	11row	<div>Numeric</div>	1 1 1 1 1 1 1 1 0
12	12row	<div>Numeric</div>	0 0 0 0 0 0 0 0 0
13	13row	<div>Numeric</div>	0 0 0 0 0 0 0 0 1
14	14row	<div>Numeric</div>	0 1 0 0 0 0 0 0 0

Figura 4. Opción "Parse" de H2O

Seguido de esto y de visualizar el fichero, se ha procedido a hacer la división del fichero en conjunto de entrenamiento y conjunto de validación.

H2O permite seleccionar el porcentaje que se quiere de división, fraccionando así el dataset en dos conjuntos de datos distintos: el conjunto de entrenamiento, que se encargará de proporcionar a la red neuronal los distintos ejemplos u observaciones de los que esta aprenderá patrones y características para realizar sus predicciones; y el conjunto de validación, que se ocupa de encontrar los mejores hiperparámetros posibles de la red neuronal para mejorar su rendimiento.



*Figura 5. Opción "Split" de H2O*

Una vez se han tenido los conjuntos de datos disponibles en la plataforma para que ésta los use, se ha creado el modelo.

H2O ofrece variedad en cuanto a los tipos de modelos de aprendizaje automático disponibles para implementar, como, por ejemplo, bosque de aislamiento (Isolation Forest) o modelos lineales generalizados (Generalized Linear Modeling).

En este caso, se ha seleccionado el modelo de aprendizaje profundo (Deep Learning), ya que este tipo de técnica es capaz de procesar grandes cantidades de datos y aprender de ellos para realizar diferentes tareas, como puede ser identificar patrones.

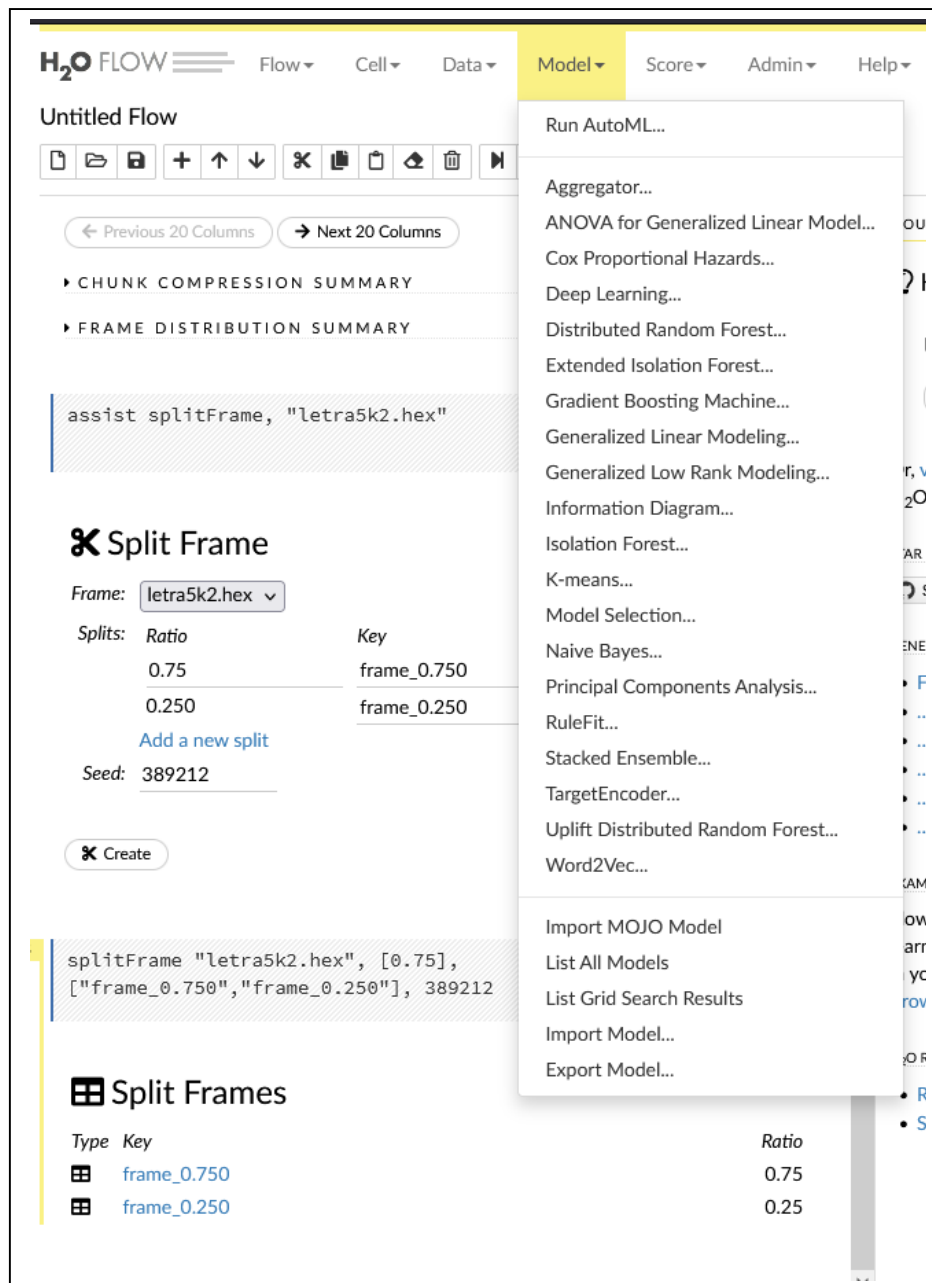


Figura 6. Menú de tipos de modelos disponibles en H2O

Para su creación, se han indicado los conjuntos de entrenamiento y validación anteriormente comentados, y en la opción “response\_column”, se ha indicado la columna “output”, ya que es ésta la que contiene las etiquetas con los valores reales, es decir, los que el modelo intentará predecir.

Por otro lado, la plataforma permite seleccionar gran cantidad de parámetros, de los cuáles se ha hecho uso principalmente de los siguientes:

- Hidden: indica las capas ocultas y el número de neuronas que se están introduciendo. Por ejemplo, si se introduce “200, 100”, el modelo tendrá 2 capas ocultas, la primera de 200 neuronas y la segunda de 100 neuronas.
- Epochs: es el número de veces que el modelo va a ser iterado durante el entrenamiento.

- **Loss:** especifica la función de pérdida a utilizar por el modelo, manera de cuantificar las diferencias entre las predicciones hechas por la red y los valores reales.

Todos los demás parámetros se mantendrán como los proporciona la plataforma por defecto. De entre todos estos, cabe destacar que se va a usar la función de activación “Rectifier”, también conocida como “ReLU”.

**Build a Model**

Select an algorithm: **Deep Learning**

**PARAMETERS**

<b>model_id</b>	deeplearning-74ed3fe8-0998-4ac7	Destination id for this model; auto-generated if not specified.																				
<b>training_frame</b>	frame_0.750	Id of the training data frame.																				
<b>validation_frame</b>	frame_0.250	Id of the validation data frame.																				
<b>nfolds</b>	0	Number of folds for K-fold cross-validation (0 to disable or >= 2).																				
<b>response_column</b>	output	Response variable column.																				
<b>ignored_columns</b>	Search... Showing page 1 of 4. -36 ignored. <table border="1"> <tr><td><input type="checkbox"/> 1row</td><td>INT</td></tr> <tr><td><input type="checkbox"/> 2row</td><td>INT</td></tr> <tr><td><input type="checkbox"/> 3row</td><td>INT</td></tr> <tr><td><input type="checkbox"/> 4row</td><td>INT</td></tr> <tr><td><input type="checkbox"/> 5row</td><td>INT</td></tr> <tr><td><input type="checkbox"/> 6row</td><td>INT</td></tr> <tr><td><input type="checkbox"/> 7row</td><td>INT</td></tr> <tr><td><input type="checkbox"/> 8row</td><td>INT</td></tr> <tr><td><input type="checkbox"/> 9row</td><td>INT</td></tr> <tr><td><input type="checkbox"/> 10row</td><td>INT</td></tr> </table> <div> <input checked="" type="checkbox"/> All    <input type="checkbox"/> None            <span>← Previous 10</span>    <span>→ Next 10</span> </div>		<input type="checkbox"/> 1row	INT	<input type="checkbox"/> 2row	INT	<input type="checkbox"/> 3row	INT	<input type="checkbox"/> 4row	INT	<input type="checkbox"/> 5row	INT	<input type="checkbox"/> 6row	INT	<input type="checkbox"/> 7row	INT	<input type="checkbox"/> 8row	INT	<input type="checkbox"/> 9row	INT	<input type="checkbox"/> 10row	INT
<input type="checkbox"/> 1row	INT																					
<input type="checkbox"/> 2row	INT																					
<input type="checkbox"/> 3row	INT																					
<input type="checkbox"/> 4row	INT																					
<input type="checkbox"/> 5row	INT																					
<input type="checkbox"/> 6row	INT																					
<input type="checkbox"/> 7row	INT																					
<input type="checkbox"/> 8row	INT																					
<input type="checkbox"/> 9row	INT																					
<input type="checkbox"/> 10row	INT																					
Only show columns with more than 0 % missing values.																						
<b>ignore_const_cols</b>	<input checked="" type="checkbox"/>	Ignore constant columns.																				
<b>activation</b>	Rectifier	Activation function.																				
<b>hidden</b>	200, 200	Hidden layer sizes (e.g. [100, 100]).																				

*Figura 7. Construcción de un modelo en H2O*

Las funciones de activación son funciones matemáticas que se aplican a los nodos o neuronas de la red, su propósito es transformar la señal de entrada que el nodo recibe, en una señal de salida.

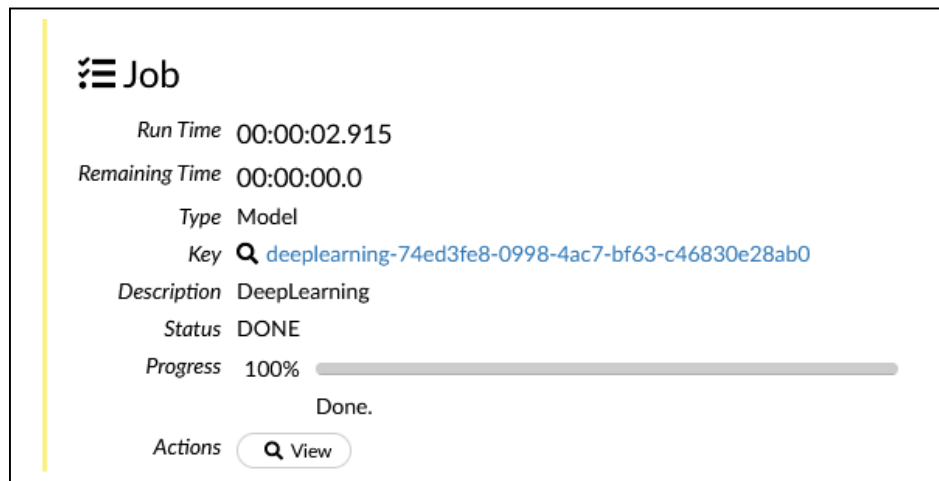
Estas funciones permiten aplicar la no linealidad a las redes neuronales, lo que permite que estas aprendan patrones complejos.

En este caso se ha usado la función “ReLU”, que se define como:

$$f(x) = \max(0, x)$$

Donde “x” es la entrada de la neurona.

A la hora de construir el modelo, la plataforma muestra una barra de progreso, una vez que ésta se ha completado, es cuando se puede visualizar el entrenamiento del modelo.



*Figura 8. Entrenamiento de un modelo en H2O*

A la hora de visualizar el entrenamiento del modelo, H2O ofrece distintas acciones a realizar sobre el modelo y gran cantidad de información, como, por ejemplo, gráficos con información sobre el “loss” e información sobre los parámetros y las variables.

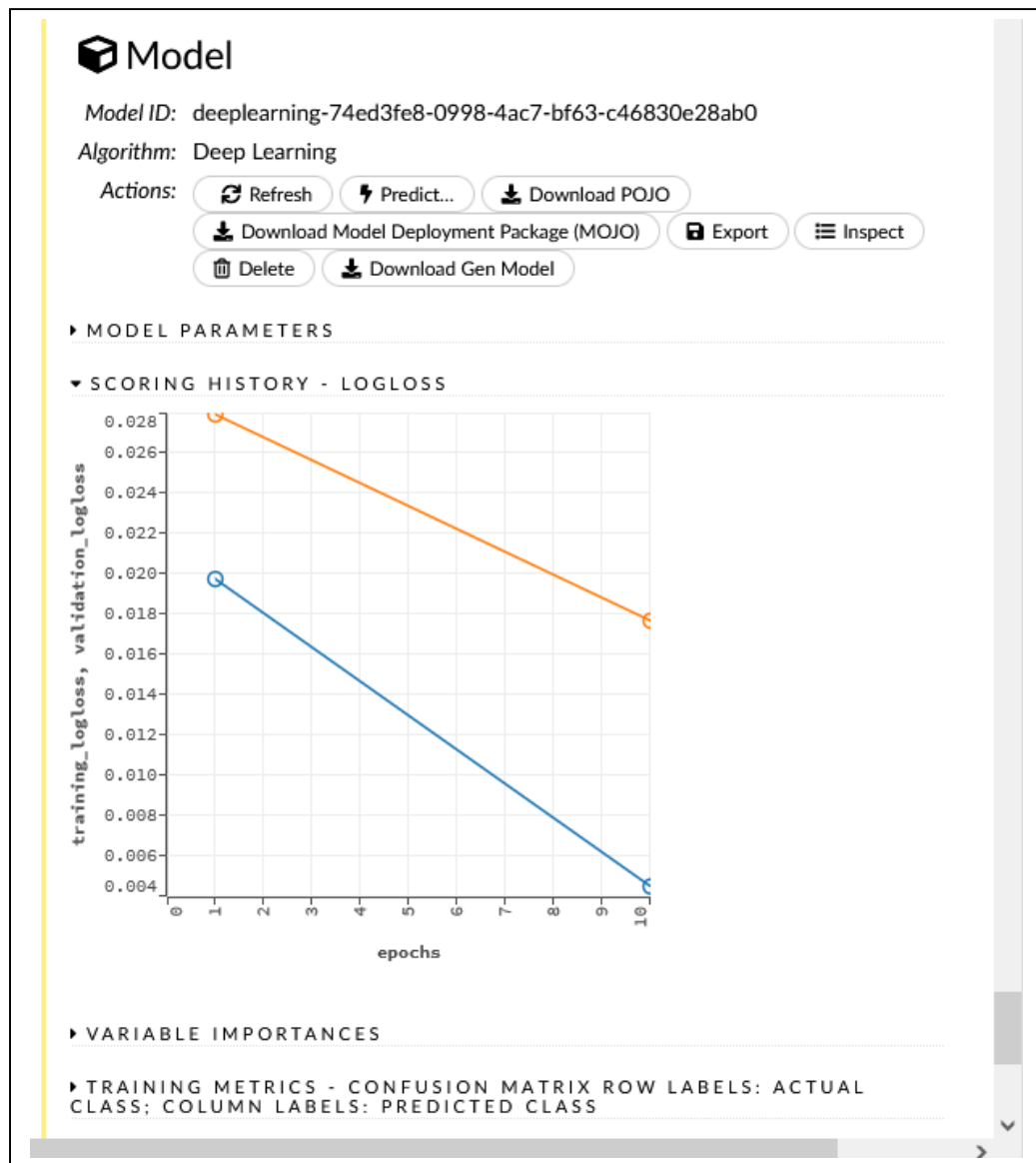
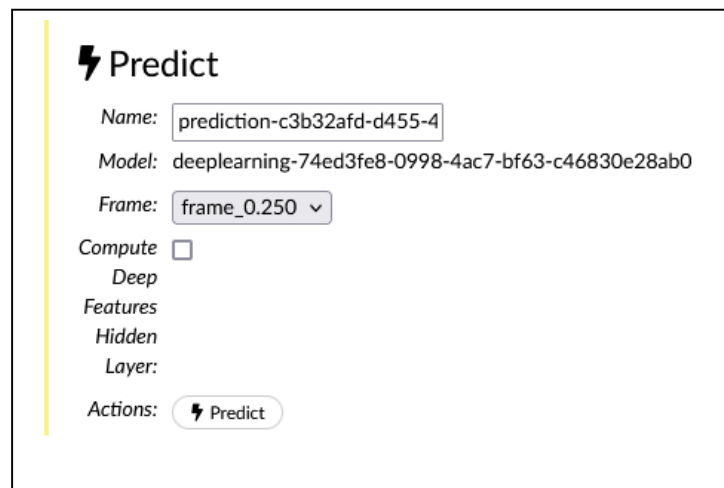


Figura 9. Resultados de un entrenamiento en H2O



Para efectuar las predicciones y comprobar que el modelo las realiza correctamente se ha utilizado la acción “Predict...”.

Además, se ha elegido el frame de validación sacado de la división de los datos comentada anteriormente, y se ha combinado con el frame principal.



**Predict**

Name:

Model: deeplearning-74ed3fe8-0998-4ac7-bf63-c46830e28ab0

Frame:

Compute ☐

Deep

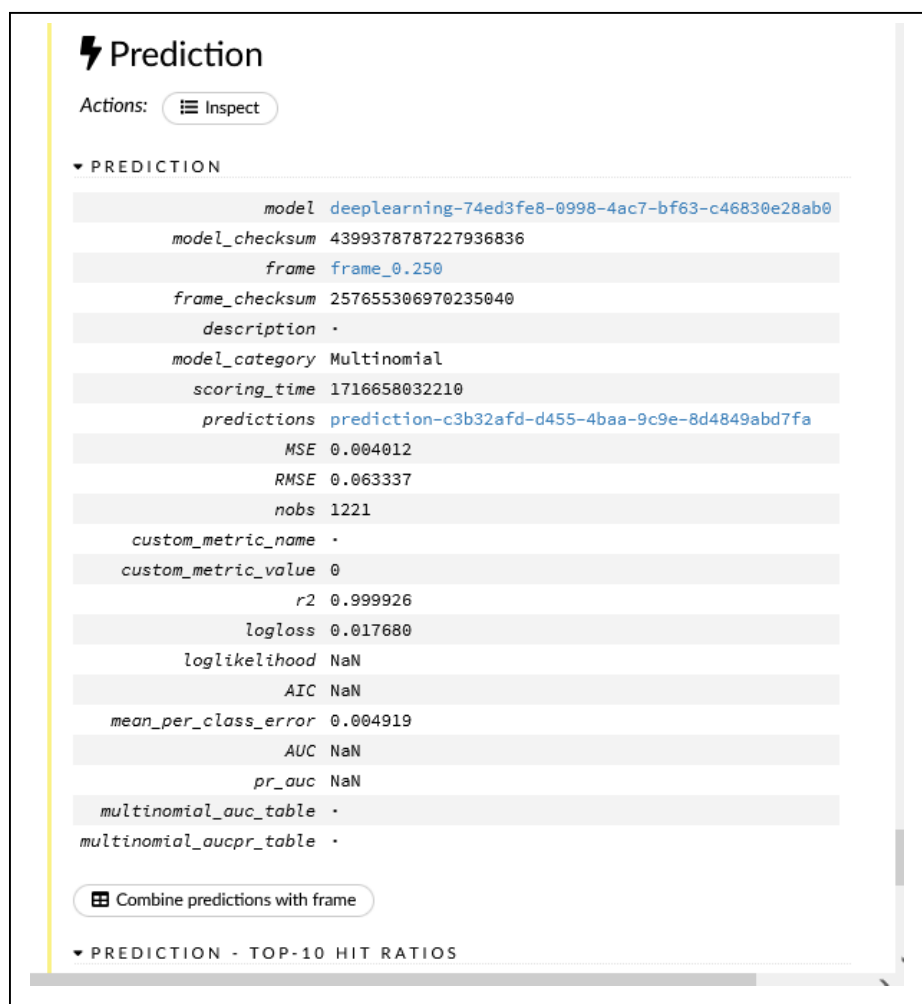
Features

Hidden

Layer:

Actions: ⚡ Predict

Figura 10. Opción de realizar una predicción en H2O



**Prediction**

Actions: Inspect

▼ PREDICTION

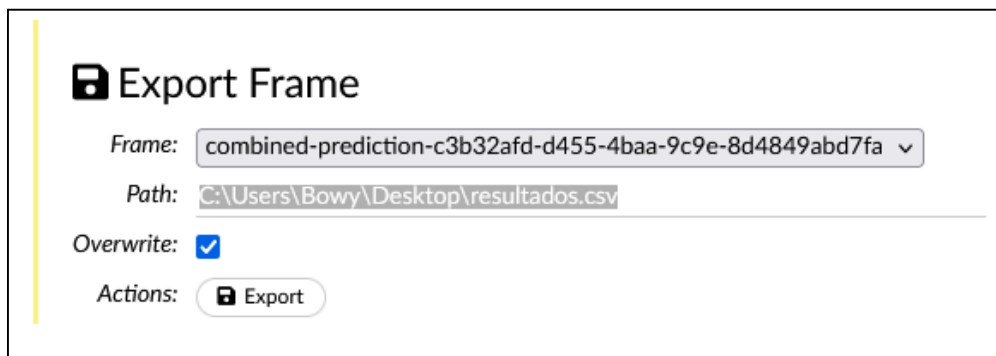
model	deeplearning-74ed3fe8-0998-4ac7-bf63-c46830e28ab0
model_checksum	4399378787227936836
frame	frame_0.250
frame_checksum	257655306970235040
description	-
model_category	Multinomial
scoring_time	1716658032210
predictions	prediction-c3b32afd-d455-4baa-9c9e-8d4849abd7fa
MSE	0.004012
RMSE	0.063337
nobs	1221
custom_metric_name	-
custom_metric_value	0
r2	0.999926
logloss	0.017680
loglikelihood	NaN
AIC	NaN
mean_per_class_error	0.004919
AUC	NaN
pr_auc	NaN
multinomial_auc_table	-
multinomial_aucpr_table	-

Combine predictions with frame

▼ PREDICTION - TOP-10 HIT RATIOS

Figura 11. Predicción realizada en H2O

Después de combinarlo, H2O Flow nos da la opción de exportar los resultados de la predicción, con los que, gracias al desarrollo de un script en Python que compara los valores predichos con los valores reales, se ha calculado la precisión de la predicción hecha por el modelo.



**Export Frame**

Frame:

Path:

Overwrite: ☒

Actions: Export

*Figura 12. Exportar frame en H2O*

```
1 #
2 # Víctor Bouhaben
3 # 25/05/2024
4 # Contexto: TFG Ingeniería Informática EUPT
5 #
6 # Programa para calcular la precisión de un modelo de H2O,
7 # compara la columna con el valor predicho con la columna con
8 # el valor real.
9 #
10
11 import pandas as pd
12
13 # Cargar el archivo CSV
14 df = pd.read_csv('C:\\Users\\Bowly\\Desktop\\resultados.csv')
15
16 columna1 = df['predict']
17 columna2 = df['output']
18
19 aciertos = 0
20
21 # Comparar las dos columnas
22 comparacion = columna1 == columna2
23 aciertos = comparacion.sum()
24
25 print("Precisión: " + str((aciertos / len(df) * 100)) + " %")
```

*Figura 13. Script para calcular la precisión de una predicción de H2O*