

PROCEDIMIENTO DE NUEVOS PROYECTOS		
 <b>TRANSMURAL</b> BIOTECH	Guía de utilización de la Plataforma	<b>Edición:</b> 1
		<b>Fecha:</b> 01-04-20
		<b>Hoja:</b> 1 de 29



Guía de utilización de la plataforma

## Contenido

1.	Introducción .....	4
2.	Selección del tipo de experimento.....	5
2.1.	Segmentación.....	5
2.2.	Clasificación.....	5
2.3.	Regresión.....	5
3.	Selección del tipo de métodos .....	6
4.	Selección del <i>dataset</i> de datos de imágenes .....	7
4.1.	Formatos de imagen .....	8
4.2.	Tamaños de imagen .....	8
4.3.	Análisis de color .....	8
4.4.	Etiquetas o <i>outcomes</i> clínicos .....	9
4.5.	Uso de ROIs .....	9
4.6.	Almacenamiento de la base de datos .....	10
4.7.	Visualización de la base de datos.....	10
5.	Selección del <i>dataset</i> de datos clínicos.....	10
5.1.	Formatos de datos .....	11
5.2.	Fichero de datos.....	11
5.3.	Almacenamiento de la base de datos .....	12
6.	Experimentos de predicción con <i>Machine Learning</i> .....	13
6.1.	Estructura algoritmos <i>Machine Learning</i> .....	13
Image Processing:	.....	13
LaunchPad:	.....	13
6.2.	Selección de conjunto de <i>ejecución</i> .....	14
6.3.	Consejos de ejecución .....	14
6.4.	Métodos de <i>Image Processing</i> .....	15
6.5.	Métodos de <i>LaunchPad</i> .....	15
6.6.	Optimización de resultados.....	16

7.	Experimentos de predicción con <i>Deep Learning</i> .....	17
7.1.	Selección de conjunto de <i>ejecución</i> .....	17
7.2.	Consejos de ejecución .....	17
7.3.	Optimización de resultados.....	17
8.	Herramienta de Segmentación ( <i>Deep Learning</i> ).....	17
8.1.	Selección de conjunto de <i>ejecución</i> .....	18
8.2.	Consejos de ejecución .....	18
8.3.	Optimización de resultados.....	18
9.	Validaciones del sistema .....	18
9.1.	General (imágenes) [GI] .....	19
9.2.	General (datos clínicos) [GD].....	19
9.3.	Predicción (clasificación o regresión).....	20
9.4.	Segmentación [CS] .....	21
10.	Visualización de resultados .....	21
10.1.	Clasificación Binaria.....	22
10.2.	Clasificación Múltiple .....	23
10.3.	Regresión.....	24
10.4.	Segmentación.....	25
10.5.	<i>Datasets</i> de datos clínicos.....	26
	Anexo: Parámetros configurables .....	27
A1.	Parámetros <i>Machine Learning</i> .....	27
A2.	Parámetros <i>Deep Learning</i> .....	28
A3.	Parámetros Segmentación ( <i>Deep Learning</i> ) .....	29

 <b>TRANSMURAL</b> BIOTECH	<b>Guía de Utilización de la Plataforma</b>	Rev.: 14	Fecha: 15-03-2022
		Página 4 de 29	

## 1. Introducción

Ésta guía es un manual de usuario para aquellas personas que quieran hacer uso de la plataforma.

Al visitar la página principal de la plataforma lo primero que encontramos es una pantalla como la que muestra la Ilustración 1, dónde debemos elegir qué acción queremos realizar. Las posibilidades son:

- 1) Entrenar segmentación: Entrenar algoritmos de segmentación de imágenes para delinear automáticamente las regiones de interés (ROIs) de cada una de ellas.
- 2) Crear algoritmo: Configurar métodos de inteligencia artificial para crear algoritmos de predicción.
- 3) Listado de algoritmos: Ver los algoritmos creados en 1) y 2) y analizar los resultados obtenidos de cada uno de sus métodos.
- 4) Gestión de imágenes: Gestionar los *datasets* de imágenes necesarios para realizar los experimentos de predicción o segmentación sobre ellos.
- 5) Gestión de datos: Gestionar los *datasets* de datos clínicos necesarios para realizar experimentos de predicción sobre ellos.



Ilustración 1. Página de bienvenida de la plataforma, que permite seleccionar la tarea que se quiere realizar.

A continuación, se analizan distintos aspectos a tener en cuenta para poder utilizar correctamente las herramientas de la plataforma, dando un conjunto de consejos útiles, sobre todo para aquellos usuarios menos expertos en inteligencia artificial. Al final del documento también se incluye un anexo con el listado de todos los parámetros configurables que contiene nuestra plataforma.

 <b>TRANSMURAL</b> BIOTECH	<b>Guía de Utilización de la Plataforma</b>	Rev.: 14	Fecha: 15-03-2022
		Página 5 de 29	

## 2. Selección del tipo de experimento

Primeramente, habrá que seleccionar el tipo de experimento en el que se quiere trabajar. Podremos seleccionar si queremos entrenar un segmentador (primer botón de la Ilustración 1) o configurar algoritmos de predicción (segundo botón de la Ilustración 1). En éste segundo caso, cuando se nos muestre el formulario a rellenar con los datos de configuración del algoritmo, se deberá escoger también si se quiere ejecutar un algoritmo de clasificación (binaria o múltiple) o de regresión.

### 2.1. Segmentación

La herramienta de segmentación la usaremos en proyectos donde se quiera segmentar o delinear de manera automática una o más regiones de interés (en adelante “ROIs”) de las imágenes.

Ejemplo: Tenemos imágenes de cerebro en las que se necesita segmentar de manera automática las diferentes estructuras visibles (cerebelo, tálamos, etc.) para ayudar al clínico en la visualización.

Para utilizar ésta herramienta se requiere de una base de datos de imágenes con las delineaciones de sus ROIs asociadas.

### 2.2. Clasificación

Experimento para proyectos donde se necesita predecir la clase o un valor categórico de una determinada patología. Puede ser una clasificación binaria (cuándo solamente hay 2 clases) o múltiple (cuando hay más de 2). Hay que tener en cuenta que no se permite la clasificación no excluyente (que una imagen/registro pueda pertenecer a más de una clase).

Ejemplo clasificación binaria: Tenemos imágenes de mamografías con lesiones y queremos ver si se podría hacer un diagnóstico automático (o dar una probabilidad) de si éstas son cancerígenas o no.

Ejemplo clasificación múltiple: Tenemos imágenes de lesiones de piel, y queremos detectar (o dar una probabilidad) de qué tipo de lesión es (lunar, melanoma, carcinoma maligno o carcinoma benigno) para ayudar en el diagnóstico del médico.

Para realizar éste tipo de experimentos se requiere de una base de datos de imágenes o de datos clínicos, las etiquetas o *outcomes* clínicos (*groundtruth*) asociados a cada una y opcionalmente las ROIs asociadas a cada imagen.

### 2.3. Regresión

Experimento para proyectos donde se necesita predecir un valor continuo de una patología, sin clases definidas. También puede ser útil probar este tipo de experimento en un proyecto de

 <b>TRANSMURAL</b> BIOTECH	<b>Guía de Utilización de la Plataforma</b>	Rev.: 14	Fecha: 15-03-2022
		Página 6 de 29	

clasificación multiclase cuando las clases guardan relación de grado entre ellas, para obtener un resultado distinto al que nos puede dar la Clasificación.

Ejemplo: Tenemos imágenes de lesiones tumorales que los clínicos puntúan con un grado del 1 al 10 según su gravedad y queremos observar si se podría hacer lo mismo de manera automática.

Para realizar éste tipo de experimentos se requiere de una base de datos de imágenes o de datos clínicos, las etiquetas o *outcomes* clínicos (*groundtruth*) asociados a cada una y opcionalmente las ROIs asociadas a cada imagen.

### 3. Selección del tipo de métodos

De haber escogido realizar un experimento de predicción (clasificación o regresión) mediante imágenes, a continuación habrá que decidir si se van a usar métodos de *Machine Learning* (ML) o de *Deep Learning* (DL) para ello. En el caso de utilizar un *dataset* de datos clínicos, solamente estará disponible mediante algoritmos de ML. En cambio, en el caso de realizar un experimento de segmentación de imágenes, sólo estará disponible mediante algoritmos de DL.

Tanto los algoritmos de ML como los de DL son algoritmos de inteligencia artificial que permiten a las máquinas analizar datos y evaluar el contexto para desarrollar tareas similares a las que lleva a cabo un ser humano.

Las técnicas de ML por un lado, permiten que las máquinas aprendan de los datos para lograr metas concretas como organizar datos o reconocer patrones. De ésta manera, las máquinas son entrenadas para aprender a ejecutar diferentes tareas de forma autónoma y cuando son expuestas a nuevos datos, se adaptan a partir de los cálculos anteriores y los patrones se moldean para ofrecer respuestas confiables.

Las técnicas de DL en cambio, son algoritmos jerárquicos de alto nivel que emulan el aprendizaje humano mediante redes neuronales artificiales. De esta manera, los datos son sometidos a varias capas de procesamiento no lineares que simulan la forma de pensar de las neuronas. Para ello, estos algoritmos requieren alimentarse de cantidades inmensas de datos para aprender.

De esta forma, la recomendación genérica es usar DL sólo cuando se dispone de una base de datos suficientemente grande (esto es, tener más de 500 imágenes aprox. en total). De tener menos, el sistema podría no entrenarse correctamente o crear un *overfitting* no deseado. Así pues, si la base de datos es menor, lo mejor es lanzar algoritmos de ML que requieren de menos datos de entrenamiento, y que dan unos resultados más entendibles.

Para hacernos una idea, la Ilustración 2 muestra una gráfica donde se observa la diferencia de resultados que puede obtener ML comparado con DL en relación a la cantidad de datos que se tienen para entrenar.

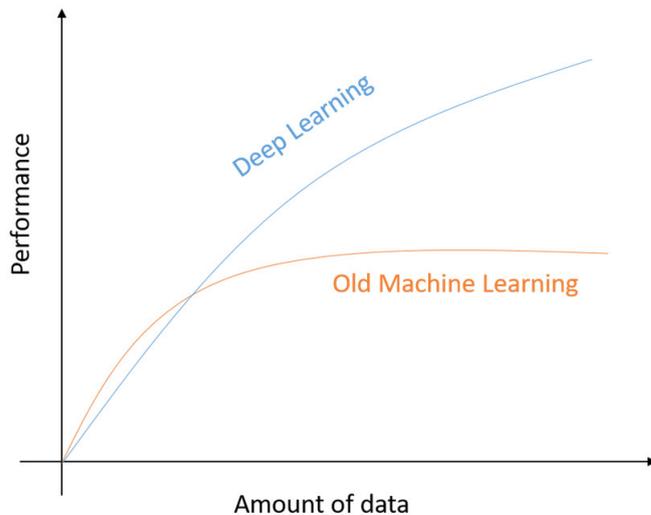


Ilustración 2. Gráfica que compara el rendimiento de los algoritmos de *Deep Learning* con los algoritmos de *Machine Learning* conforme aumenta la cantidad de datos usados.

Otra opción -sobre todo si no tenemos gran cantidad de datos- es usar primeramente algoritmos de ML que son más sencillos y suelen requerir menos tiempo de ejecución y posteriormente, probar de utilizar algoritmos de DL para buscar una optimización o un resultado distinto. Puede ser una buena opción sobre todo si con ML se ha observado un mejor resultado para algoritmos que usan *Learning* (como FVCNN), que puedan hacer pensar que algoritmos de DL también puedan dar buen resultado.

Por otro lado, si tenemos una base de datos de miles de imágenes, pero sin filtrar (con diferentes tamaños, distintas resoluciones, con falta de información, etc.) quizá usar DL a primeras puede que no sea la mejor opción ya que requerirá de un gran tiempo computacional y los resultados pueden ser no deseados o poco entendibles. Por esta razón, sería conveniente analizar la base de datos y obtener previamente un conjunto pequeño de imágenes con características similares que sea representativo del total y hacer unas pruebas con ML para ver si obtiene unos resultados mínimos. En el caso de ver resultados coherentes y satisfactorios, se puede plantear usar algoritmos de DL con un conjunto de datos mayor o ya con toda la base de datos.

#### **4. Selección del *dataset* de datos de imágenes**

Para realizar experimentos con imágenes, antes hay que analizar y seleccionar las que se van a usar y pretatarlas si fuera necesario. A continuación se detallan los formatos, tamaño, tipos de imagen, etc. que permite nuestra plataforma y dónde y cómo guardarlas.

 <b>TRANSMURAL</b> BIOTECH	<b>Guía de Utilización de la Plataforma</b>	Rev.: 14	Fecha: 15-03-2022
		Página 8 de 29	

#### 4.1. Formatos de imagen

Nuestra plataforma acepta 3 tipos distintos de formato de imagen: PNG, JPEG y DICOM. Se debe usar un único formato de imagen para el mismo experimento y si puede ser, que todas las imágenes sean parecidas entre sí (hay veces, que pese a estar en un mismo formato, las imágenes son muy diversas al haberlas adquirido con distintas máquinas, por ejemplo).

Por otro lado, se recomienda que las imágenes tengan la mínima información clínica de los pacientes posible. Para ello, se sugiere tener las cabeceras de las imágenes anonimizadas (o recortadas) y en el caso de estar en formato DICOM, también sus metadatos.

#### 4.2. Tamaños de imagen

No hay un tamaño específico que tengan que tener las imágenes, pero en general se recomienda usar resoluciones entorno a los 1200x1200 píxeles para obtener un buen rendimiento. De tener una resolución mayor, nuestra plataforma se encargará de escalarlas para poderlas procesar tanto con los algoritmos de ML como de DL.

Por el contrario, usar imágenes pequeñas de poca resolución puede hacer que los métodos (sobre todo los de procesado de imagen en ML) no obtengan el rendimiento deseado, por lo que no se recomienda usar imágenes de menos de 500x500 píxeles.

Por otro lado, se recomienda no tener imágenes de tamaños muy dispares, a no ser que esté justificado su uso. Esto podría dar más dificultades al sistema para obtener buenos resultados. Siempre puede ser útil recortar (sin perder calidad) todas las imágenes al mismo tamaño o similar, eliminando las partes de ella que no aportan información relevante (cabeceras, bordes, etc.).

#### 4.3. Análisis de color

No hay una modalidad de color en la que deben estar las imágenes para usarse en el sistema. Ahora bien, en ciertos casos podría ir mejor usar imágenes en escala de grises y en otros en color (ya sea RGB, HSV u otros). Hay que tener en cuenta que siempre necesitará más tiempo de computo usar imágenes a color con varios canales, que en escala de grises (con sólo un canal). Sea cual sea la modalidad de color, deberá ser siempre la misma para todas las imágenes del mismo experimento.

Si las imágenes que vamos a usar sabemos que lo más importante es su textura, sus bordes o sus cambios de intensidad, quizá lo más recomendable es que se transformen a escala de grises (siempre sin perder calidad) a no ser que ya lo estén. Éste sería el caso de imágenes ecográficas o rayos X, por ejemplo.

En cambio, si la información de color es relevante desde un punto de visto clínico, lo mejor es que las tengamos en color. Éste sería el caso de imágenes de histología o imágenes de lesiones de piel. En éste último caso por ejemplo, quizá podría ser conveniente transformarlas a

 <b>TRANSMURAL</b> BIOTECH	<b>Guía de Utilización de la Plataforma</b>	Rev.: 14	Fecha: 15-03-2022
		Página 9 de 29	

formato HSV en vez de RGB, ya que es un formato más representativo de como los humanos (en este caso los clínicos) perciben los colores y sus propiedades.

#### 4.4. Etiquetas o *outcomes* clínicos

Para poder realizar el entrenamiento de los algoritmos de predicción deberemos indicar las etiquetas o *outcomes* clínicos (*groundtruth*) que tiene cada una de las imágenes con las que realizaremos los experimentos. Estos datos deberán estar en un fichero a parte en formato CSV estándar (con separador de coma entre valores). Cada archivo deberá contener una columna que con los nombres de cada fichero de imagen y al lado otra con sus *outcomes*. Opcionalmente podrán tener una tercera columna con otro *outcome* distinto, si se quiere tener otro objetivo.

En el caso de clasificación binaria estos *outcomes* deberán ser “1s” para una clase (casos) y “0s” para la otra (controles). En el caso de la regresión se permiten números enteros o reales.

Para la clasificación múltiple se permitirá tanto usar números enteros (0, 1, 2, 3... y así consecutivamente hasta el número de clases -1), como textos alfanuméricos distintos para cada clase. Además, el máximo de clases posibles es de 15 y se pedirá el nombre de cada una de ellas al usuario.

#### 4.5. Uso de ROIs

La plataforma admite el uso de delineaciones de la región de interés (ROI) de cada imagen para realizar los experimentos de predicción (clasificación y regresión). Poder contar con esta información extra puede hacer que el sistema obtenga mejores resultados al no tener que fijarse en partes de la imagen que no contienen información útil. Ahora bien, esto no siempre será así, ya que quizá ciertos métodos puedan obtener información útil de partes de la imagen que al ojo humano pueden parecer inservibles.

Para poder utilizar esta funcionalidad, será suficiente con incluir ROIs para todas las imágenes del experimento y automáticamente se ejecutarán los experimentos con ellas. En caso de que no estén todas las ROIs, los experimentos siempre se lanzarán sin ROIs.

Las delineaciones de las ROIs deberán estar en formato CSV, JPG o PNG y sus valores deben ser de 0 la parte de fondo que no interesa (*background*) y un valor fijo mayor que 0 en la región de interés (ROI). Además, estos archivos deben tener exactamente el mismo tamaño de valores que las imágenes de las cuales hacen referencia en píxeles. Además, la ROI de cada imagen debe ser suficientemente grande (al menos 1000 píxeles de área) para no dificultar el correcto funcionamiento de la herramienta.

Cabe destacar, que la plataforma también cuenta con una herramienta para segmentar imágenes y así obtener ROIs de manera automática (capítulo 8). Esta funcionalidad permite crear algoritmos de segmentación y observar el rendimiento que obtendrían para segmentar

las ROIs de nuevas imágenes. Esto nos podría ayudar a crear un algoritmo previo que nos delinee las imágenes para así obtener un mejor resultado en la predicción.

#### 4.6. Almacenamiento de la base de datos

El conjunto de datos (tanto imágenes, como ROIs, como *outcomes*) que se quiera utilizar para los experimentos deberá estar almacenado en la base de datos de la plataforma. En el caso que no esté, deberá subirse en el apartado “gestión de imágenes”.

Todos estos datos, podrán subirse también como un archivo comprimido ZIP sin carpetas ni subcarpetas. La plataforma no admitirá archivos repetidos en la base de datos.

Una vez subidas, cada una de las imágenes tendrá una única etiqueta asociada, que será usada para seleccionarla cuando se quieran lanzar los experimentos.

#### 4.7. Visualización de la base de datos

En el apartado “gestión de imágenes” aparte de cargar nuevas imágenes podremos visualizar cada una de ellas tal como muestra la Ilustración 3. Además, podremos seleccionarlas individualmente para poder borrarlas, descargarlas, modificar sus etiquetas o sus *outcomes*.

TUS IMÁGENES (1 a 1000 de 25326) GESTIONAR ETIQUETAS    AÑADIR IMÁGENES    FILTRAR

	NOMBRE	ETIQUETA	ROI	OUTCOME	FECHA SUBIDA	ACCIONES	SELECCIÓN
	ISIC_0000000.jpg	Sin etiqueta	NO	NO	25/10/2019	 	<input checked="" type="checkbox"/>
	ISIC_0000007.jpg	Cerebro	SI	1	25/10/2019	 	<input checked="" type="checkbox"/>
	ISIC_0000008.jpg	Pulmón	SI	MEL	25/10/2019	 	<input checked="" type="checkbox"/>
	ISIC_0000009.jpg	Pulmon axial	SI	25	25/10/2019	 	<input checked="" type="checkbox"/>
	ISIC_0000010.jpg	Sin etiqueta	SI	MEL	25/10/2019	 	<input checked="" type="checkbox"/>
	ISIC_0000011.jpg	Pulmón	NO	NO	25/10/2019	 	<input checked="" type="checkbox"/>

Ilustración 3. Pantalla de gestión de imágenes, dónde se pueden ver cada una de las imágenes subidas a la plataforma.

### 5. Selección del *dataset* de datos clínicos

Además de utilizar imágenes para los experimentos, también tenemos la opción de usar únicamente datos clínicos. Estos pueden ser cualquier tipo de datos clínicos (resultados de análisis, de tests, medidas, etc.) que creamos que podrían contener información asociada a una patología.

Se pueden utilizar tantos datos como se desee y la herramienta realizará 2 tipos de pruebas: una con todos los datos que se proveen en el dataset, y una segunda con una optimización de

los mismos, informando en esta segunda qué datos ha considerado la plataforma que contienen información útil.

Antes de realizar los experimentos con un *dataset* de datos clínicos, hay que analizar y seleccionar los datos que se van a usar. Es responsabilidad del usuario seleccionar los datos que realmente se quieren usar, así como anonimizarlos si es necesario. A continuación se detalla el formato que permite nuestra plataforma y dónde y cómo guardarlos.

### 5.1. Formatos de datos

Nuestra plataforma acepta únicamente datos en formato Excel (xls o .xlsx). Cada fichero de Excel tiene que corresponder a un *dataset* distinto con el que se quiera realizar unos experimentos determinados. Todos los datos incluidos en el Excel serán utilizados para realizar los experimentos, si bien el sistema será capaz de analizar dichos datos y mostrar los resultados obtenidos con un subconjunto de los mismos, para optimizar la cantidad de datos necesarios.

### 5.2. Fichero de datos

Dentro del Excel los datos deben estar organizados de la siguiente manera:

patientID	outcome	Dato 1	Dato 2	Dato 3	...	Dato M
Valor 1, 1	Valor 2, 1	Valor 3, 1	Valor 4, 1	Valor 5, 1	Valor ..., 1	Valor M, 1
...	...	...	...	...	...	...
Valor 1, N	Valor 2, N	Valor 3, N	Valor 4, N	Valor 5, N	Valor ..., N	Valor M, N

Tabla 1. Tabla que representa cómo deben estar organizados los datos en el archivo Excel.

Dónde:

- Fila 1: fila de encabezado de datos. Esta fila es **obligatoria**. Indica los nombres de los distintos datos que conforman el dataset.
  - Hay 2 nombres reservados para los nombres de datos que deben estar en la columna 1 y 2 respectivamente, que son:
    - patientID. Este campo debe aparecer siempre en la primera columna del Excel. Puede estar vacío (todos los registros estar en blanco) o relleno (todos los registros con un valor alfanumérico), pero no puede estar en parte relleno y en parte no.
    - outcome. Este campo de debe aparecer siempre en la segunda columna del Excel. Todos sus registros deben contener con algún valor.
  - Resto de columnas:
    - Dato 1..M. Columnas con los distintos datos que va a contener el *dataset*. Los valores son de texto libre, pero no deben estar repetidos ni deben contener espacios o caracteres especiales (se permiten letras, números, espacio, guión, guión bajo y símbolos \$, % y &).

- Fila 2..N: filas de registros. Estas contendrán por cada fila, los valores correspondientes a cada dato identificado en la Fila 1, y el total de registros compondrá la N usada en la ejecución de los algoritmos.

NOTA: Es responsabilidad del usuario que los datos que contiene el fichero sean realmente los que quiere usar en los experimentos, así como que aquella información que necesita estar anonimizada lo esté, ya que el sistema no modificará ni añadirá ninguna información al *dataset* cargado.

Asimismo, todos los valores de los distintos datos deben estar rellenos, obviando el sistema de carga aquellos registros que contengan datos en blanco.

### 5.3. Almacenamiento de la base de datos

El conjunto de datos (archivo Excel) que se quiera utilizar para los experimentos deberá estar almacenado en la base de datos de la plataforma. En el caso que no esté, deberá subirse en el apartado “gestión de datos” (Ilustración 4).

TUS DATOS (4) AÑADIR DATASET

NOMBRE	Nº REGISTROS	ID PACIENTE	FECHA SUBIDA	Nº DATOS	ACCIONES
Dataset 1 - Prueba de nuevo	9	N	2020-09-11 08:49:40	11	   
DataSet 2 - Datos generales de obtención sencilla	9	S	2020-09-11 08:49:40	14	   
Microbioma	82	S	2020-09-23 11:03:09	24	   
Microbioma-con.error	81	S	2020-09-22 17:58:42	24	   

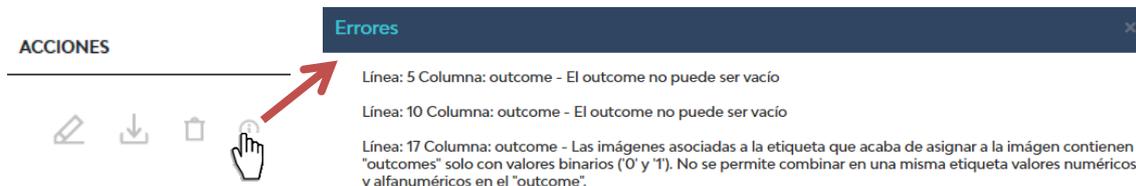
Ilustración 4. Pantalla de gestión de datos, dónde se pueden visualizar cada uno de los *datasets* de datos clínicos subidos a la plataforma.

Cada fichero de datos clínicos subido a la plataforma por el usuario se convertirá en un *dataset* distinto que podrá ser usado (si las validaciones son correctas) para la ejecución de algoritmos.

El usuario podrá recargar un *dataset* de nuevo las veces que quiera, tomándose el nuevo fichero como si fuera una carga completa, es decir, eliminándose los datos anteriores y cargando los nuevos que hay en el fichero.

Si se producen errores en la carga del *dataset* debido a problemas de formato del fichero de datos (errores en las cabeceras, columnas obligatorias inexistentes, etc.) el fichero no se podrá cargar y habrá que solucionarlos para poder subir el archivo. En cambio, si se producen errores en las filas de registros (valores vacíos, valores inválidos, etc.) la plataforma permitirá cargar el fichero pero invalidará dichas filas (por lo que no serán utilizadas para los experimentos). Igualmente, se informará de estos errores en el listado de *datasets* de datos clínicos a través de un enlace a tal efecto (Ilustración 5).

 <b>TRANSMURAL</b> BIOTECH	<b>Guía de Utilización de la Plataforma</b>	Rev.: 14	Fecha: 15-03-2022
		<b>Página 13 de 29</b>	



**Ilustración 5.** Vista de las acciones posibles a realizar en la pantalla “gestión de datos” y ejemplo de mensaje de error que se muestra al clicar el botón de información.

## 6. Experimentos de predicción con *Machine Learning*

A continuación se van a detallar varios aspectos a tener en cuenta si se han seleccionado métodos de *Machine Learning* para realizar los experimentos de predicción.

### 6.1. Estructura algoritmos *Machine Learning*

Hay que tener en cuenta que en ésta herramienta, los experimentos mediante *Machine Learning* constan de 2 partes claramente diferenciadas en el caso de experimentos con imágenes: *Image Processing* y *LaunchPad*. En el caso de experimentos con datos clínicos únicamente se contará con la segunda parte (*LaunchPad*).

#### **Image Processing:**

Primera fase de los experimentos con imágenes dónde se extraen las características de cada una de ellas. Los distintos métodos disponibles para esta parte, se encargan de extraer la información asociada de la imagen desde distintos puntos de vista (analizando la textura, la geometría, el tamaño de sus regiones, etc.). De ésta manera, cada método obtiene un conjunto de valores (características) distintos que serán los que utilizará el *LaunchPad* para realizar la predicción.

#### **LaunchPad:**

Segunda fase de los experimentos dónde cada uno de los distintos métodos disponibles recoge los valores de características obtenidos en *Image Processing* y realiza la predicción correspondiente para cada imagen. En el caso de usar datos clínicos en vez de imágenes, los valores de características serán directamente los valores de los registros de los *datasets*.

Para realizar la predicción, en una primera etapa de entrenamiento el algoritmo crea un modelo utilizando la clasificación introducida (la etiqueta o *outcome* clínico) para cada imagen/registro del conjunto de datos de *train*.

Posteriormente, en la segunda etapa, el algoritmo será capaz de predecir o clasificar las/los nuevas/os imágenes/registros del conjunto de *test* a partir del modelo creado. Con las predicciones finales se podrán dar unos resultados del rendimiento que obtiene el sistema comparándolas con los *outcomes* clínicos reales de cada imagen/registro de *test*.

## 6.2. Selección de conjunto de ejecución

Previo a la propia ejecución, hay que escoger el conjunto de datos que se va a usar para la ejecución del algoritmo. En función del tipo de Kfold escogido (Fijo o Kfold) el sistema repartirá los datos entre los conjuntos de entrenamiento (train) y test.

Siempre que la base de datos no sea muy grande, se aconseja utilizar el método de validación cruzada (*KFold*). Con éste método se obtiene un resultado estadísticamente más fiable del rendimiento del sistema. El valor de la "K" es el número de conjuntos en los que dividiremos nuestro *dataset* y el número de ejecuciones distintas que se lanzaran. Este valor puede variar, pero en general se recomienda usar un valor alrededor de 5-10. Hay que tener en cuenta que este método requiere más tiempo de computación que utilizar un conjunto fijo, y cuanto mayor sea la "K", más tiempo requerirá.

La otra opción es observar el rendimiento del sistema con un conjunto fijo (*Fixed*). Para ello, es aconsejable dejar un 70-80% de los datos para entrenar, y el resto para testear. Esto se fijará en los valores de porcentaje de train/test a definir en el formulario de ejecución.

Hay que tener en cuenta que la plataforma se encargará de utilizar las mismas prevalencias (porcentaje de casos-contróles o de cada una de las clases) o lo más similares posible, para el conjunto de *train* y para el de *test*. Al entrenar el sistema con una prevalencia similar a la del test (teóricamente la real), los resultados serán más favorables.

## 6.3. Consejos de ejecución

En las primeras pruebas que se realicen, se recomienda probar todos los métodos tanto del *Image Processing* como del *LaunchPad*, a no ser que tengamos claro cuáles usar o cuáles descartar por sus características. En los apartados 5.4 y 5.5 hay una descripción de cada uno de ellos.

Eso sí, en éstas primeras pruebas lo más recomendable es probar solamente los parámetros por defecto de cada método para obtener los resultados de manera más rápida y observar el rendimiento global de todos los métodos.

En pruebas posteriores se pueden añadir más parámetros de aquellos métodos que hayan resultado mejores en las primeras pruebas, para así obtener unos resultados más exhaustivos de éstos (Ilustración 5).

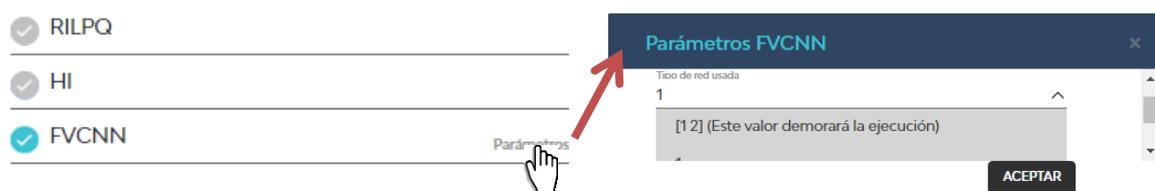


Ilustración 6. Vista de la selección de un método en concreto y cómo configurar los parámetros a ejecutar.

Por otro lado, en el caso de haber utilizado *K-Fold*, se recomienda empezar usando entre 3 y 5 repeticiones del *Search* (validación y búsqueda de mejores métodos) y entre 5 y 10 repeticiones del *Apply* (aplicación de los mejores métodos). Si se observa variabilidad en los resultados (observando su varianza), en posteriores pruebas se puede aumentar el número de repeticiones para obtener unos resultados más precisos.

#### 6.4. Métodos de *Image Processing*

La Tabla 1 muestra la descripción de los métodos de procesado de imagen (*Image Processing*) que se utilizan durante los experimentos de *Machine Learning*, que puede ser útil para saber cuándo escoger unos u otros:

Método	Descripción	Características
1) HOG	Descriptor usado para detección de objetos, que calcula la dirección del cambio de claro a oscuro en distintas ventanas de la imagen.	Insensible a cambios de iluminación en la imagen.
2) RILPQ	Variación del método LBP que tiene en cuenta la fase.	Insensible para imágenes más borrosas e invariante a los cambios de iluminación.
3) LBP	Descriptor usado para clasificación de texturas o detección de caras entre otros.	Robusto para variaciones en nivel de gris. Invariante a la rotación y traslación.
4) GLCM		
5) GIST		
6) HI		
7) SURF	Detector de los puntos de interés de una imagen invariantes al escalado.	Robusto a perturbaciones y transformaciones locales y globales que puede sufrir la imagen
8) Texton		
9) Cuboids		
10) FVCNN	Descriptor que utilizar Fisher Vector con redes neuronales convolucionales para incorporar así información de múltiples escalas y describir regiones de formas y tamaños arbitrarios.	

Tabla 2. Métodos ejecutables de *Image Processing*, con su descripción y características.

#### 6.5. Métodos de *LaunchPad*

La Tabla 2 muestra la descripción de los métodos de predicción del *LaunchPad* que se utilizan durante los experimentos de *Machine Learning*, que puede ser útil para saber cuándo escoger unos u otros:

Método	Descripción	Características
1) Boosting	Clasificador robusto creado a partir de la	

	combinación de clasificadores débiles.	
<b>2) Random Forest</b>	Combinación de árboles predictores aleatorios... Más simple para entrenar y ajustar que boosting...	Eficiente para bases de datos grandes?
<b>3) SVM</b>		
<b>4) KNN</b>		
<b>5) Discriminant Analysis</b>	Método que busca una combinación lineal de características entre dos o más clases.	
<b>6) Naive Bayes</b>	Clasificador probabilístico simple basado en la aplicación del teorema de Bayes que supone independencia entre características.	
<b>7) Ferns</b>		
<b>8) SPLS</b>		
<b>9) KNS</b>		
<b>10) RVM</b>		
<b>11) LogReg</b>		
<b>12) SetWise</b>		
<b>13) MLP</b>		

 Tabla 3. Métodos ejecutables de *LaunchPad* con su descripción y características.

## 6.6. Optimización de resultados

Una vez visualizados los resultados (ver capítulo 9) o incluso antes de hacer las ejecuciones, hay algunos parámetros a tener en cuenta que podrían hacer que los experimentos obtengan un mejor rendimiento.

Una de las formas de optimizar los resultados, puede ser elegir una métrica de visualización de resultados distinta (*Error Metric*). Hay varias opciones de métricas disponibles para visualizar los mejores resultados o para optimizar un valor u otro (en el Anexo se muestran todas). En general, se recomienda que si la patología / enfermedad que se está analizando es letal o que importa mucho detectar todos los casos positivos, se utilice el *Error Metric* de *Sensitivity*. De ésta manera, los resultados mostrados optimizaran el NPV y la Sensibilidad para no pasar por alto casos positivos. En cambio, si la patología no es letal y preferimos tener más equilibrio entre positivos-negativos, se recomienda usar la métrica F1 o F2-score que son las más justas.

Otra opción para obtener mejores resultados, o al menos resultados que sean más interesantes para nuestra aplicación, sería ajustar el FPR (ratio de falsos positivos). Deberíamos modificar el parámetro *maxFPR* de manera que se ajuste a la necesidad de nuestro proyecto. Es decir, si necesitamos que nuestra herramienta tenga una alta Especificidad para no pasar por alto los casos negativos, habrá que disminuir el *maxFPR* hasta encontrar un valor que nos satisfaga. Evidentemente como más pequeño sea éste valor mejor Especificidad vamos a obtener, en detrimento de la Sensibilidad de la herramienta.

 <b>TRANSMURAL</b> BIOTECH	<b>Guía de Utilización de la Plataforma</b>	Rev.: 14	Fecha: 15-03-2022
		Página <b>17</b> de <b>29</b>	

A parte de estos dos parámetros, las únicas recomendaciones que podrían mejorar los resultados obtenidos son, como ya se dijo en capítulos anteriores, cambiar la modalidad de color, añadir ROIs si no se han utilizado y en último lugar, conseguir un conjunto de imágenes mayor que permita entrenar mejor el sistema.

## 7. Experimentos de predicción con *Deep Learning*

A continuación se van a detallar varios aspectos a tener en cuenta si se han seleccionado métodos de *Deep Learning* para realizar los experimentos. Estos métodos solamente están disponibles para una base de datos de imágenes, no de datos clínicos.

### 7.1. Selección de conjunto de ejecución

En el caso de *Deep Learning*, el conjunto de entrenamiento y test deberá ser siempre fijo (*Fixed*). Al igual que se ha recomendado en el apartado de ML, es aconsejable dejar un 70-80% de los datos para entrenar (ver apartado 6.2), fijándose los valores deseados en el formulario de ejecución. La plataforma se encargará de tener la misma (o muy similar) prevalencia en el *train* y en el *test*.

### 7.2. Consejos de ejecución

Se recomienda siempre probar todos los métodos de DL disponibles, a no ser que ya hayamos comprobado que alguno de ellos no funciona bien en nuestro conjunto de datos. En éste caso no hay parámetros configurables para cada método, pero sí que hay algún parámetro que puede ser importante para obtener buenos resultados.

### 7.3. Optimización de resultados

Al igual que con los algoritmos de ML, los dos parámetros configurables que pueden mejorar nuestros resultados son el *ErrorMetric* y el *maxFPR*. Leer el apartado 5.6 para tener una información más exhaustiva sobre ellos.

Igualmente que en ML, a parte de estos dos parámetros, las únicas recomendaciones que podrían mejorar los resultados obtenidos son, cambiar la modalidad de color, añadir ROIs si no se han utilizado y en último lugar, conseguir un conjunto de imágenes mayor que permita entrenar mejor el sistema.

## 8. Herramienta de Segmentación (*Deep Learning*)

La herramienta de Segmentación nos permitirá obtener de manera automática la delineación de las regiones de interés (ROIs) de las imágenes. En éste caso, a diferencia de las delineaciones usadas para la predicción, se permiten varias ROIs distintas para cada imagen.

 <b>TRANSMURAL</b> BIOTECH	<b>Guía de Utilización de la Plataforma</b>	Rev.: 14	Fecha: 15-03-2022
		Página <b>18</b> de <b>29</b>	

Para utilizar la herramienta, se requerirá de imágenes previamente segmentadas, de las cuales, las usadas para el *train* servirán para entrenar el sistema y las usadas para el *test* servirán para obtener la delineación automática y analizar el rendimiento del sistema. No será necesario que cada imagen contenga todas las ROIs existentes, pero sí que tendrá que tener al menos la delineación de una de ellas.

Ésta herramienta, sólo está disponible mediante algoritmos de *Deep Learning* y hay que tener en cuenta que no se permite la segmentación no excluyente (que una región pueda pertenecer a más de una ROI al mismo tiempo).

### **8.1. Selección de conjunto de ejecución**

Para utilizar ésta herramienta también será necesario elegir el conjunto de datos a usar en la ejecución. Éste se dividirá automáticamente en los conjuntos de entrenamiento y test en función del porcentaje indicado en el formulario de ejecución. En éste caso no hay un porcentaje recomendado, pero hay que tener en cuenta que cómo mayor sea el conjunto de entrenamiento mejores serán los resultados sobre el conjunto de test.

### **8.2. Consejos de ejecución**

Se recomienda siempre probar todos los métodos existentes de DL para la segmentación. En éste caso no hay parámetros configurables para cada método, pero sí que hay algunos parámetros que pueden ser importantes para obtener buenos resultados de la segmentación.

Uno de los parámetros que puede ser útil configurar es el *Single ROI*. De activar éste parámetro todas las segmentaciones resultantes tendrán solamente una ROI de salida por imagen. Una vez activado se podrá configurar también el parámetro *Priority ROI*. Éste parámetro dará prioridad a la ROI que más nos interese: se podrá seleccionar “size” para que la ROI resultante sea la más grande o “center” si queremos que sea la ROI más grande centrada en la imagen.

### **8.3. Optimización de resultados**

En éste caso, no hay parámetros configurables que permitan tener una optimización de resultados, a parte de los comentados en el apartado anterior de *Single ROI* y *Priority ROI*. Por ésta razón, la única manera de mejorar los resultados es añadir más imágenes segmentadas al entrenamiento.

## **9. Validaciones del sistema**

Una vez configurados los algoritmos y experimentos, la herramienta realizará un conjunto de validaciones técnicas para evitar errores o resultados incorrectos durante la ejecución. De no cumplirse estas validaciones, el sistema enviará un mensaje de error que no permitirá seguir

 <b>TRANSMURAL</b> BIOTECH	<b>Guía de Utilización de la Plataforma</b>	Rev.: 14	Fecha: 15-03-2022
		Página <b>19</b> de <b>29</b>	

con la ejecución. Aparte, también habrá avisos para notificar al usuario de ciertas recomendaciones, aunque en este caso, permitirán ejecutar igualmente los experimentos pese a no cumplirlas.

A continuación se listan las validaciones y avisos que realizará el sistema, divididos por secciones, para que se tengan en cuenta a la hora de configurar y ejecutar los experimentos. Cada una de estas validaciones y avisos se darán en distintos sitios de la plataforma: en la pantalla “configurar segmentación” (indicado como “[CS]”), en la pantalla “configurar algoritmo” (indicado como “[CA]”), en la pantalla “gestión de imágenes” (indicado como “[GI]”) y en la pantalla “gestión de datos” (indicado como “[GD]”).

### 9.1. General (imágenes) [GI]

- Se validará que todas las imágenes de un experimento (misma etiqueta) sean en blanco y negro (1 canal) o que todas sean en color (3 canales).
- Se validará que todas las imágenes de un experimento (misma etiqueta) sean del mismo formato (PNG, JPG o DICOM).
- Se notificará con un aviso si los tamaños de las imágenes de un experimento (misma etiqueta) son muy dispares entre sí (más de un 50%).
- Se validará que en el fichero CSV del *outcome*, haya un máximo de 15 valores distintos (número de clases en una clasificación múltiple).
- Se validará que en la carga de *outcomes* desde fichero, los identificadores de cada una de las imágenes existan.
- Se validará que, tanto en la carga desde fichero como desde la edición de una imagen (ya sea al asignarle una nueva etiqueta como al modificar el *outcome*) el valor del *outcome* para la imagen (si está ya etiquetada) sea del mismo formato que el del resto de imágenes de esa misma etiqueta. Es decir, que si las imágenes de esa etiqueta tienen valores numéricos, no se permitirá que el *outcome* de la etiqueta sea alfanumérico, y a la inversa.
- Se validará que todas las Rois de un experimento (misma etiqueta) sean del mismo formato (PNG, JPG o CSV).

### 9.2. General (datos clínicos) [GD]

Tal como se explica en el apartado 5.3, habrá errores que no permiten la carga del fichero de datos clínicos y otros que simplemente descartaran la fila de datos incorrecta de la carga y por tanto, no se tendrán en cuenta para los experimentos.

#### No permiten la carga del fichero:

- Se validará que los nombres de los datos (valores en la fila 1) sean únicos y no lleven espacios ni caracteres especiales (se permiten letras, números, espacio, guión, guión bajo y símbolos \$, % y &).

 <b>TRANSMURAL</b> BIOTECH	<b>Guía de Utilización de la Plataforma</b>	Rev.: 14	Fecha: 15-03-2022
		Página <b>20</b> de <b>29</b>	

- Se validará que la primera y la segunda columna (*patientID* y *outcome*) existan y tengan el nombre correcto en la primera fila.

Permiten la carga del fichero (y notificarán de los errores):

- Se validará que todos los registros de la primera columna (*patientID*) estén igual que el primer registro (segunda fila). Esto es: si el primer registro está en blanco, se descartarán todos los registros rellenos; si el primer registro está relleno con un valor alfanumérico, se descartarán todos los registros en blanco.
- Se validará que todos los registros de la segunda columna (*outcome*) estén igual que el primer registro (segunda fila). Esto es: si el primer registro es un número (entero o decimal), se descartarán todos los registros con letras; si el primer registro contiene letras, se descartarán todos los registros únicamente numéricos.
- Se validará que en la segunda columna (*outcome*) no haya más de 15 tipos de registros distintos (número de clases en una clasificación múltiple) siempre y cuando los valores contenidos sean alfanuméricos. Si el *outcome* contiene valores únicamente numéricos, permite más de 15 valores distintos (para permitir la regresión).
- Se validará que los registros no contengan algún dato con valor vacío.
- Se validará que todos los valores de las columnas restantes a partir de la columna 2 no estén en blanco (0 es un valor válido). Si una fila contiene un valor en blanco, se descartará de la carga y se mostrará en el listado de errores.

### 9.3. Predicción (clasificación o regresión)

#### **Globales [CA]:**

- En el caso de clasificación binaria, se validará que los *outcomes* tengan únicamente 2 valores ("1" o "0").
- En el caso de clasificación multiclase, se validará que los *outcomes* sean todos textos alfanuméricos.
- En el caso de la regresión, se validará que los *outcomes* tengan valores de números enteros o reales.
- Se validará que se hayan introducido los nombres de todas las clases (etiquetas o *outcomes* clínicos).

#### **Machine Learning [CA]:**

- En ejecuciones de clasificación con *KFold*, se notificará con un aviso si hay menos de 100 imágenes/registros de datos totales.
- En ejecuciones de clasificación binaria con *KFold*, se notificará con un aviso si hay menos de 2\*K imágenes/registros de datos de casos positivos ("1") en el *outcome*.
- En ejecuciones de clasificación multiclase con *KFold*, se notificará con un aviso si hay menos de 2\*K imágenes/registros de datos de cada clase en el *outcome*.

 <b>TRANSMURAL</b> BIOTECH	<b>Guía de Utilización de la Plataforma</b>	Rev.: 14	Fecha: 15-03-2022
		Página <b>21</b> de <b>29</b>	

- En ejecuciones de regresión con *KFold*, se notificará con un aviso si hay menos de 20\*K imágenes/registros de datos totales en el *outcome*.
- En ejecuciones de clasificación con conjunto *Fixed*, se notificará con un aviso si hay menos de 70 imágenes/registros de datos de *train* y 30 de *test*.

#### **Deep Learning [CA]:**

- En ejecuciones de clasificación, se notificará con un aviso si hay menos de 500 imágenes de *train* y 150 de *test*.

#### **En caso de utilizar ROIs:**

- Se notificará con un aviso si algunas imágenes del experimento contienen ROIs y otras no. Si el usuario acepta, el experimento se lanzará sin ROIs [CA].
- Se validará que todas las ROIs tengan el mismo tamaño en píxeles que las imágenes asociadas [GI].
- Se validará que todas las ROIs tengan un mínimo de 1000 píxeles con valor distinto de cero [GI].
- Se validará que todas las Rois sean del mismo formato (PNG, JPG o CSV). No se permite continuar si es así.

#### **9.4. Segmentación [CS]**

- Se validará que todas las imágenes tengan una ROI asociada, y por tanto haya el mismo número de archivos ROIs que de imágenes.
- Se validará que todas las ROIs tengan un mínimo de 1000 píxeles con valor distinto de cero.
- Se notificará con un aviso si hay menos de 500 imágenes para *train* y 150 para *test*.
- Se validará que todas las Rois sean del mismo formato (PNG, JPG o CSV). No se permite continuar si es así.

### **10. Visualización de resultados**

En el apartado “Listado de Algoritmos” podremos visualizar las ejecuciones en curso, y una vez acabadas, visualizar los resultados obtenidos. Para cada tipo de experimento se obtiene una visualización de resultados distinta (Ilustración 6). Si clicamos en “Ver detalle” se observarán los resultados más detallados y divididos por métodos. A continuación se muestran distintos ejemplos del detalle de los resultados para diferentes ejecuciones.

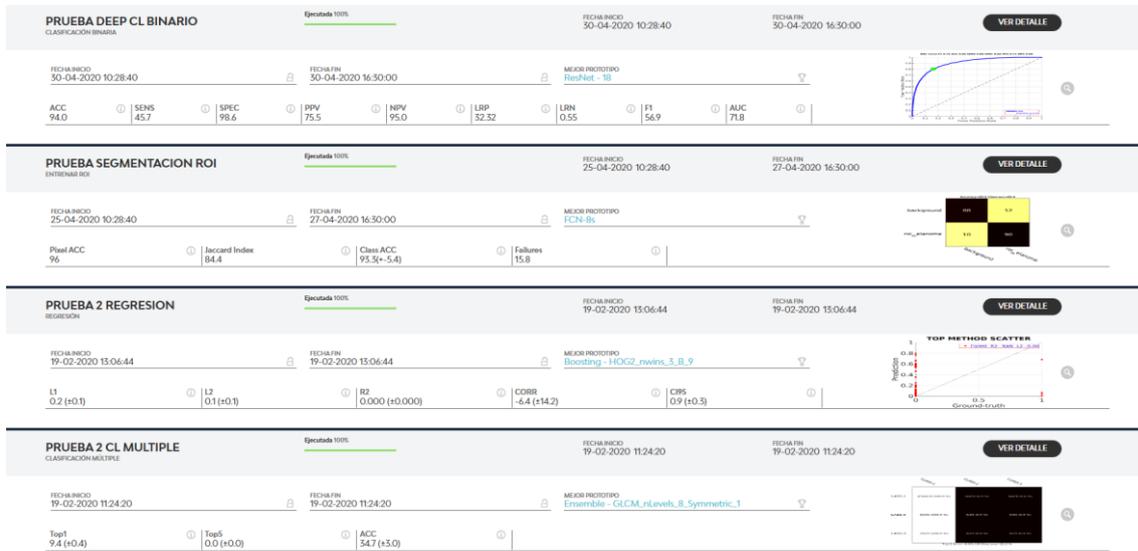


Ilustración 7. Vista global del apartado “Listado de Algoritmos” con ejemplos de 4 ejecuciones distintas.

## 10.1. Clasificación Binaria

Para un experimento de Clasificación Binaria se podrán observar los resultados obtenidos por cada método de DL o ML para las siguientes métricas estadísticas:

- **ACC (Accuracy):** Precisión global del algoritmo. Se define cómo el porcentaje de aciertos totales de entre todos los datos.  $\rightarrow ACC = (TP+TN)/(TP+FP+FN+TN)^*$
- **SENS (Sensibility):** Capacidad de la prueba para detectar la enfermedad en sujetos enfermos. Se define cómo el porcentaje de aciertos positivos de entre todos los positivos reales.  $\rightarrow SENS = TP/(TP+FN)^*$
- **SPEC (Specificity):** Capacidad de la prueba para detectar la ausencia de la enfermedad en sujetos sanos. Se define cómo el porcentaje de aciertos negativos de entre todos los negativos reales.  $\rightarrow SPEC = TN/(TN+FP)^*$
- **PPV (Positive Predictive Values):** Probabilidad de tener la enfermedad si el resultado de la prueba diagnóstica es positivo. Se define cómo el porcentaje de aciertos positivos de entre el total de resultados positivos.  $\rightarrow PPV = TP/(TP+FP)^*$
- **NPV (Negative Predictive Values):** Probabilidad de no tener la enfermedad si el resultado de la prueba diagnóstica es negativo. Se define cómo el porcentaje de aciertos negativos de entre el total de resultados negativos.  $\rightarrow NPV = TN/(TN+FN)^*$
- **LRP (Likelihood Ratio Positive):** Probabilidad de que una persona que tiene la enfermedad resulte positiva dividida por la probabilidad de que una persona que no tiene la enfermedad sea positiva.  $\rightarrow LRP = SENS/(1-SPEC)$

\* TP (True Positives) = Verdaderos positivos. TN (True Negatives) = Verdaderos negativos. FP (False Positives) = Falsos positivos. FN (False Negatives) = Falsos negativos.

- **LRN** (*Likelihood Ratio Negative*): Probabilidad de que una persona que tiene la enfermedad resulte negativa dividida por la probabilidad de que una persona que no tiene la enfermedad sea negativa.  $\rightarrow LRN = (1-SENS)/SPEC$
- **F1-score**: Medida de precisión que tiene un test que se considera como una media armónica que combina los valores de la precisión (PPV) y de la exhaustividad (SENS).  $\rightarrow F1 = 2*(PPV*SENS)/(PPV+SENS) = 2*TP/(2TP+FP+FN)*$
- **AUC** (*Area Under the Curve*): Probabilidad de que el modelo clasifique correctamente un positivo al azar por encima de un negativo al azar. Se obtiene calculando el área bajo la curva ROC.

Además, se observará también un gráfico de radar que permite comparar el resultado que obtiene cada método por cada métrica estadística de manera más gráfica y visual.

Finalmente, también se muestra la gráfica de la curva ROC, para los métodos que se quiera visualizar. Esta gráfica compara la Sensibilidad (eje vertical) frente a la 1-Especificidad (eje horizontal) según se varíe el umbral de discriminación y así ver los distintos resultados Sensibilidad-Especificidad que se podrían obtener con cada método.



Ilustración 8. Vista detallada de los resultados para un experimento de Clasificación Binaria.

## 10.2. Clasificación Múltiple

Para un experimento de Clasificación Múltiple se podrán observar los resultados obtenidos por cada método de DL o ML para las siguientes métricas estadísticas:

- **Top1**: Porcentaje de error global del algoritmo (con su desviación estándar) para todas las muestras cuando se asigna la clase más probable a cada una.
- **Top5**: Porcentaje de error global del algoritmo (con su desviación estándar) para todas las muestras cuando se asigna la quinta clase más probable a cada una (en el caso de existir).
- **ACC** (*Accuracy*): Media de la precisión de todas las clases (con su desviación estándar), sin tener en cuenta el número de casos de cada clase.

Además, se observará también un gráfico de radar que permite comparar el resultado que obtiene cada método por cada métrica estadística de manera más gráfica y visual.

Finalmente, también se muestra una matriz de confusión donde, para un método en concreto, se compara la clasificación predicha de todas las imágenes con la clasificación real para cada una de las clases.

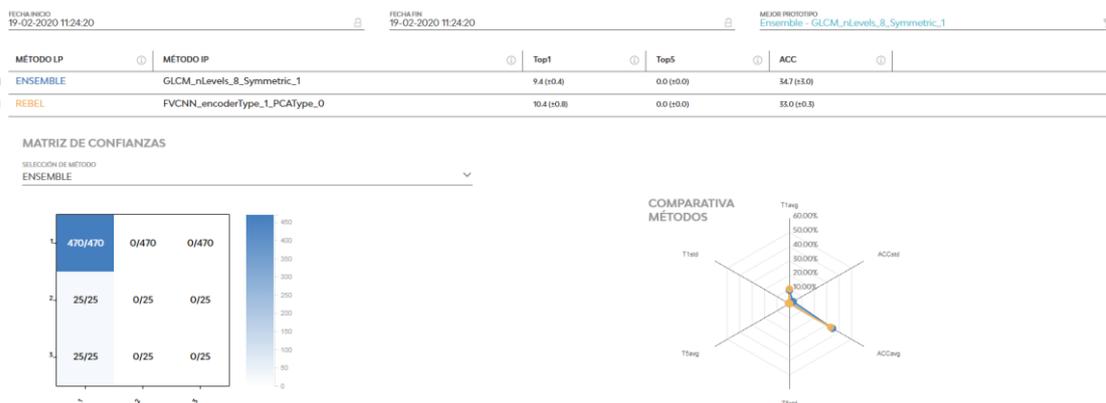


Ilustración 9. Vista detallada de los resultados para un experimento de Clasificación Múltiple.

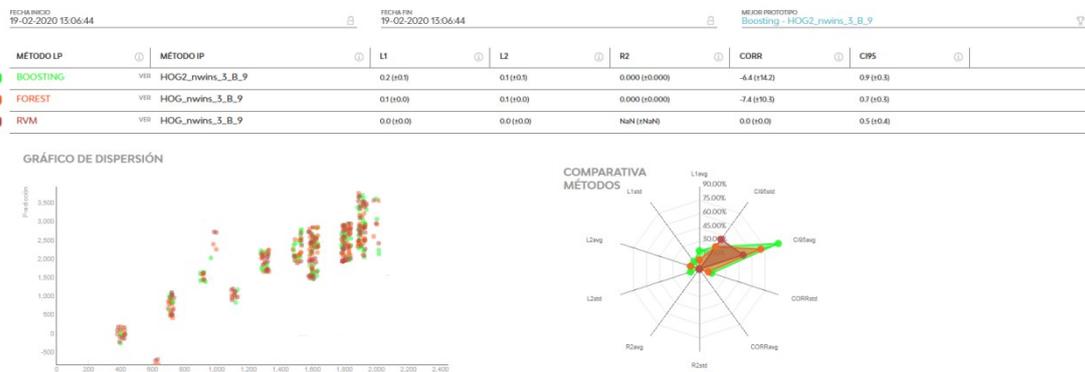
### 10.3. Regresión

Para un experimento de Regresión se podrán observar los resultados obtenidos por cada método de DL o ML para las siguientes métricas estadísticas:

- L1: Error cuadrático obtenido mediante una regularización Lasso (L1).
- L2: Error cuadrático obtenido mediante una regularización Ridge (L2).
- R2: Coeficiente de determinación que determina la calidad del modelo para replicar los resultados.
- Corr: Medida de correlación entre los valores reales y los valores predichos.
- CI95: Intervalo de confianza del 95% entre el valor real y el valor predicho.

Además, se observará también un gráfico de radar que permite comparar el resultado que obtiene cada método por cada métrica estadística de manera más gráfica y visual.

Finalmente, también se muestra un gráfico de dispersión para comparar los valores predichos de cada método con los valores reales.


**Ilustración 10**

Vista detallada de los resultados para un experimento de Regresión.

## 10.4. Segmentación

Para un experimento de Segmentación se podrán observar los resultados obtenidos por cada método de DL para las siguientes métricas estadísticas:

- Pixel ACC: Porcentaje de precisión en la segmentación de los píxeles. Esto es, el porcentaje de píxeles predichos de cada una de las ROIs en relación con los píxeles reales de cada ROI.
- Jaccard Index: Porcentaje de similitud entre las ROIs predichas y las ROIs reales. Se define como la cantidad de píxeles de la intersección de ambos conjuntos dividida por la cantidad de píxeles de su unión.
- Class ACC: Media de la precisión (con su desviación estándar) obtenida para todas las ROIs, sin tener en cuenta el número de píxeles de cada ROI.
- Failures: Porcentaje de imágenes que no superan el 50% del Jaccard Index.

Además, se observará también un gráfico de radar que permite comparar el resultado que obtiene cada método por cada métrica estadística de manera más gráfica y visual.

Finalmente también se muestra una matriz de confusión donde se compara el porcentaje de píxeles de cada una de las ROIs y el *background* reales con el porcentaje de píxeles de cada una de las ROIs y *background* seleccionados por el algoritmo que queramos analizar.



Ilustración 11. Vista detallada de los resultados para un experimento de Segmentación.

### 10.5. *Datasets* de datos clínicos

En el caso de haber realizado experimentos con un *dataset* de datos clínicos, los detalles de los resultados serán iguales que los descritos en los apartados anteriores, pero para 2 experimentos distintos.

Primeramente aparecerán los resultados para todas las variables proporcionadas por el usuario (columnas del Excel), y posteriormente los resultados para una optimización creada por el sistema con las mejores variables o combinaciones de las mismas.

En el caso de los resultados de la optimización de variables, el botón mostrado en la Ilustración 12 permitirá visualizar cuáles son las variables útiles con las que se quedó el sistema para ese experimento.

 <b>TRANSMURAL</b> BIOTECH	<b>Guía de Utilización de la Plataforma</b>	Rev.: 14	Fecha: 15-03-2022
		Página <b>27</b> de <b>29</b>	

## Anexo: Parámetros configurables

### A1. Parámetros *Machine Learning*

- **Tipo de experimento:**
  - Clasificación Binaria: Para proyectos con patologías de caso-control o de 2 clases.
  - Clasificación Múltiple: Para proyectos con patologías de más de 2 clases.
  - Regresión: para proyectos con datos de valor continuo, sin clases.
  
- **Configuración del conjunto de datos:**
  - K-Fold: Cuando se quiere testear el sistema con una validación cruzada de los datos (ver *Cross Validation* o *K-Fold*).
  - Fixed: Cuando se quiere testear el sistema con un conjunto fijo de entrenamiento y uno de test.
  
- **Métodos de *Image Processing***: Escoger los métodos de extracción de características de la imagen que se quieran utilizar y con qué parámetros.
  
- **Métodos del *LaunchPad***: escoger los métodos de clasificación/regresión que se quieran utilizar y con qué parámetros.
  
- **K** (sólo para K-Fold): Número de grupos “k” a dividir en la validación cruzada.
  
- **Repeticiones Search / Apply** (sólo para K-Fold): Número de repeticiones aleatorias a realizar de la validación cruzada en la etapa de “Search” o “Apply”.
  
- **maxFPR** (sólo para Clasificación Binaria): Ratio de falsos positivos (FPR) que el proyecto tolera. Buscará el mejor modelo para éste valor y mostrará los mejores resultados por debajo de éste punto de la curva ROC.
  
- **Error Metric**: Métrica que se quiere usar para mostrar los mejores resultados. Cada tipo de experimento tendrá unas métricas u otras disponibles:
  - Clasificación Binaria:
    - F1 score: Calcula la media de la *Precision* y *Recall*.
    - F2 score: Similar al F1 pero da más importancia al *Recall* (útil para proyectos donde es más importante clasificar correctamente tantas muestras positivas como sea posible).
    - Sensitivity: Da más importancia al NPV y la Sensibilidad (útil para proyectos dónde no se quiere pasar por alto ningún caso positivo).
    - DOR (*Diagnostic Odds Ratio*): Métrica que suele ser familiar para el personal clínico, que mide la efectividad de una prueba independiente de la prevalencia que tenga.

 <b>TRANSMURAL</b> BIOTECH	<b>Guía de Utilización de la Plataforma</b>	Rev.: 14	Fecha: 15-03-2022
		Página <b>28</b> de <b>29</b>	

- Accuracy: Indicador de la exactitud global de la prueba pero que depende directamente de la prevalencia o sesgo de los datos.
- Youden Index: informa del rendimiento de una prueba de diagnóstico en forma dicotómica. Puede usarse como criterio para seleccionar el valor de corte óptimo de la ROC, dando igual importancia a la Sensibilidad y a la Especificidad.
- Clasificación Multiclase:
  - Top-1 error: Optimiza el error de la case más probable.
  - Top 2-error: Optimiza el error de las 2 clases más probables.
  - Diagonal average: Optimiza la media de los valores de la diagonal de la matriz de confusión entre clases.
- Regresión:
  - L1: Optimiza el error para la regularización de Lasso (L1). Útil cuando sospechemos que varias características son irrelevantes.
  - L2: Optimiza el error para la regularización de Ridge (L2). Útil cuando sospechemos que varias características están correlacionadas entre ellas.
  - R2: Optimiza el coeficiente de determinación que determina la calidad del modelo para replicar los resultados.
  - Corr:
- **Balance Classes** (sólo para Clasificación Múltiple): De estar activado (1), creará una función de optimización que balanceará las clases en el test.

## A2. Parámetros *Deep Learning*

- **Tipo de experimento**:
  - Clasificación Binaria: Para proyectos con patologías de caso-control o de 2 clases.
  - Clasificación Múltiple: Para proyectos con patologías de más de 2 clases.
  - Regresión: para proyectos con datos de valor continuo, sin clases.
- **Métodos de *Deep Learning***: Escoger los métodos de *Deep Learning* que se quieran utilizar y con qué parámetros.
- **maxFPR** (sólo para Clasificación Binaria): Igual que en *Machine Learning*, es el FPR que el proyecto tolera. En este caso no buscará el mejor modelo para éste valor pero sí que mostrará los mejores resultados por debajo de éste punto de la curva ROC.
- **Error Metric**: Igual que en *Machine Learning*, es la métrica que se quiere usar para mostrar los mejores resultados. Hay las mismas opciones que en *Machine Learning* para cada tipo de experimento.
- **Balance Classes** (sólo para Clasificación Múltiple): De estar activado (1), creará una función de optimización que balanceará las clases en el test.

 <b>TRANSMURAL</b> BIOTECH	<b>Guía de Utilización de la Plataforma</b>	Rev.: 14	Fecha: 15-03-2022
		Página <b>29</b> de <b>29</b>	

### **A3. Parámetros Segmentación (*Deep Learning*)**

- **Single ROI:** De estar activado (1), para cada imagen sólo se obtendrá una ROI de salida.
  
- **Priority ROI:** Siempre que esté activado el parámetro anterior “Single ROI”, se podrá escoger qué ROI darle prioridad (“center” la más centrada o “size” la de mayor tamaño).