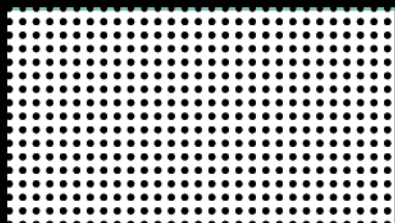


ACADEMIA IBERIA SUMMIT 2020:

Machine Learning
con InterSystems IRIS



 **InterSystems®**
Health | Business | Government

Table of Contents

Introducción	2
Escenario ML - Paciente no se presenta a Cita Médica	2
Ejercicio 1: Arrancar el entorno de la Academia y Explorar los Datos	3
Ejercicio 2: Trabajo Data Scientist desde Jupyter Notebook	5
Ejercicio 3: exportar algoritmo PMML.....	8
Ejercicio 4: Operacionalizar Algoritmo	9
Ejercicio 5: Uso de los Gateway Python y R desde la capa de Interoperabilidad	14
Ejercicio 6: IntegratedML.....	20
Anexo 1: Referencias	22
Anexo 2: Contenidos adicionales.....	23
Anexo 3: Instalación y Configuración.....	24
Instalación de pre-requisitos.....	25
cliente git	25
Git en Linux (ejemplo Amazon Linux 2)	25
Docker y docker-compose (ejemplo Amazon Linux 2).....	25
Descarga e Instalación del proyecto los contenedores MLAcademy	25
Instalación de Dbeaver + InterSystems IRIS JDBC	26
Prueba de Verificación de la instalación	29

Introducción

Esta academia muestra cómo se puede trabajar con tecnologías de Aprendizaje máquina (Machine Learning o ML) en conjunción con la plataforma de datos IRIS, y operacionalizar (poner en producción) algoritmos ML sobre la plataforma para su uso en tiempo real.

Los Ejercicios se dividen en 3 partes para mostrar 3 aspectos fundamentales de la plataforma:

- Acceder a IRIS desde entornos ML
 - Python con Jupyter Notebook: Ejercicios 1 & 2
- Operacionalizar algoritmos de ML
 - PMML: Ejercicios 3 & 4
 - El Python Gateway: Ejercicio 5
- IntegratedML: Machine Learning Automatizado desde SQL
 - Ejercicio 6

Escenario ML - Paciente no se presenta a Cita Médica

Este escenario publicado en el repositorio Kaggle consiste en un conjunto de 110,000 citas médicas en Vitória, Espírito santo, Brazil. Se intenta determinar la presencia/ausencia del Paciente en su cita, en función de 14 parámetros:

parametro	tipo	descripción
PatientID		Identificador de paciente - Eliminado
AppointmentID		Identificador de cita - Eliminado
Gender		Genero "M" o "F"
ScheduledDay		Fecha y Hora de la petición de Cita
AppointmentDay		Fecha de la Cita médica
Age		Edad del paciente
Neighborhood		Barrio de residencia
Scholarship		Indicador de Ingresos bajos (becas de estudios)
Hipertension		Hipertensión diagnosticada (si/no)
Diabetes		Diabetes diagnosticado (si/no)
Alcoholism		Alcoholismo diagnosticado (si/no)

parametro	tipo	descripción
Handcap		Nivel de Minusvalía (0..3)
SMS_received		Ha recibido SMS de recordatorio
No-Show		Variable a predecir (si/no)

Ejercicio 1: Arrancar el entorno de la Academia y Explorar los Datos

- Tarea 1: Desde la línea de comando, ir al directorio de instalación del proyecto. (C:\irismlacademy)

```
cd c:\irismlacademy
```

```
docker-compose up
```

- Tarea 2: Con el Browser (Chrome), abrir 1 pestaña con las siguiente URL, para llega al portal de gestión de IRIS:

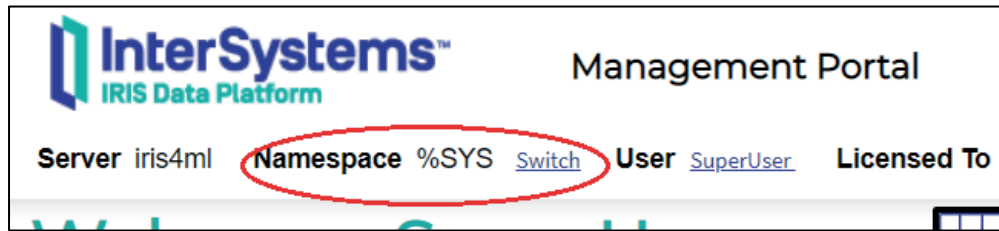
```
http://localhost.52773/csp/sys/UtilHome.csp
```

Hacer el Login con

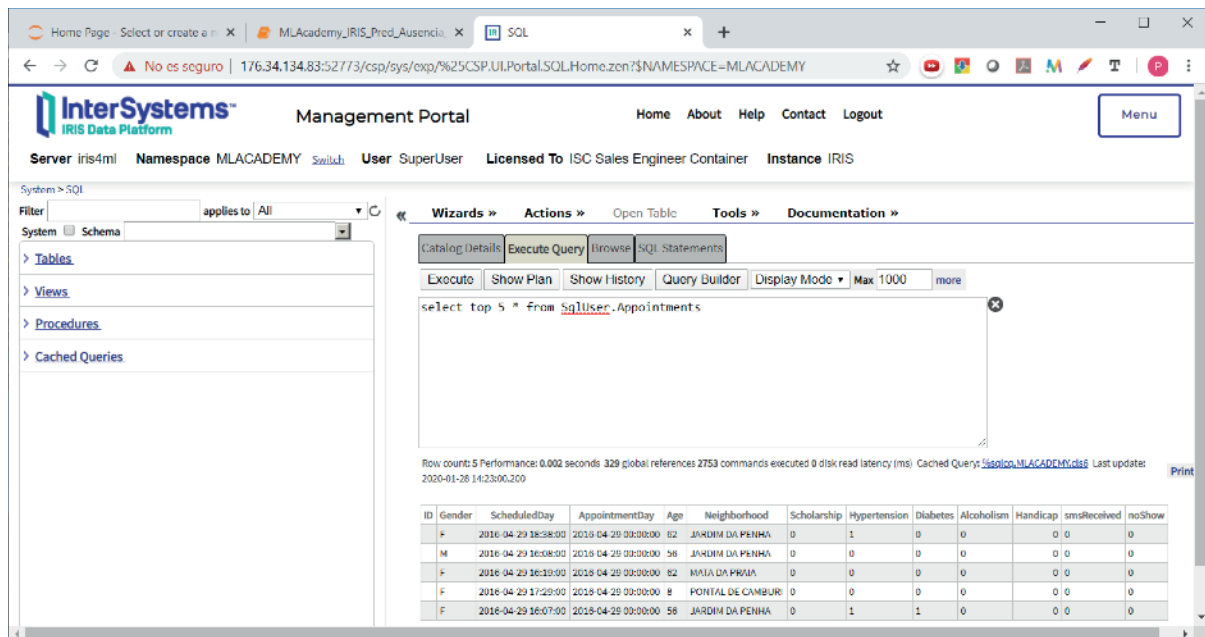
usuario	SuperUser
passord	sys

Y Navegar a "System Explorer" / "SQL" para ejecutar queries SQL en la Base de Datos MLACADEMY (Una alternativa es usar DBeaver configurado para conectarse a la Base de Datos correspondiente):

- Hacer Click en "Switch", y escoger "MLACADEMY"



- Hacer click en "System Explorer" y "SQL" par obtener la pantalla SQL



Mirar el formato de la Tabla a analizar

```
select top 10 * from SQLUser.appointments
```

InterSystems IRIS Añade proyecciones llamadas "Abstract Base Tables" para facilitar el trabajo de ML. Las ABT pueden tener campos ya pre-calculados, tratados y limpiados, y realizar un "One Hot Encoding".

```
select top 10 * from PublishedABT.MLSimpleAppointmentsGetFeatures()
```

Mirar el contenido de la Proyección "MLSimpleAppointmentsGetFeatures()":

columna	comentario
Delay	Tiempo de espera para la cita (AppointmentDay - ScheduledDay)
Gender_M	columna Gender transformada en 2 columnas (true si es Masculino)

This document and the information disclosed herein are confidential and proprietary to InterSystems. This information may not be used or disclosed to any third party for any purpose other than that specifically authorised in writing by InterSystems

columna	comentario
Gender_F	columna Gender codificada "One-Hot" para Femenino

Nota: La definición de las columnas contenidas en la ABT se ha realizado mediante la definición de un cubo de Analytics.

Mirar el contenido de "PublishedABT.AppointmentGetFeatures()"

- Notar la presencia de columnas agrupadas "AgeGroup" y codificadas "One-Hot"
- Que columnas adicionales han sido codificadas en modo "One-Hot" ?

Ejercicio 2: Trabajo Data Scientist desde Jupyter Notebook

- Tarea1: Con el browser (Chrome), abrir otra pestaña, y abrir el Entorno de desarrollo ML (jupyter Notebook) en la siguiente URL:

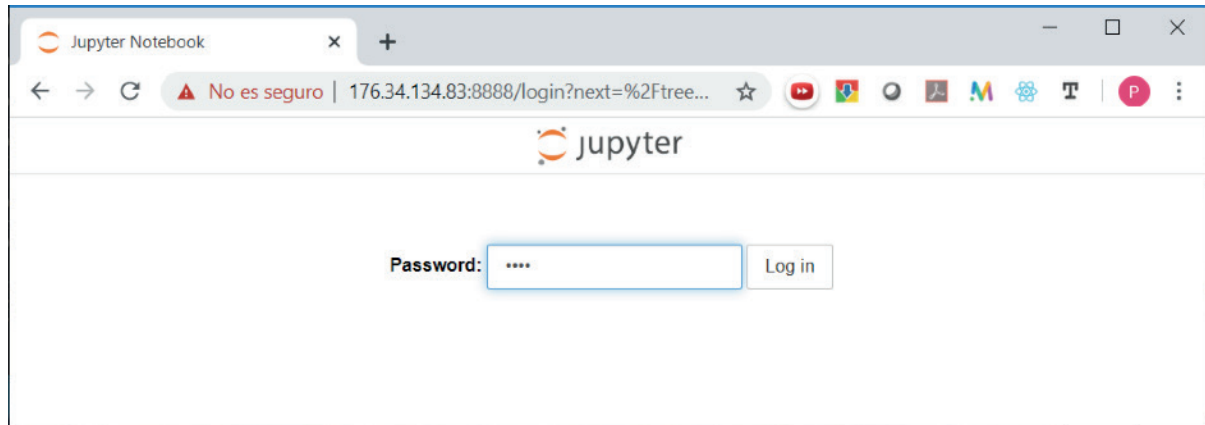
<http://localhost:8888>

Hacer el login con:

password

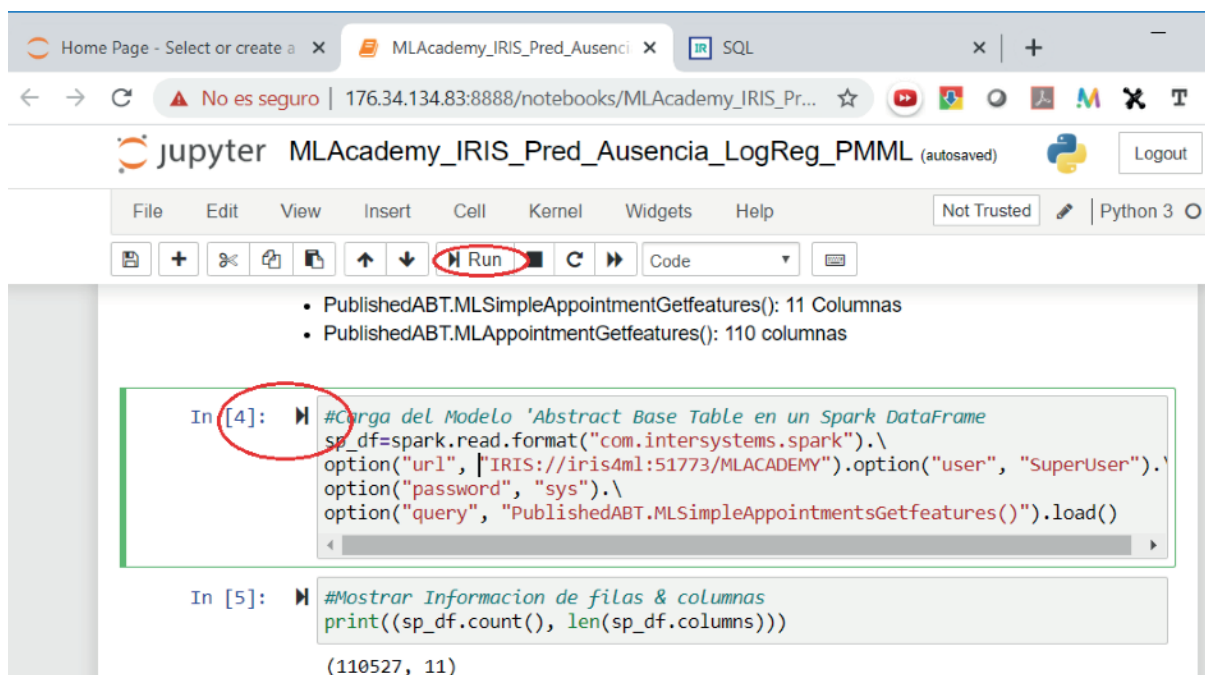
IRIS

La pantalla de Login:



Cargar el Notebook: MLAcademy_IRIS_Pred_Ausencia_LogReg_PMML.ipynb

- **Tarea 2:** Ejecutar las Celdas una a una en orden, leyendo las explicaciones, hasta llegar al apartado "Eportación a PMML". Para ejecutar una Celda Puede posicionar el cursor encima y hacer click en en botón "Run" de la barra de tareas o sobre el icono al lado de una celda:



Nota: Para trabajar directamente con JDBC en pandas dataframes, (sin usar spark), el ejemplo del Jupyter Notebook usa la sintaxis siguiente:

#Connexión jdbc

import jaydebeapi

```
cxn=jaydebeapi.connect("com.intersystems.jdbc.IRISDriver","jdbc:IRIS://iris4ml:51773/  
MLACADEMY", ["SuperUser", "sys"], "/usr/local/lib/intersystems/intersystems-jdbc-3.1.0.jar")  
data=pd.read_sql('select top 20000 * from  
PublishedABT.MLSimpleAppointmentsGetFeatures()',cxn)  
cxn.close()
```

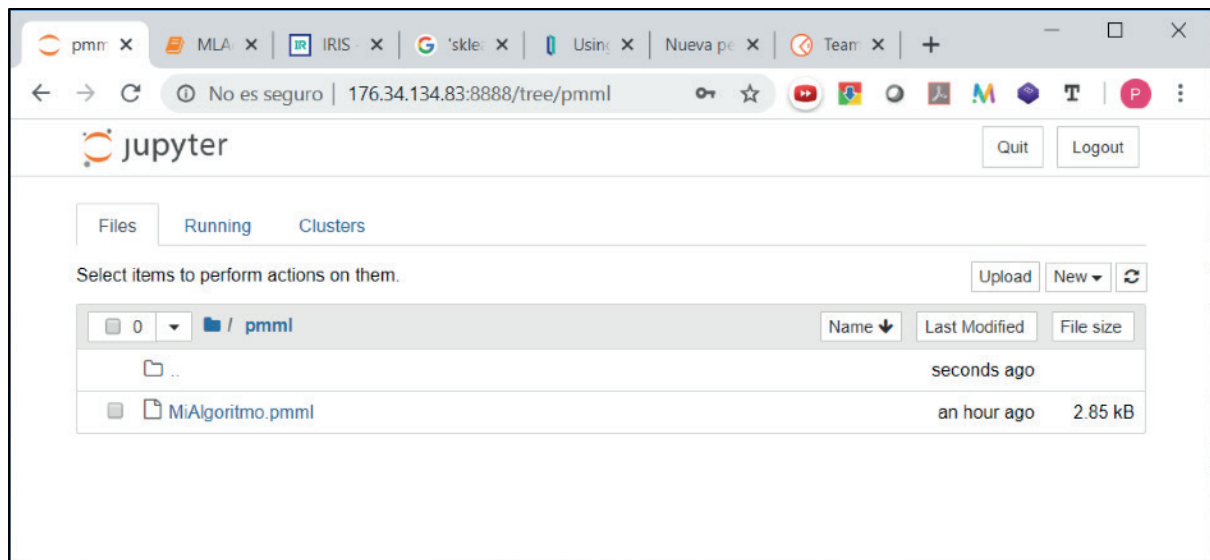

Ejercicio 3: exportar algoritmo PMML

- Tarea 1: Renombrar el PMML a generar y escoger la ubicación correcta:

El directorio /root/mlacademy/ se puede leer desde IRIS (como /shared) para importar allí el PMML.

#El Directorio /root/mlacademy/pmmml de este contenedor esta compartido como /shared/pmmml del contenedor IRIS sklearn2pmmml(myPMMLPipeline, pmmml="/root/mlacademy/pmmml/MiAlgoritmo.pmmml")

Se puede validar desde la pestaña principal de Jupyter Notebook que el fichero haya sido generado:



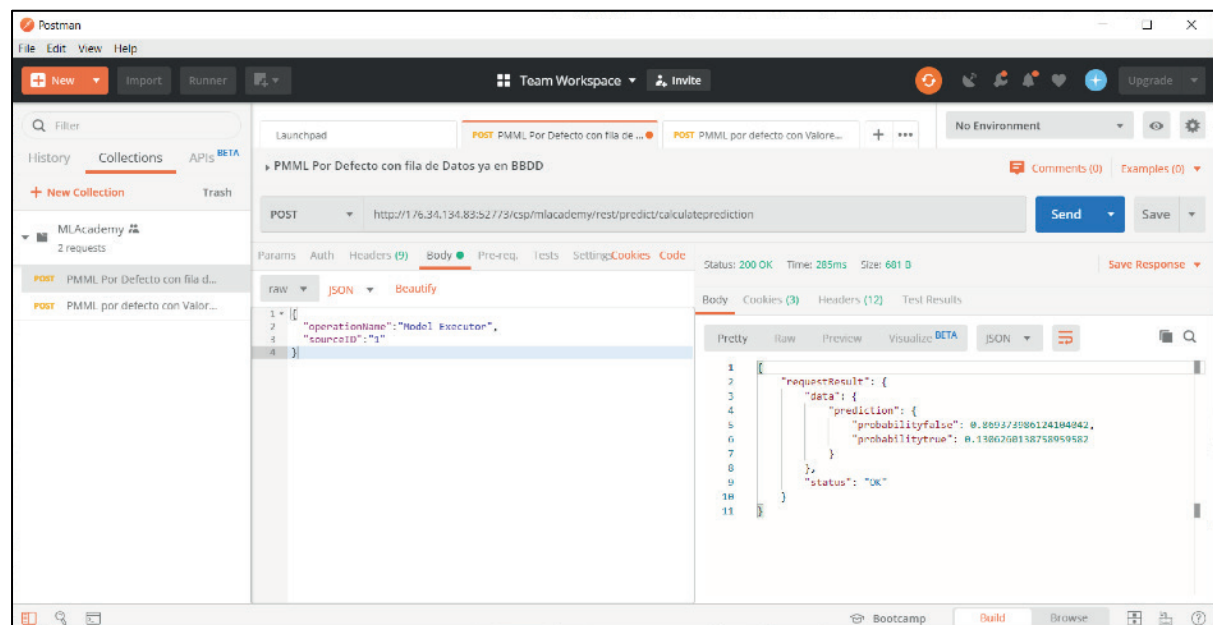
Ejercicio 4: Operacionalizar Algoritmo

- **Tarea 1:** Ejecutar una predicción con un modelo precargado

Desde un cliente REST (PostMan, SoapUI, ARC, etc), **llamar a IRIS** para ejecutar una predicción sobre la fila con ID=1 de la tabla Appointments:

Tipo Petición REST	POST
URL	http://localhost:52773/csp/mlacademy/rest/predict/calculateprediction
Header Content-Type	application/json
Body	{ "operationName": "Model Executor", "sourceID": "1" }

Ejemplo de Llamada desde Postman:



Notar que la capa de Interoperabilidad de IRIS ha abierto la fila con ID=1 del "Abstract Base Table2" sin que tengamos que especificar todos los valores de columnas "One Hot Encoded".

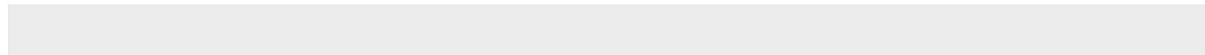
¿Cuál es la Probabilidad que el Paciente No acude a la Cita?

Ejecutar la predicción especificando valores para todos los campos:

Tipo de Petición REST	POST
URL	http://localhost:52773/csp/mlacademy/rest/predict/calculatprediction
Header Content-Type	application/json
Body	{ "operationName":"Model Executor", "dataMap": { "Age":"53", "Alcoholism":0, "Delay":182, "Diabetes":0, "Gender_F":0, "Gender_M":1, "Handicap":0, "Hypertension":1, "Scholarship":1, "smsReceived":1 } }

Notar como esta vez hemos tenido que especificar "Gender_F" y "Gender_M".

¿Cuál es la probabilidad que el Paciente No acude a la Cita?



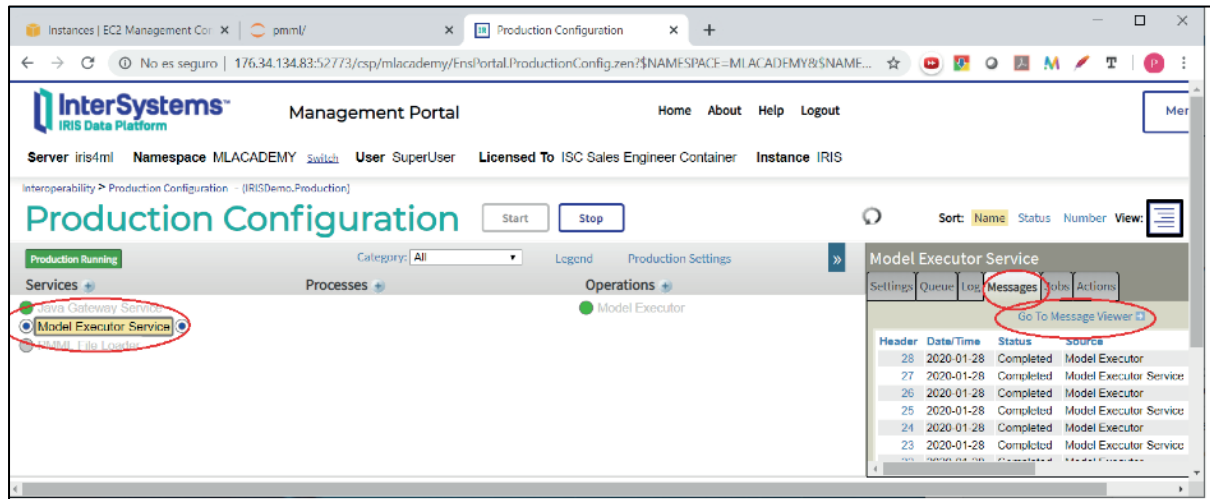
- Tarea 2: Ver trazas de Interoperabilidad (resultado de llamadas REST)

En esta tarea se miran las trazas de mensajería de Interoperabilidad de IRIS. se pueden ver todos los detalles del procesamiento de las peticiones de predicción.

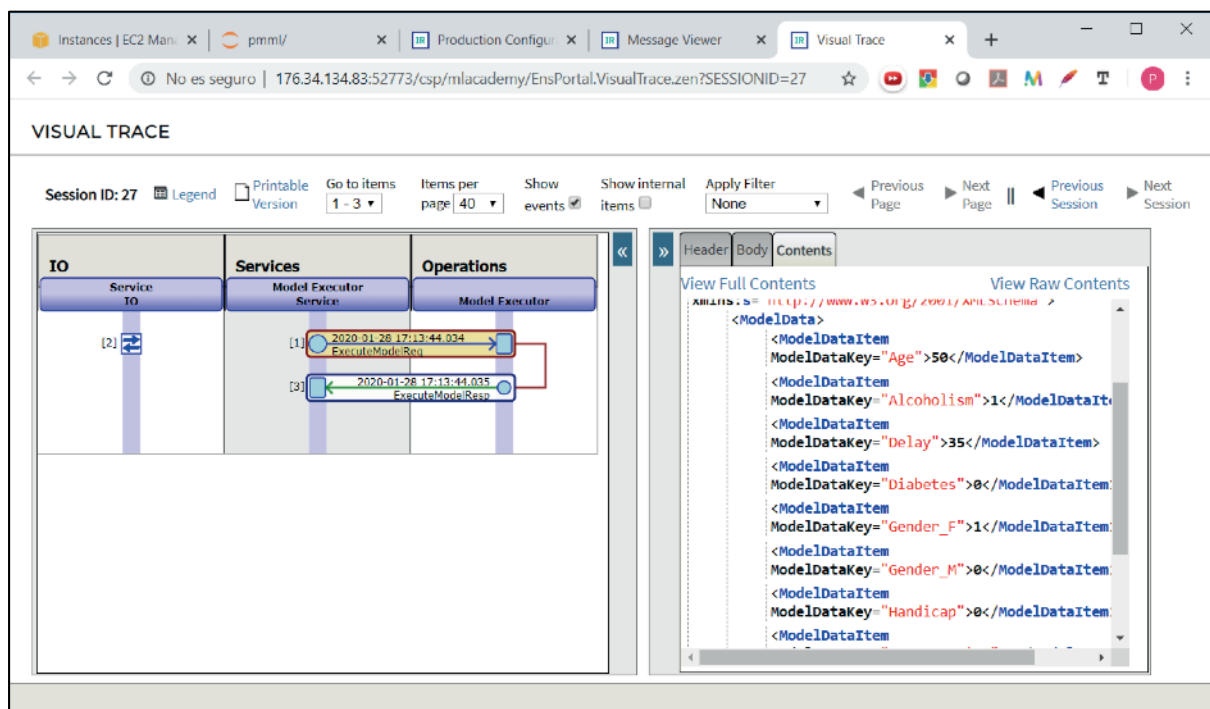
Abrir la pestaña del portal de gestión de InterSystems IRIS (<http://176.34.134.83:52773/csp/sys/UtilHome.csp>)

Cambiar el Namespace a MLACADEMY (hacer click en "switch" en la barra horizontal superior).

Seleccionar "Interoperability" / "Configure" / "Production":



Seleccionar "Model Executor Service" (Es el Servicio REST que se ha llamado), escoger el apartado de "Messages" y hacer click sobre "Go To Message Viewer". Allí hacer click sobre la primera fila de datos. Permite ver la traza de la petición de predicción y su respuesta, con todos los detalles (metadatos y Datos).

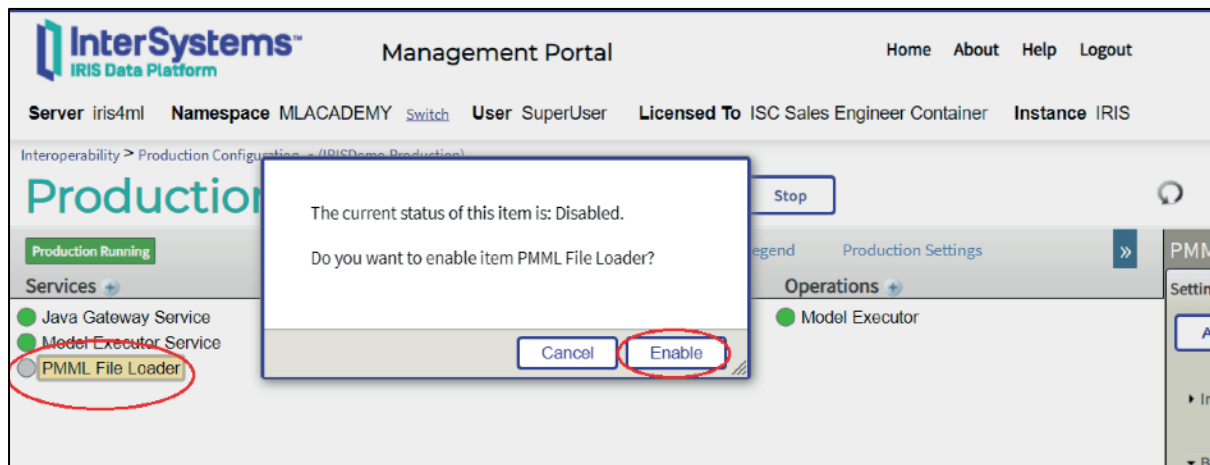


- **Tarea 3:** Importar el Algoritmo PMML en IRIS

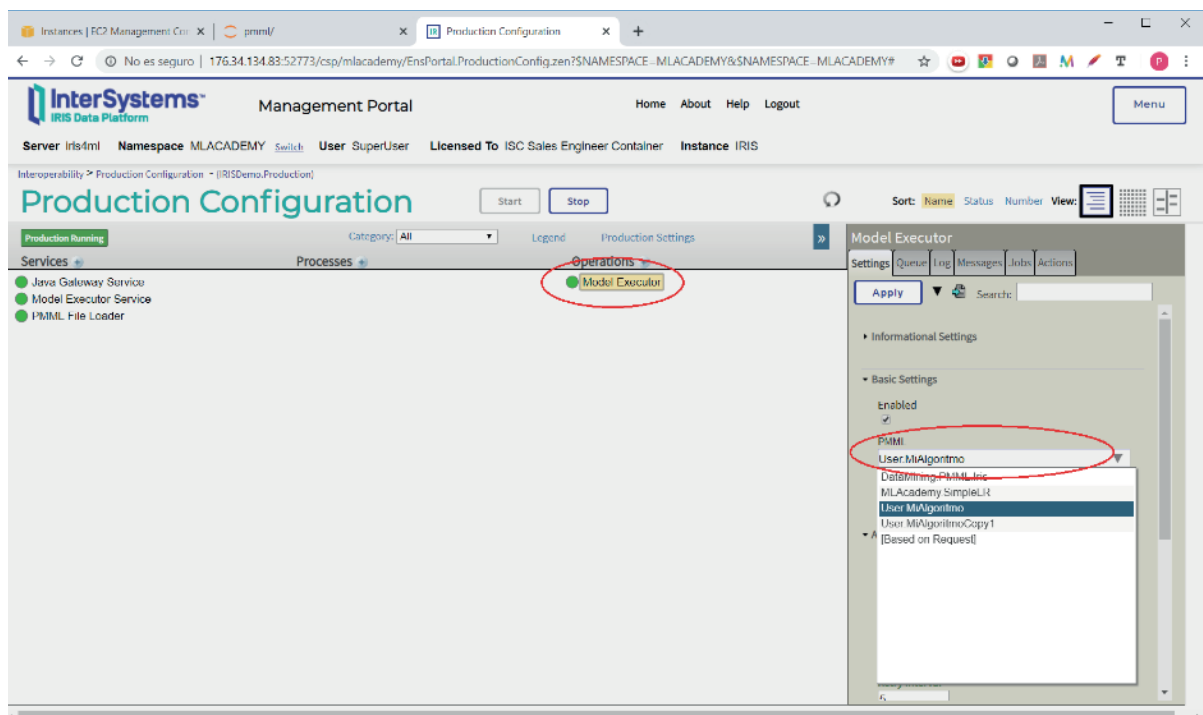
Volver a la Pestaña del browser intitulada "Production Configuration":

This document and the information disclosed herein are confidential and proprietary to InterSystems. This information may not be used or disclosed to any third party for any purpose other than that specifically authorised in writing by InterSystems

InterSystems IRIS ya tiene un "Servicio" llamado "PMML File Loader" que permite importar los PMML. Está desactivado. Hacer doble click para "Activarlo" e importar los ficheros PMML del directorio / shared/pmml:



Seleccionar el Operation "Model executor" para, en el apartado "Settings" cambiar el nombre del Algoritmo PMML a ejecutar. Seleccionar "User.MiAlgoritmo" para escoger el nuevo PMML importado. Finalizar la Operación con el botón "Apply":



- Tarea 4: (Opcional). Llamar al nuevo algoritmo importado



Se puede volver a repetir la llamadas REST desde el cliente REST. Ahora IRIS usa el nuevo PMML que se ha importado.

Ejercicio 5: Uso de los Gateway Python y R desde la capa de Interoperabilidad

Los gateways de Python y R permiten ejecución de código Python o R desde la plataforma IRIS, y un intercambio de información bidireccional entre python e IRIS. Más información disponible en:

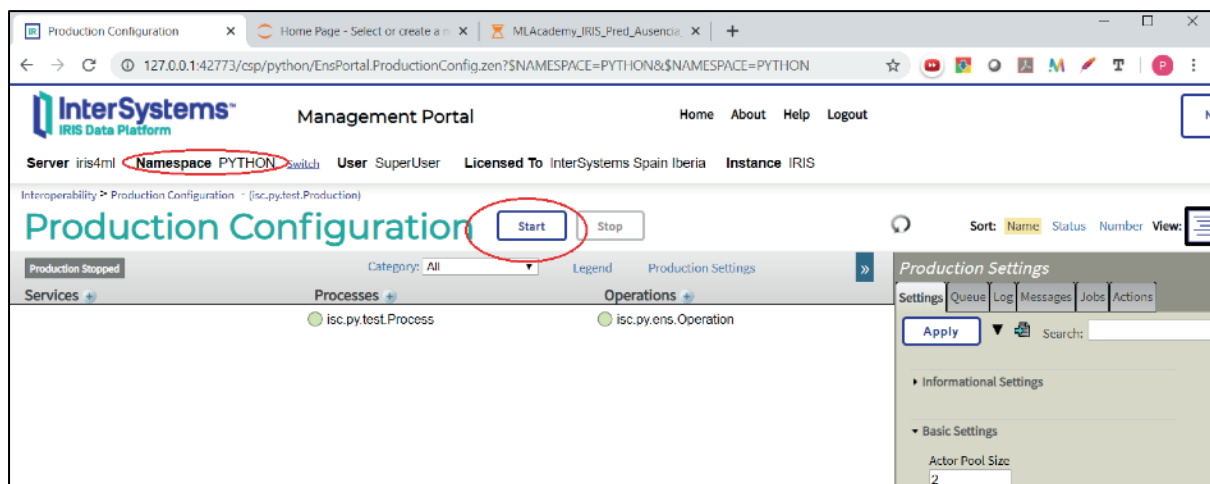
<https://community.intersystems.com/post/python-gateway-part-i-introduction>

En este ejemplo se usa el Python gateway desde la capa de interoperabilidad de IRIS para generar una grafico tipo "heatMap" con la librería Python Seaborn:

- Tarea 1: Arrancar la Producción "Python" y Configurar el Proceso

Abrir el Portal de Gestión en el Namespace **Python** y visualizar la Producción de Interoperabilidad.

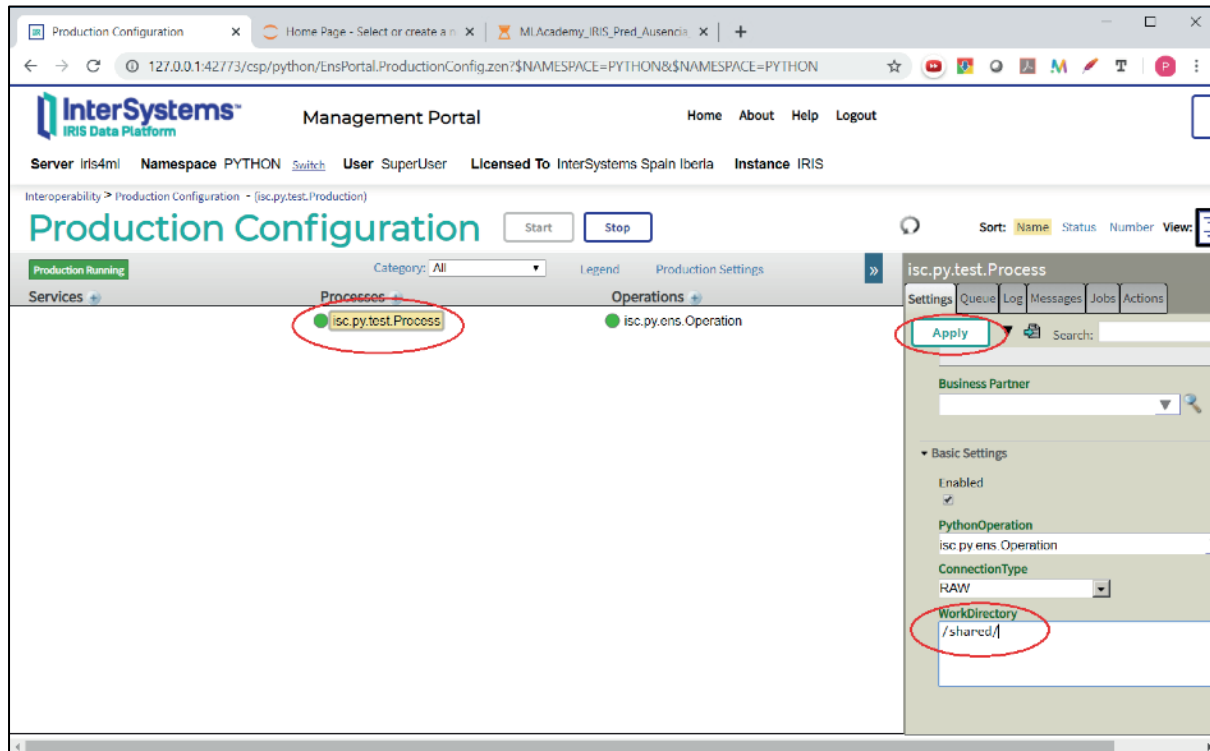
Menus "Interoperability" / "Configure" / "Production":



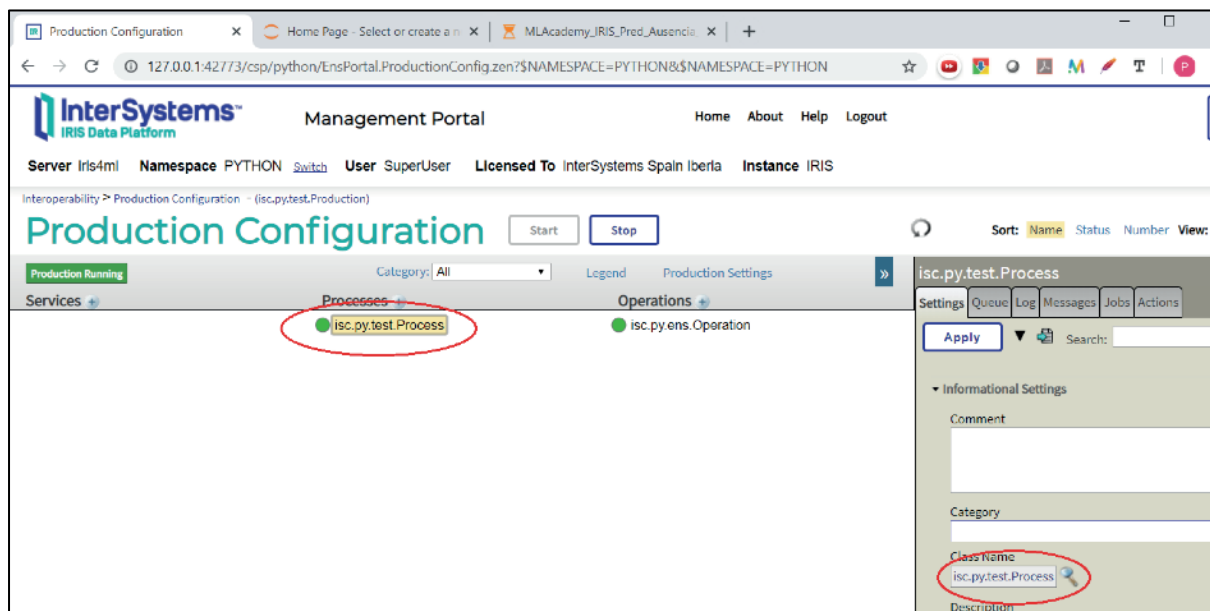
Arrancar la Producción con el botón "start".

Seleccionar el proceso "isc.py.test.Process". Revisar su configuración a la derecha en la pestaña "Settings" / ("Basic Settings" / "WorkDirectory"). Es la ubicación donde se genera el ".png" de la gráfica "HeatMap".

Cambiar el WorkDirectory a "/shared/" (sin olvidar la "/" final!), para generar allí el fichero .png:



- **Tarea 2 (Opcional):** Revisar la definición del Proceso "isc.py.test.Process"
Seleccionar el Proceso, y en la pestaña "Settings", en el apartado "Informational Settings" hacer click sobre la lupa a la derecha de "Class Name":



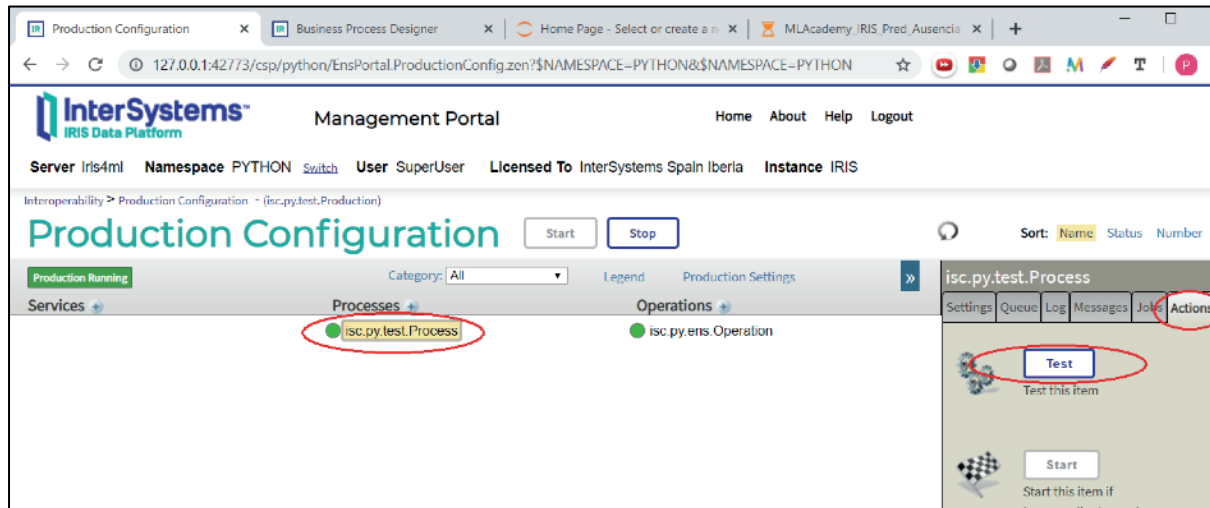
This document and the information disclosed herein are confidential and proprietary to InterSystems. This information may not be used or disclosed to any third party for any purpose other than that specifically authorised in writing by InterSystems

Revisar las casillas "RAW", "Correlation Matrix: tabular" y "Correlation Matrix Graph". En "Annotations, puede ver el código Python:

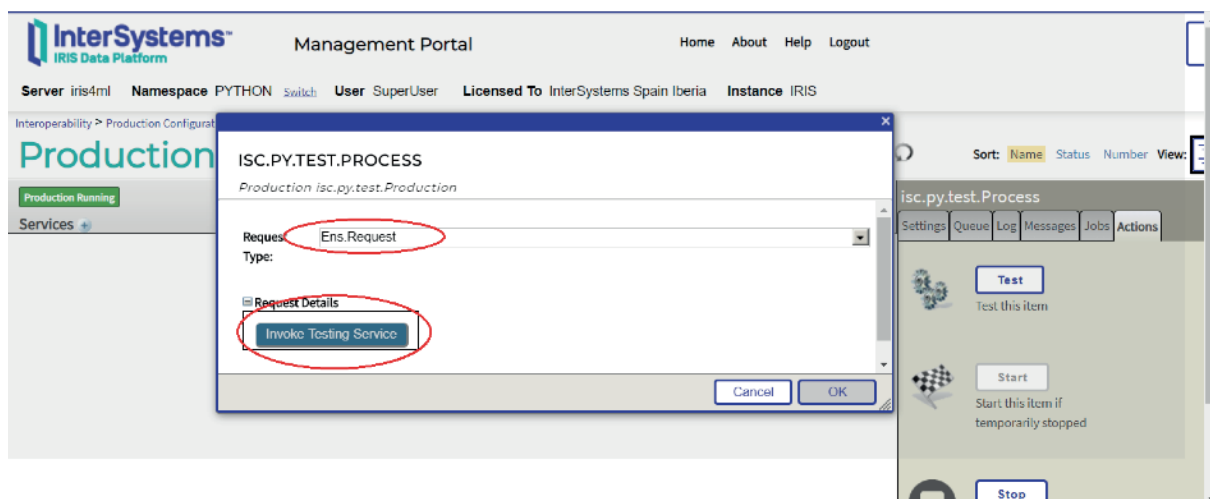
Casilla	Código SQL o Python
RAW	<pre>select Gender, Age, {fn TIMESTAMPDIFF(SQL_TSI_DAY, ScheduledDay, AppointmentDay)} as Delay , Neighborhood, scholarship, Hypertension, diabetes, Alcoholism, Handicap, smsReceived, noShow from SqlUser.Appointments</pre>
Correlation Matrix: Tabular	<pre>from dython import nominal #TimestampDiff -> STR Data['Delay']=Data['Delay'].astype(str).astype(int) corrmat=nominal.associations(Data, nominal_columns=['Gender','Neighborhood'],plot=False,return_results=True) #corrmat=Data.corr()</pre>
Wait Correlation Matrix: Graph	<pre>import matplotlib matplotlib.use('Agg') import matplotlib.pyplot as plt import seaborn as sns f=plt.figure(figsize=(20,20)) sns.set(font_scale=1.5) sns.heatmap(corrmat,annot=True,cmap="Blues") plt.title('Characteristics Correlation') f.savefig(r'#{process.WorkDirectory}SHOWCASE\${%PopulateUtils:Integer:1:100}.png') plt.close(f)</pre>

- Tarea 3: Ejecutar el Proceso y Revisar la traza de mensajes

En la pestaña de browser "Production Configuration", seleccionar el Proceso "isc.py.test.Process". A la derecha, seleccionar la pestaña "Action", y el botón "Test":



Seleccionar el Request Type= "Ens.Request", y enviar en mensaje al proceso con "Invoke testing service".



Hacer click sobre "Visual trace" para revisar todas las operaciones (comandos SQL, Python, etc), realizados por el proceso de interoperabilidad:

ISC.PY.TEST.PROCESS
Production isc.py.test.Production

Request Type: Ens.Request

Request Details

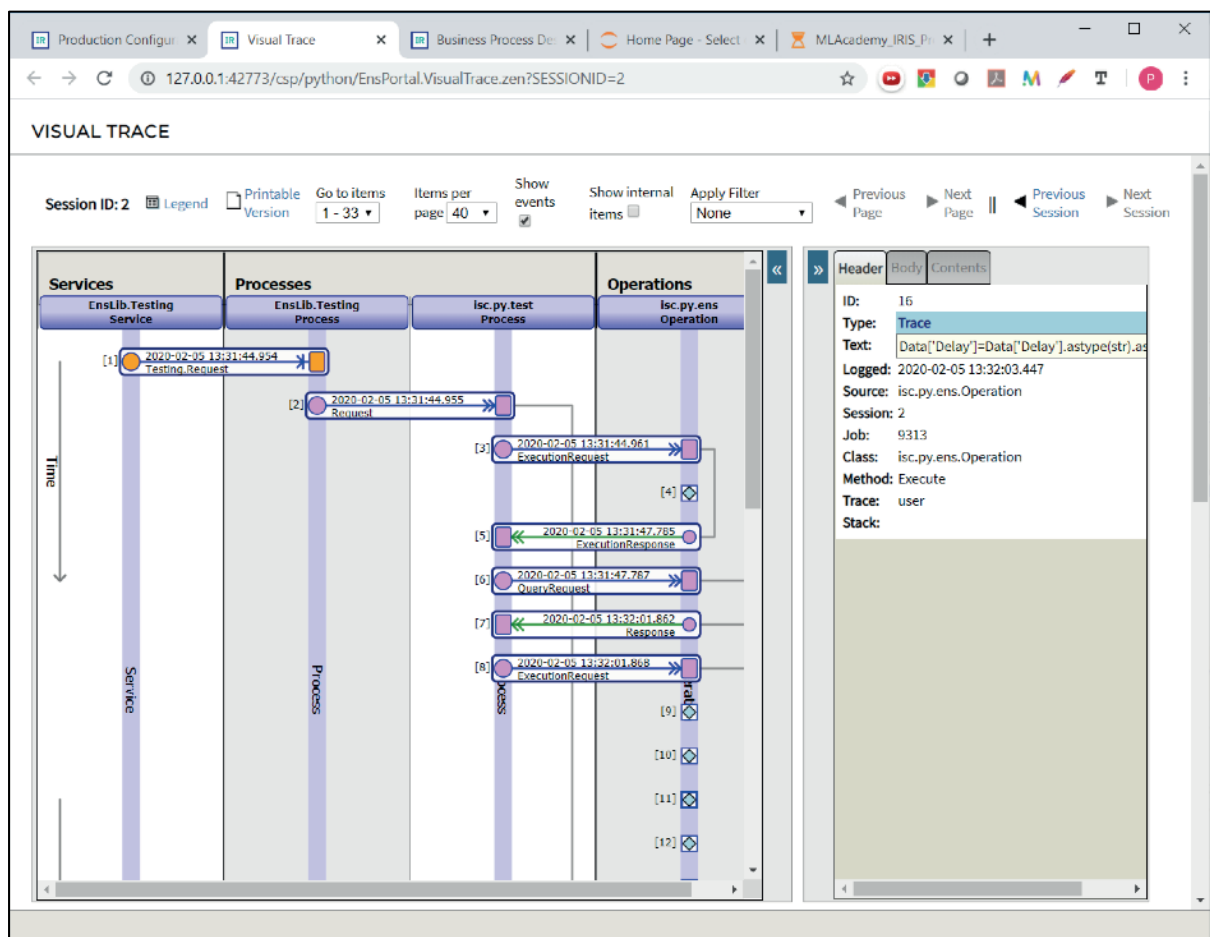
Test Results

Session Id: 2 **Visual Trace**
Request Sent: 2020-02-05 13:31:44.953
Response Received: 2020-02-05 13:32:10.649

Ens.Response
<ObjectId> 4

Cancel OK

La Traza de mensajes "Visual Trace" permite revisar los detalles de todas las operaciones entre componentes de InterSystems IRIS:



This document and the information disclosed herein are confidential and proprietary to InterSystems. This information may not be used or disclosed to any third party for any purpose other than that specifically authorised in writing by InterSystems

- Tara 4: Abrir el fichero generado

Abrir el fichero generado en el directorio compartidos "shared" (La tarea3 debe haber generado un nuevo fichero "ShowcaseNN.png", con NN un número aleatorio". La instalación ya contiene un fichero antiguo "SHOWCASE76.PNG").

Mirar el HeatMap:

Opcional: ¿Cuáles son las 3 variables que más incidencia tienen sobre el resultado (noShow)?

- Opcional: ¿Cuales son los 2 factores que más relación tienen con la presencia de hipertensión?

Ejercicio 6: IntegratedML

InterSystems Integrated ML es una funcionalidad futura (aún no liberada) de InterSystems IRIS para facilitar el uso de Machine Learning desde la propia plataforma. Está orientado a perfiles que NO son analistas de datos, y expone un interfaz simplificado y menos detallado. IntegratedML se encarga de preparar los Datos, probar varios algoritmos y entrenar el mejor algoritmo. Además, viene con un motor integrado, pero permite también hacer estas operaciones con motores externos (en este ejemplo se usa H2O).

Desde un editor SQL (de preferencia DBeaver), se puede pedir la creación de un modelo, entrenarlo, y realizar predicciones usando el mejor modelo entrenado:

- Tarea 1: Predecir si una persona sobrevive al naufragio del Titanic

Aquí usamos este ejemplo típico de ML porque no permite trabajar con un conjunto de datos más pequeño y obtener un entrenamiento más rápido (Como IntegratedML tiene que escoger entre varios algoritmos, tarda más en realizar el training y las optimizaciones).

En el Namespace "MLACADEMY", Revisar el contenido de la tabla:

```
select top 10 * from Titanic.Passenger
```

Crear un Modelo.

InterSystems IRIS incluye unos modelos propios, y la posibilidad de llamar a modelos externos. [Por ejemplo { 'ml_provider' : 'H2O' }]. Aquí usamos el modelo propio de InterSystems IRIS:

```
create model ISCSurvival Predicting (Survived) from titanic.Passenger
```

Entrenar el modelo (approx 2 minutos):

```
train model ISCSurvival from titanic.Passenger
```

Verificar los detalles del training y del mejor algoritmo, en el campo "Log":

```
select * from %ML.TrainingRun
```

TrainingRun									
select * from %ML.TrainingRun									
ID	CompletedAt	Log	Model	Name	Provider	RunStatus	StartedAt	TrainingDuration	TrainingQuery
1	2020-02-02 17:50:42	started importing data from IRIS... [2519]	1	FlowersH2O2	%ML.H2O.Provider	completed	2020-02-02 17:50:21	20.79	SELECT * FROM DataM
2	2020-02-05 13:50:09	started importing data from IRIS... [9772]	2	survival?	%ML.H2O.Provider	completed	2020-02-05 13:48:21	107.99	SELECT * FROM titanic

Revisar el campo Log en detalles:

```

preparing data (...)
determining feature columns (...)
building transform function (...)
analyzing feature correlations reducing features from 2104 to ..... 282
transforming training data (...)
evaluating as a Classification model
determining best estimator
  trying TensorFlow DNN 0.0
  trying Logistic Regression 0.9943399694178273
  trying Random Forest 0.994080869216493
picking LogisticRegression(C=1.0, class_weight='balanced', dual=False, fit_intercept=True,
intercept_scaling=1, l1_ratio=None, max_iter=100, multi_class='auto', n_jobs=1, penalty='l2',
random_state=None, solver='liblinear', tol=0.0001, verbose=0, warm_start=False)
that took 2.3005340099334717 seconds training model returning model
(...)
  
```

Realizar unas predicciones:

```

select top 10 ID, Name, Predict(ISCsurvival) as SurvivalPredicted, Survived from
Titanic.Passenger
  
```

Anexo 1: Referencias

Referencias de Estudios sobre Ausencias de Pacientes. Es un problema que ha sido ampliamente estudiado, Aquí algunos ejemplos

URL	Descripción
https://www.ncbi.nlm.nih.gov/pmc/articles/PMC6243417/	Data Analytics and Modeling for Appointment No-show in Community Health Centers (publicacion 2018)
https://academic.oup.com/jamia/article/25/8/924/4849782	Designing risk prediction models for ambulatory no-shows across different specialties and clinics
https://journals.plos.org/plosone/article?id=10.1371/journal.pone.0214869	Development and validation of a patient no-show predictive model at a primary care setting in Southern Brazil
https://towardsdatascience.com/predicting-missed-hospital-appointments-using-machine-learning-what-are-the-risks-a388348109d	Predicting missed hospital appointments using machine learning - what are the risks?

Anexo 2: Contenidos adicionales

Aquí unos recursos disponibles para ahondar más en los temas abordados en esta academia.

Tema	URL	Descripción
InterSystems Elearning	https://learning.intersystems.com/	La plataforma de aprendizaje de InterSystems
InterSystems Community	https://community.intersystems.com/	Comunidad de Desarrolladores
Python & R Gateways y Ejemplos adicionales	https://github.com/intersystems-community/PythonGateway https://openexchange.intersystems.com/package/PythonGateway	
Ejemplos de Python & R Gateways	https://github.com/intersystems-community/Convergent-Analytics	Ejemplos avanzados de ML con IRIS

Anexo 3: Instalación y Configuración

El material de la academia está basado en contenedores Docker que contienen todo el software y los ejemplos necesarios. Se puede instalar y ejecutar en cualquier sistema operativo que permita la ejecución de docker y docker-compose y un browser Web. Opcionalmente, se pueden usar un editor (VSstudio Code o Atelier o IRIS Studio), un cliente REST , y un editor SQL (DBeaver recomendado).

Para realizar los ejercicios, es necesario instalar adicionalmente un browser web y se recomienda instalar un editor SQL (dbeaver), con el driver JDBC de InterSystems IRIS.

software	Comentarios
git	conveniente para descargar el proyecto de github
docker	docker-ce o docker-desktop
docker-compose	
browser web	Chrome, Firefox,...
REST Client	Cliente REST: ARC, o Postman, o FireFox RESTClient..
dbeaver	Editor SQL. Se prefiere dbeaver para la visualización del Log de IntegratedML-

Los contenedores instalados usan los siguientes puertos TCP/IP para la comunicación. Es importante que no estén bloqueados (por Firewall de Windows u otro) y accesibles desde las herramientas gráficas de usuario (browser Web, Editor, Cliente REST y DBeaver).

Puerto TCP	uso
8888	Contenedor de Servidor Jupyter Notebook
51773	SuperServer IRIS (puerto para acceso JDBC, Spark, IDE)
52773	Servidor Web de IRIS para Portal de gestión, pruebas REST, Terminal Web
22	SSH si necesario para un una instalación en una máquina Virtual local o remota Linux
3389	RDP si necesario para una instalación en una máquina Virtual local o remota Windows

Los Ejercicios acceden a varias URL con usuarios y password. Aquí un recapitulativo de las principales URLs:

función	URL	usuario	password
Portal de Gestión IRIS	http://localhost.52773/csp/sys/UtilHome.csp	SuperUser	sys
Servidor Jupyter Notebook	http://localhost:8888		IRIS
Acceso JDBC local	jdbc:IRIS//127.0.0.1:51773/MLACADEMY	SuperUser	sys
Acceso Spark desde Contenedor Jupyter4iris	http://iris4ml:51773/MLACADEMY	SuperUser	sys

Instalación de pre-requisitos

cliente git

Es necesario descargar un proyecto de github. Para hacerlo conviene instalar "git" (o escoger una descarga manual desde un browser web).

Git en Linux (ejemplo Amazon Linux 2)

```
sudo yum install -y git
```

Docker y docker-compose (ejemplo Amazon Linux 2)

```
sudo yum update -y
sudo amazon-linux-extras install -y docker
sudo service docker start
```

Es necesario reiniciar la sesión despues de usermod.

```
sudo curl -L "https://github.com/docker/compose/releases/download/1.25.3/docker-compose-$(uname -s)-$(uname -m)" -o /usr/local/bin/docker-compose
sudo chmod +x /usr/local/bin/docker-compose
sudo ln -s /usr/local/bin/docker-compose /usr/bin/docker-compose
```

Descarga e Instalación del proyecto los contenedores MLAcademy

Descargar el proyecto "PYDuquesnoy/irismlacademy.git" de github. Esta operación se puede hacer desde un terminal con:

```
git clone https://github.com/PYDuquesnoy/irismlacademy.git
```

(Una alternativa es usar un browser web, ir a "<https://github.com/PYDuquesnoy/irismlacademy>" y hacer click sobre el botón "clone or download")

Descargar los contenedores: ir al directorio del proyecto, e usar docker-compose para obtener los contenedores.

```
cd irismlacademy
```

```
docker login
```

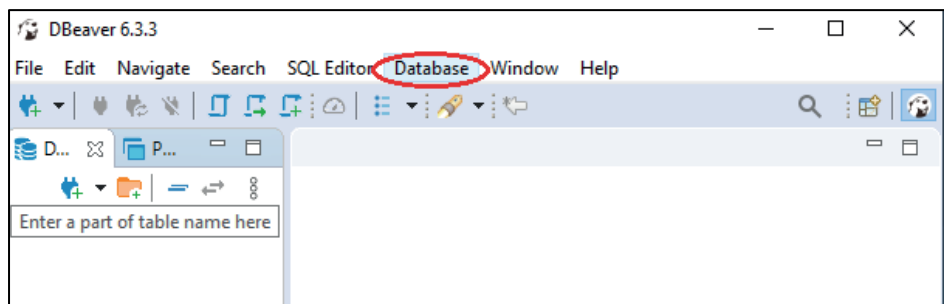
```
docker-compose pull
```

Instalación de Dbeaver + InterSystems IRIS JDBC

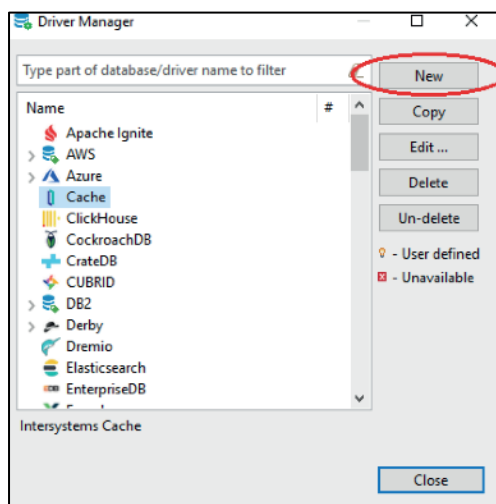
Descargar Dbeaver:

<https://dbeaver.io/download/>

Abrir DBeaver, y seleccionar el menú "Database" / "Driver Manager" para definir una nueva conexión JDBC a IRIS:



Crear una nueva definición con el botón "new":

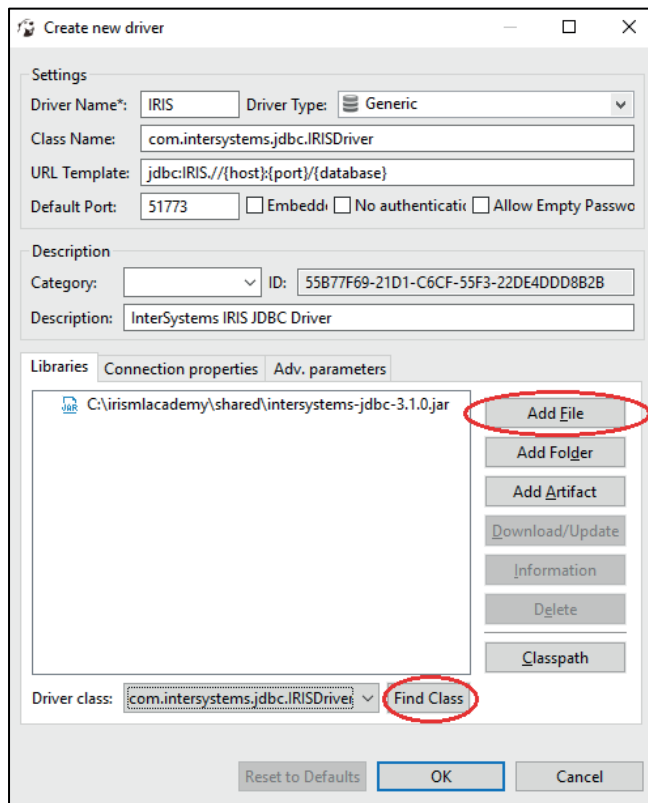


Y rellenar la información como sigue:

This document and the information disclosed herein are confidential and proprietary to InterSystems. This information may not be used or disclosed to any third party for any purpose other than that specifically authorised in writing by InterSystems

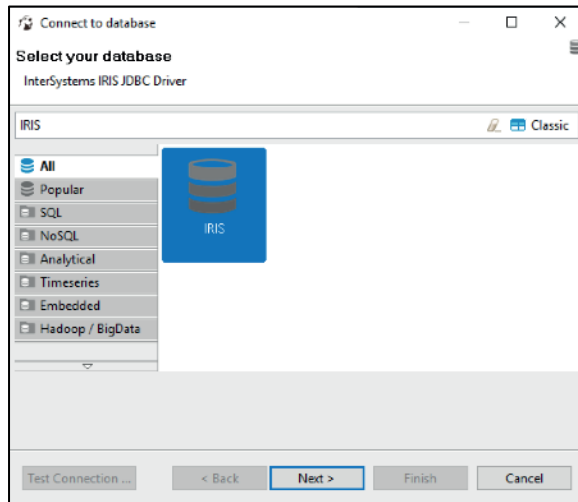
Campo	Valor
Driver Name	IRIS
Driver Type	Generic
ClassName	com.intersystems.jdbc.IRISDriver
URL Template	jdbc://IRIS://{host}:{port}/{database}
Default Port	51773
Description	InterSystems IRIS JDBC Driver
Libraries	c:\irismacademy\shared\intersystems-jdbc-3.1.0.jar
Driver class	com.intersystems.jdbc.IRISDriver

hacer click sobre **add file** para añadir la referencia al driver JDBC distribuido como parte del proyecto descargado de git, y hacer click sobre **Find Class** y finalmente OK:



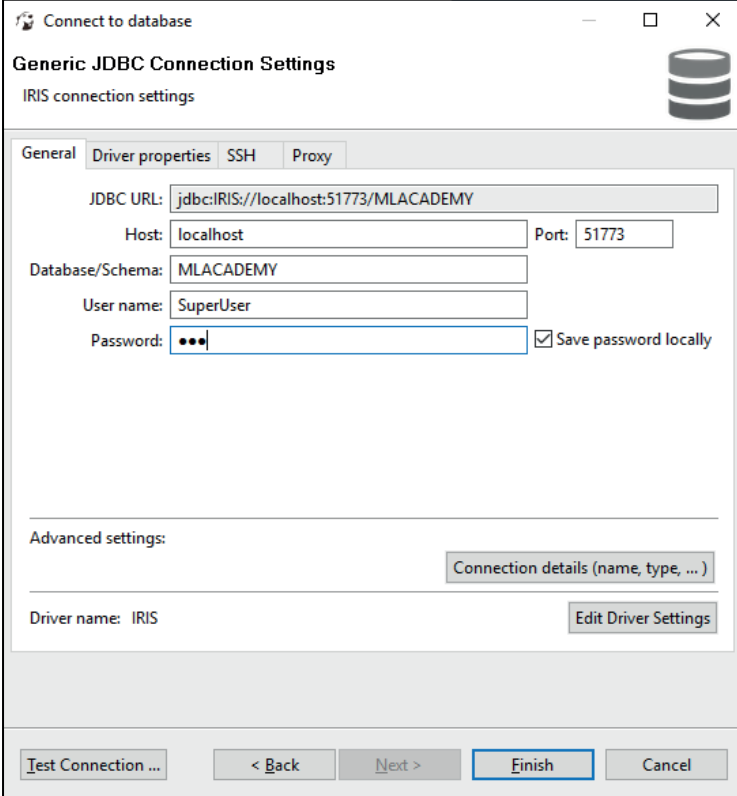
Con el menú "Database" / "New Database Connection", seleccionar IRIS para definir una nueva conexión:

This document and the information disclosed herein are confidential and proprietary to InterSystems. This information may not be used or disclosed to any third party for any purpose other than that specifically authorised in writing by InterSystems



Rellenar la información como sigue:

Campo	Valor
host	localhost o la IP del servidor/VM con la instalación del servidor iris4ml de la academia
port	51773
Database	MLACADEMY
User name	SuperUser
Password	sys



Connect to database

Generic JDBC Connection Settings

IRIS connection settings

General Driver properties SSH Proxy

JDBC URL: jdbc:IRIS://localhost:51773/MLACADEMY

Host: localhost Port: 51773

Database/Schema: MLACADEMY

User name: SuperUser

Password: ●●● ☒ Save password locally

Advanced settings:

Connection details (name, type, ...)

Driver name: IRIS

Prueba de Verificación de la instalación

```
cd irismlacademy
```

```
docker-compose up
```

Esta instrucción debe arrancar los 2 contenedores necesarios para la academia. Para Verificar que funcionen correctamente, conectarse con un browser Web a las URLs:

Contenedor	URL local
IRIS (portal de Gestión)	http://localhost.52773/csp/sys/UtilHome.csp Login con "SuperUser" / "sys"
Jupyter Notebook	http://localhost:8888 login con "IRIS"

Además se puede probar la conexión desde DBeaver definida anteriormente.