

Sistema de Recomendaciones con Azure Machine Learning

Introducción

Machine Learning facilita el análisis predictivo empleando algoritmos que iterativamente aprenden de la base de datos, es una de las áreas con mayor crecimiento dentro de la computación. Estamos rodeados de las nuevas aplicaciones de Machine Learning; como la detección de fraudes de tarjeta de crédito, autos que se manejan solos, reconocimiento óptico de caracteres y recomendaciones de productos en tiendas online. Mientras las computadoras se hacen más inteligentes, nosotros también los hacemos más inteligentes.

Azure Machine Learning es un servicio de analítica predictiva de la nube que ofrece una experiencia simplificada para científicos en ciencias de datos de todos los niveles. Está acompañado de Azure Machine Learning Studio (ML Studio) que es una página que funciona como una herramienta muy funcional, en la cual, la interfaz permite arrastrar y soltar para construir modelos de ML. Viene con una librería de experimentos y algoritmos desarrollados y probados en la vida real por Microsoft, un ejemplo claro es Bing. Y está construido en soporte con R y Python, esto significa que puedes construir tu propio código para personalizar tu modelo. Una vez construido y entrenado tu modelo, fácilmente puedes exponerlo como un servicio Web.

En este laboratorio, utilizaremos Azure Machine Learning para entrenar un sistema de recomendaciones de películas. El modelo aprende de un grupo de usuarios quienes calificaron un subconjunto de un catálogo de películas. Estas preferencias se utilizan para predecir el rating de las películas que no ha visto el usuario y así poder recomendar las películas que tienen mayor probabilidad de que al usuario le agraden.

Objetivo: Construir, entrenar y calificar un modelo usando Azure Machine Learning

Prerrequisitos

Lo siguiente es requerido para poder realizar el laboratorio:

- Una suscripción activa de Microsoft Azure

Índice

Ejercicio 1: Crear un experimento en Machine Learning Studio

Ejercicio 2: Cargar una base de datos

Ejercicio 3: Pre-procesar la base de datos

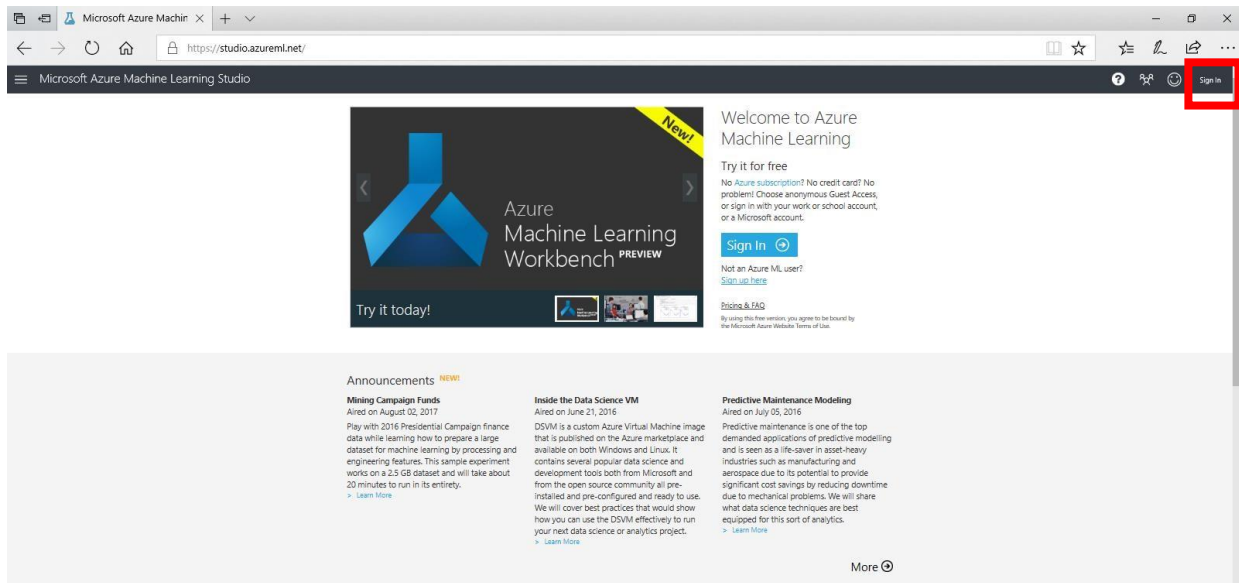
Ejercicio 4: Entrenar el modelo

Ejercicio 5: Probar el modelo

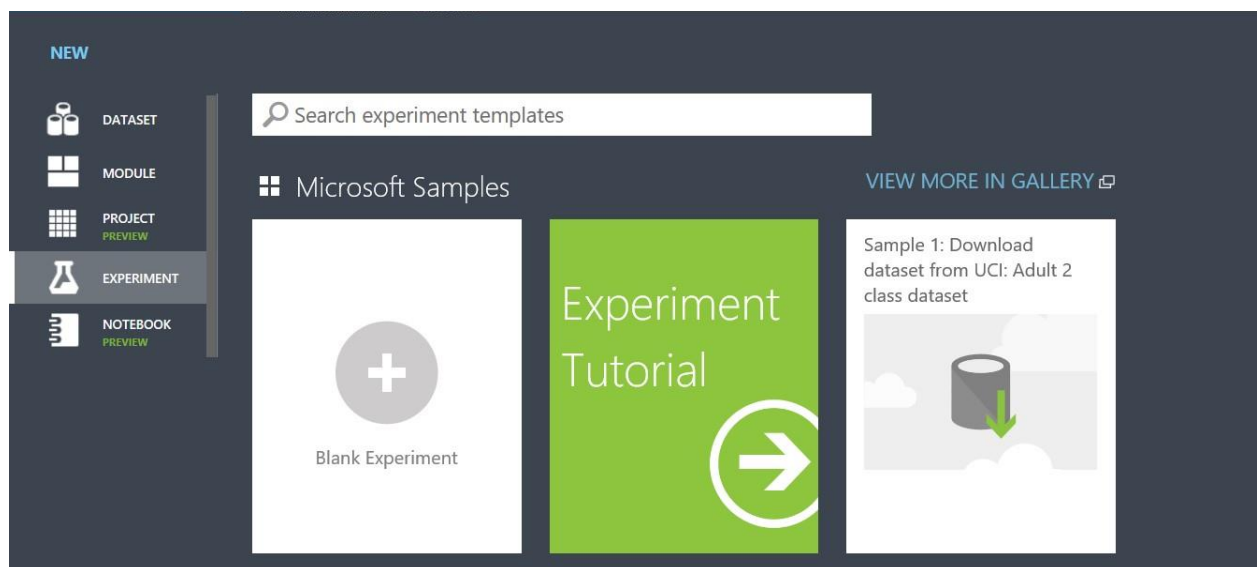
Ejercicio 1: Crear un experimento en Machine Learning Studio

1. Abrir la página de [Machine Learning Studio](https://studio.azureml.net/).

2. Dar clic en **Sign in**.



3. En el ML Studio, dar clic en **+ New** en la esquina superior izquierda. Observamos que en este menú vienen experimentos ya precargados que nos pueden servir de ejemplo ilustrando diversas situaciones, algunos de ellos provienen de casos de éxito de clientes. Abrimos un nuevo experimento dándole clic en **Blank Experiment**.



4. Da clic en el título del experimento en la parte superior de la página (“Experiment created on...”) y teclea el nombre de tu nuevo experimento, puede ser “Sistema de Recomendaciones”



Ahora que se ha creado el experimento, el siguiente paso es importar la base de datos y construir el modelo.

Ejercicio 2: Cargar una base de datos

Azure Machine Learning Studio viene con distintos ejemplos de bases de datos. Una gran variedad de datos están disponibles en diferentes recursos como: [Data.gov](https://data.gov), [Kaggle](https://www.kaggle.com) y el [repositorio de Machine Learning](https://github.com/UCI-Machine-Learning-Datasets) de la Universidad de California Irvin. Nosotros también podemos cargar nuestra propia base de datos en diferentes formatos. Lo que tendríamos que hacer es en **+ New > Dataset > From Local File** y elegir la base de datos que sea de nuestra preferencia.

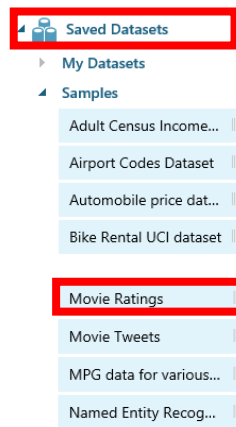
Datos

Los datos consisten en ratings extraídos de tweets. Cuenta con 225,000 ratings de 15,742 películas hechos por 26,770 usuarios de Twitter. Para más información acerca de la base de datos, puedes consultar el repositorio escrito por by Dooms, De Pessemier and Martens [1] dando clic [aquí](#).

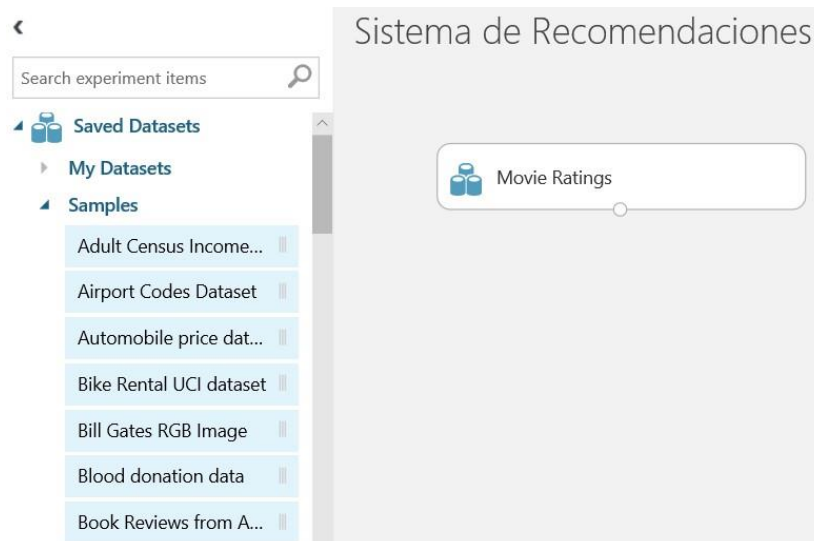
Cada columna consiste en in ID de usuario, ID de película, el rating y el instante en el que se hizo el tweet, pero este último no lo utilizaremos en el análisis.

También utilizaremos un archivo con los nombres de las películas que fue extraído de IMDB (Internet Movie Database).

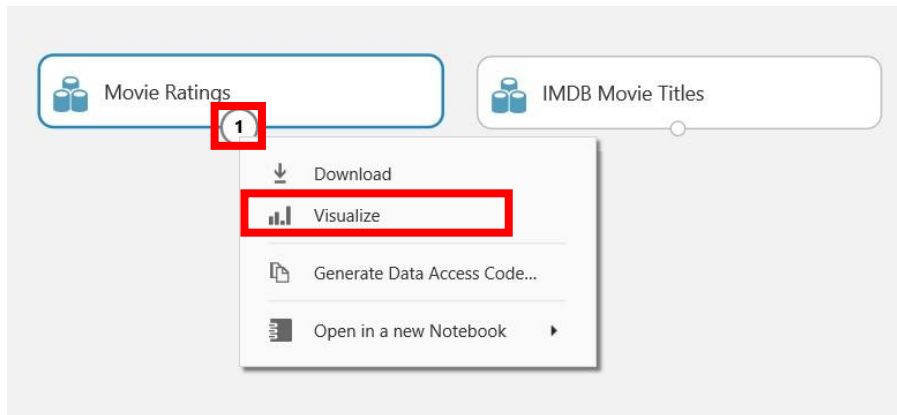
1. Ve al menú de módulos a la izquierda y busca las base de datos “Movie Ratings” del grupo *Saved Datasets* **dentro de Samples**



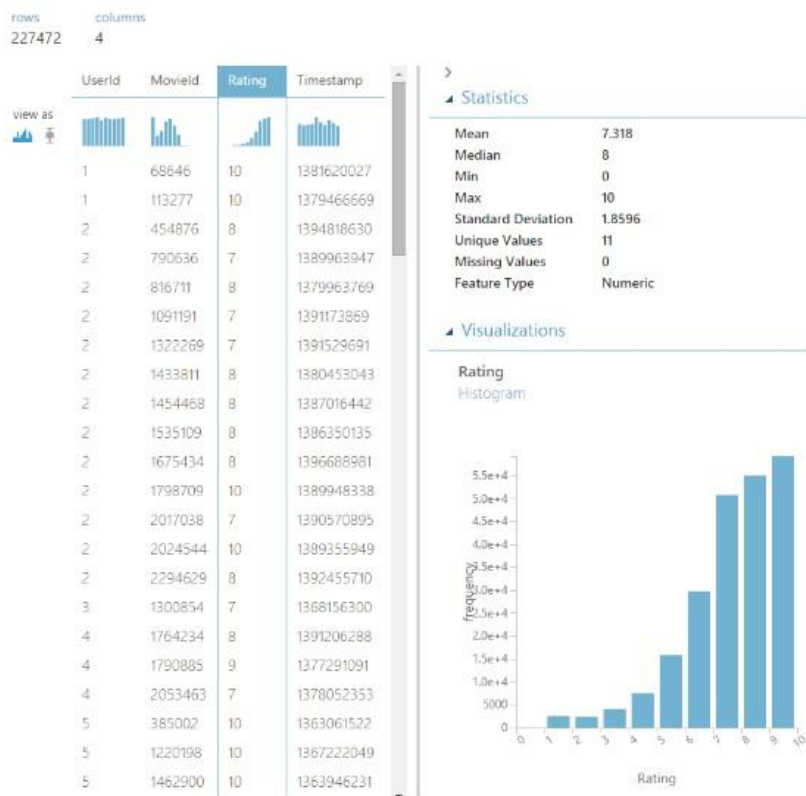
2. Arrastra la base de datos y déjala sobre el lienzo (el área gris de la derecha)



3. Lo mismo haremos con la base de datos “IMDB Movie Titles”
4. Para ver cómo se ve la base de datos, da clic en “Movie Ratings” y luego da clic derecho en el puerto de salida (el círculo con un “1”) debajo de la base de datos y selecciona **Visualize**.



Las variables de la base de datos aparecen como columnas. Esta herramienta de visualización nos permite ver medidas de dispersión y características de la variable. También podemos ver los diagramas de frecuencia



Cierra la ventana de visualización dándole clic en la "x" que está en la esquina superior derecha. Los datos están cargados, ahora es tiempo de trabajar con ellos

Ejercicio 3: Pre-procesar la base de datos

Dentro del preprocesamiento de la base de datos se incluye:

- Limpieza
- Integración
- Transformación
- Reducción
- Discretización y cuantización

En Azure ML Studio, se encuentran herramientas para poder lograr el preprocesamiento de datos. Estas herramientas las podemos encontrar en el grupo de *Data Transformation*.

1. Ve al grupo de *Data Transformation* y selecciona la opción **Manipulation**, ahí aparecerá la herramienta **Edit Metadata**, ponla en el lienzo.

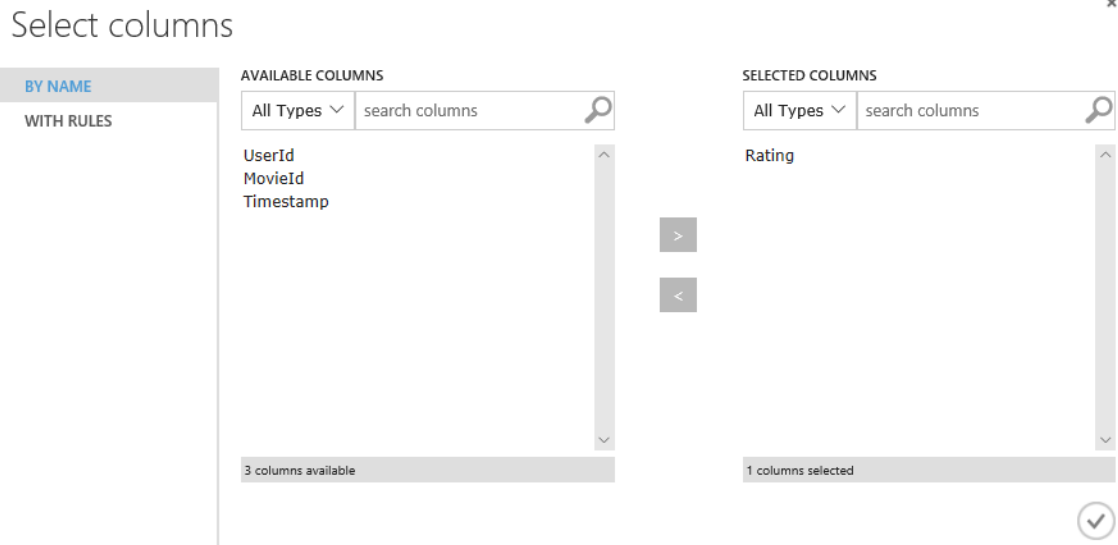


Conecta el nodo inferior de la base de datos “Movie Ratings” con el nodo superior de la herramienta.

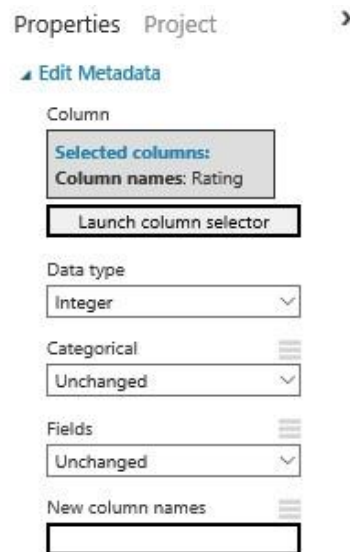


Queremos que los ratings sean del tipo entero, de ahí que tenemos que configurar las propiedades del módulo **Edit Metadata**. Damos clic en el módulo y lo configuramos de la siguiente forma:

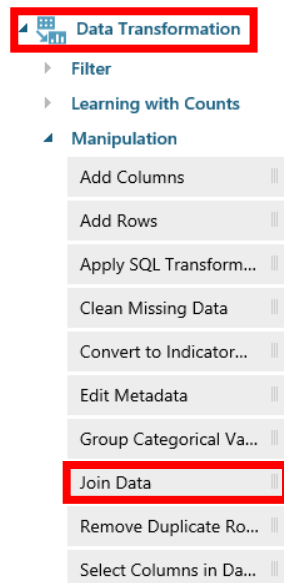
- **Selected columns:** damos clic en **Launch column selector** y seleccionamos la columna "rating".



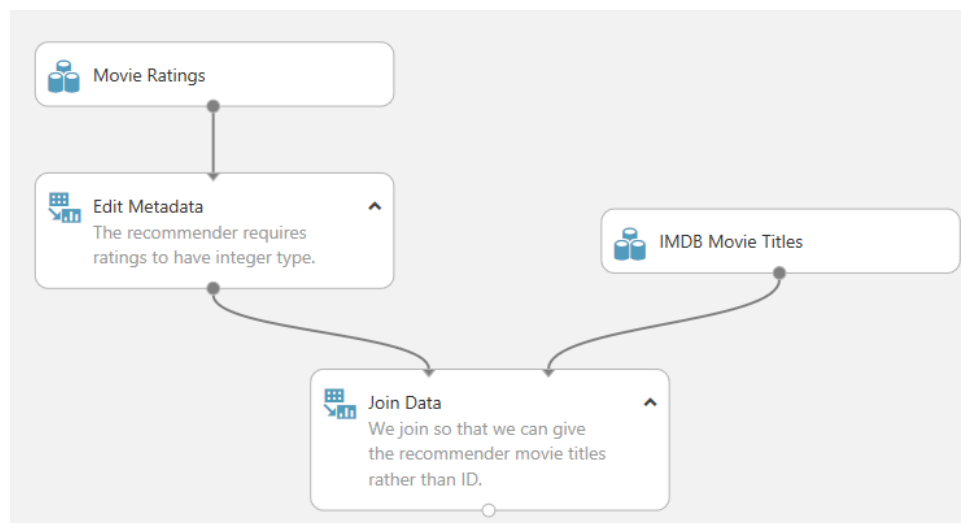
- Para las demás opciones elegimos la siguiente configuración:



2. Lo que haremos ahora es juntar las bases de datos, esto es para que podamos dar como recomendación los títulos de las películas y no su ID. En grupo *Data Transformation*, dentro de la opción **Manipulation** se encontrará la herramienta **Join Data**, la arrastramos en nuestro lienzo.



Conectamos el nodo superior izquierdo con el nodo inferior de **Edit Metadata** y el nodo superior derecho con el nodo inferior de la otra base de datos "IMDB Movie Titles"



Damos clic en el módulo y Dentro de sus propiedades seleccionamos las siguientes características:

- **Join key columns for L:** Damos clic en **Launch column selector** y seleccionamos “MovieId”.
-

Select columns

BY NAME

WITH RULES

☐ Allow duplicates and preserve column order in selection

Begin With

ALL COLUMNSNO COLUMNS

Include

column names

MovieId ✕

+

-

- **Join key columns for R:** : Damos clic en **Launch column selector** y en la opción *Available columns* elegimos “Movie Name” y en la opción *Selected columns* elegimos “Movie ID”

Select columns

BY NAME

WITH RULES

AVAILABLE COLUMNS

All Types ▼search columns 🔍

Movie Name

1 columns available

SELECTED COLUMNS

All Types ▼search columns 🔍

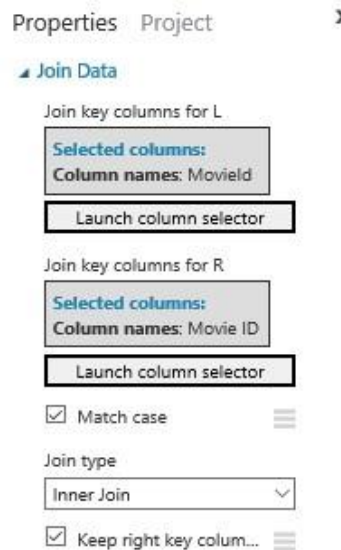
Movie ID

1 columns selected

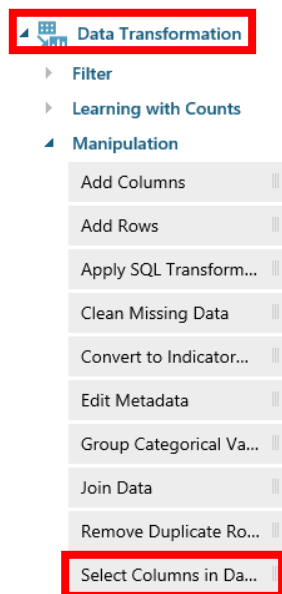
>

<

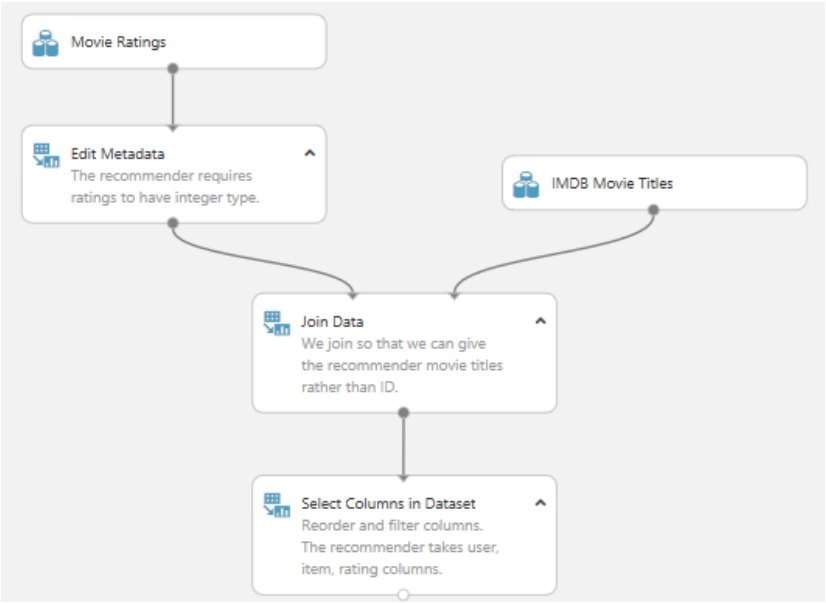
- Elegimos las siguientes propiedades de esta forma:



3. Ahora, seleccionaremos las columnas que queremos utilizar. En este caso utilizaremos: “UserId”, “Movie Name” y “Rating”. Dentro del grupo *Data Transformation*, dentro de la opción **Manipulation** se encontrará la herramienta **Select Columns in Dataset** y la arrastramos en nuestro lienzo.



Conectamos el nodo superior de este módulo con el nodo inferior de **Join Data**.



Dentro de sus propiedades, damos clic en **Launch column selector** y seleccionamos las columnas ya mencionadas.

Select columns

×

BY NAME

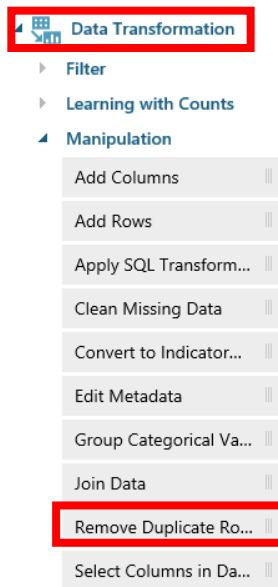
WITH RULES

☒ Allow duplicates and preserve column order in selection

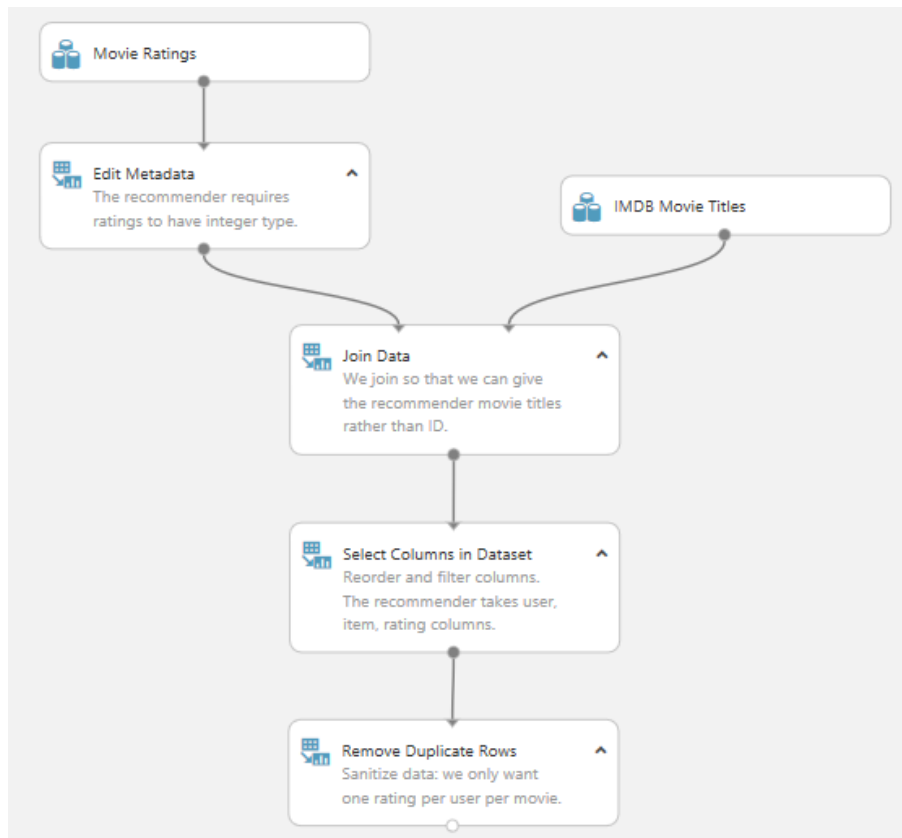
Include column names

UserId × Movie Name × Rating ×

4. Dentro de esta base de datos se pueden observar columnas duplicadas y sólo queremos un rating por usuario. Dentro del grupo *Data Transformation*, dentro de la opción **Manipulation** se encontrará la herramienta **Remove Duplicate Rows**.



Lo arrastramos en el lienzo y conectamos su nodo superior con el nodo inferior de **Select Columns in Dataset**.



Para configurar el módulo, damos clic en el módulo y del lado derecho, dentro de sus propiedades, damos clic en **Launch column selector** y seleccionamos “UserId” y “Movie Name”

Select columns

BY NAME

WITH RULES

☐ Allow duplicates and preserve column order in selection

Begin With

ALL COLUMNS NO COLUMNS

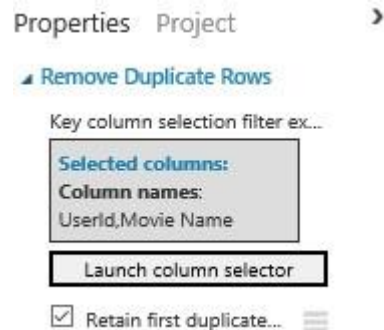
Include column names

UserId X Movie Name X

+ -

Y seleccionamos la opción **Retain first duplicate row**.

Las propiedades deben quedar de la siguiente forma:



Ejercicio 4: Entrenar el modelo

Durante el entrenamiento, la computadora busca patrones en los datos que pueda usar para predecir valores. Necesitamos elegir un algoritmo para analizar los datos. Hay muchos tipos de casos dentro de Machine Learning (clasificación, agrupaciones, regresiones, recomendaciones, etc.) con diferentes tipos de algoritmos diseñados para cada caso, dependiendo en su precisión y eficiencia. En este experimento utilizaremos un algoritmo de recomendación.

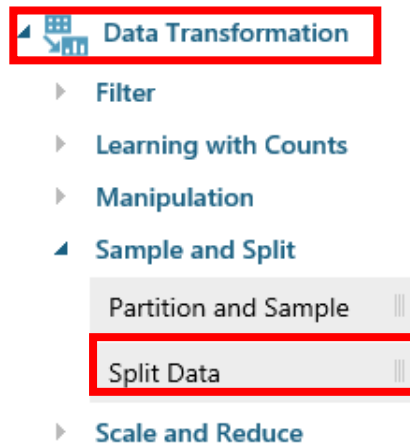
El principal objetivo de un sistema de recomendaciones es recomendar uno o más artículos a usuarios. Como ejemplos pueden surgir: películas, restaurantes, libros o canciones. Un usuario puede ser una persona, un grupo de personas u otra entidad con preferencias de artículos.

Existen dos principales categorías de sistemas de recomendación:

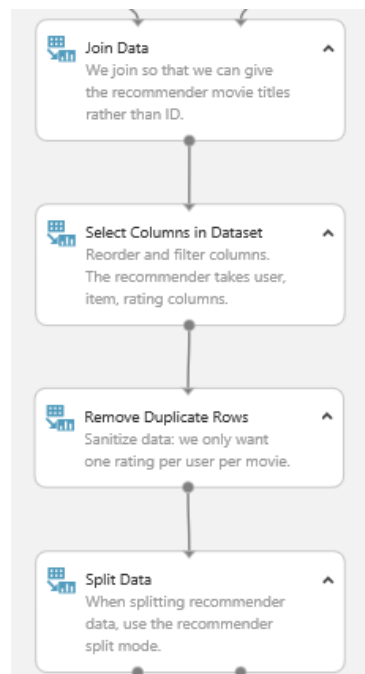
- **Enfoque basado en el contenido:** Ocupa características de los usuarios y de los artículos. Los usuarios pueden ser descritos por propiedades como: edad y género. Los artículos pueden ser descritos por propiedades como: marca y autor.
- **Filtrado colaborativo:** Utiliza sólo identificadores de los usuarios y artículos y obtiene información implícita sobre estos desde una matriz de calificaciones dada por los usuarios a los elementos. Podemos aprender sobre un usuario de los artículos que han clasificado y de otros usuarios que han calificado los mismos artículos.

Dentro del laboratorio de ML es nombrado **Matchbox Recommender**. Utilizaremos un algoritmo de filtro colaborativo, es decir, el modelo aprende de una colección de usuarios quienes ya calificaron un subconjunto de artículos.

1. Para dividir nuestros datos en datos de entrenamiento y datos de prueba del modelo, utilizaremos la herramienta **Split Data** que se encuentra dentro del grupo de *Data Transformation* en la opción **Sample and Split** y la arrastramos al lienzo.



La conectamos su nodo superior con el inferior de **Remove Duplicate Rows**.



Dentro de sus propiedades pondremos las siguientes características una vez que seleccionamos “Recommender Split” como modo de división:

- **Fraction of training-only users:** .75. Esto significa que usaremos el 75% de la base de datos para el entrenamiento y 25% para probar el modelo.

- **Fraction of test-user ratings for testing:** 0.25. Para cada usuario en el grupo de pruebas, retendremos el 25% de las calificaciones de ese usuario para probar el modelo.
- **Fraction of cold users:** 0. “Cold users” son usuarios para los que no tenemos datos previos de entrenamiento. En general, el algoritmo de Matchbox puede utilizar metadatos opcionales para hacer recomendaciones para los usuarios que no hayan calificado ninguna película. Sin embargo, para este problema no tenemos metadatos de usuario, por lo que no vamos a evaluar en los “cold users”.
- **Fraction of cold items:** 0. Utilizaremos los “cold items” como los “cold users” y evaluaremos solamente las películas que se han obtenido de los ratings.
- **Fraction of ignored users:** 0. Algunas veces se quiere probar un algoritmo o propiedad en un subconjunto de la base de datos. Aquí utilizaremos la base de datos de entrenamiento completa.
- **Fraction of ignored items:** 0. Lo mismo para los artículos que para los usuarios.

Al final, las propiedades del módulo **Split Data** deben quedar así:

Properties Project >

Split Data

Splitting mode
Recommender Split

Fraction of training-only...
0.75

Fraction of test user ratin...
0.25

Fraction of cold users
0

Fraction of cold items
0

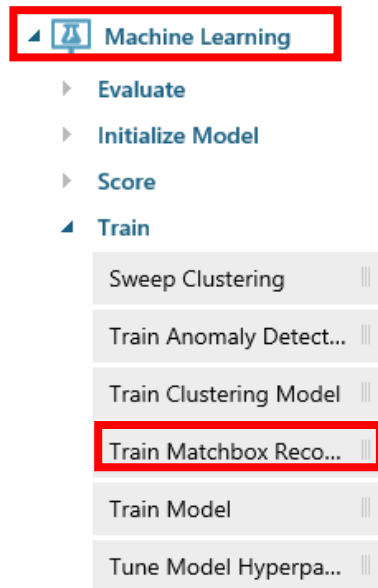
Fraction of ignored users
0

Fraction of ignored items
0

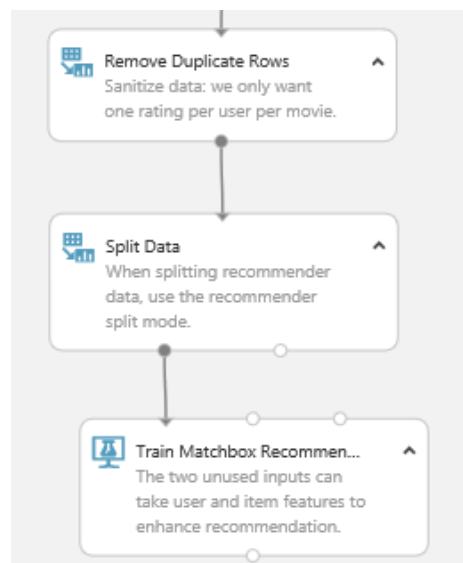
☐ Remove occasionally...

Random seed for Recom...
0

- Ahora pondremos en nuestro experimento el algoritmo. Dentro del grupo *Machine Learning*, encontraremos la opción **Train**, seleccionamos la herramienta **Train Matchbox Recommendation** y lo arrastramos al lienzo.



Lo conectamos con el módulo **Split Data**.



Dentro de sus propiedades pondremos las siguientes características:

- Number of traits:** 20. El número de características latentes que se debe aprender para cada usuario y elemento. Cuanto mayor sea el número de características, más precisas serán las predicciones; sin embargo, el entrenamiento será más lento. Este número suele oscilar entre 2-20.
- Number of recommendation algorithm iterations:** 10. Indica cuantas veces el algoritmo debe procesar los datos de entrada. Igualmente, mientras más grande sea el número, más preciso, pero se tendrá mayor tiempo de entrenamiento.
- Number of training batches:** 4. Indica el número de grupos para dividir los datos durante el entrenamiento.

Las propiedades deben quedar así:

Properties Project >

▲ Train Matchbox Recommender

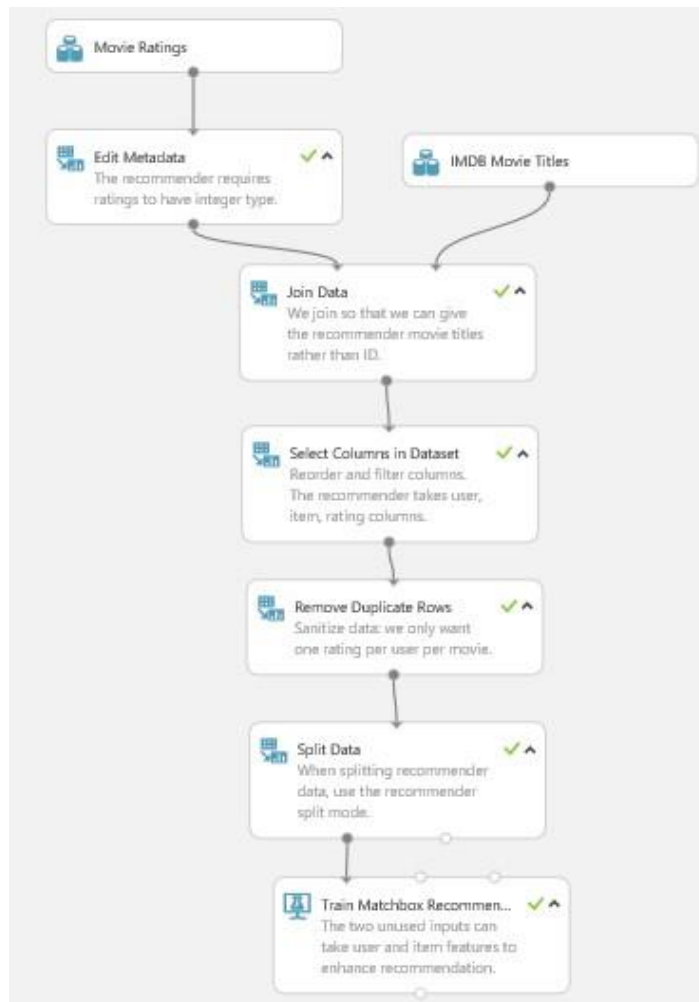
Number of traits

Number of recommendat...

Number of training batch...

3. Damos click en **RUN**, que se encuentra en la parte inferior de la página.

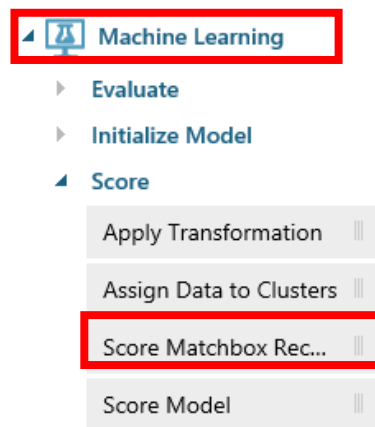
Hasta ahora el experimento nos debe quedar así:



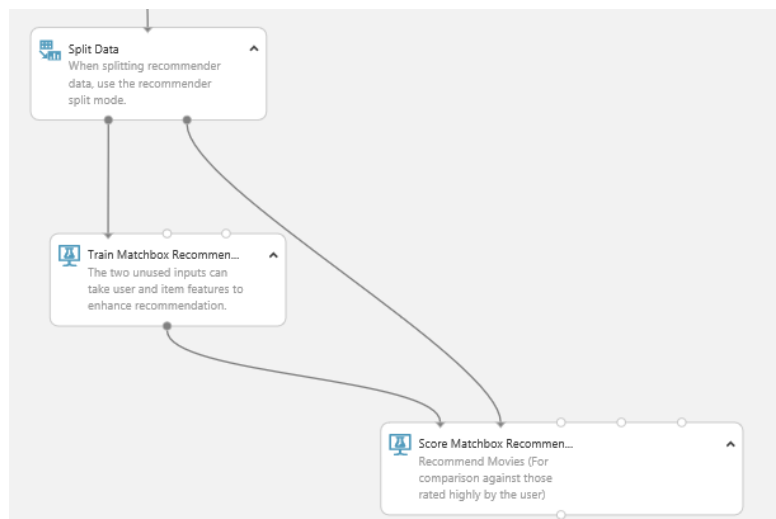
Ejercicio 5: Probar el modelo

Después de que el modelo fue entrenado, usaremos los módulos **Score Matchbox Recommender** y **Evaluate Recommender**.

- **Score Matchbox Recommender**: crea predicciones basadas en un modelo entrenado de recomendación basado en el algoritmo Matchbox.
 - **Evaluate Recommender**: mide la precisión hecha por el modelo de recomendación.
1. Encontramos la herramienta **Score Matchbox Recommender** dentro del grupo *Machine Learning* en la opción **Score** y lo arrastramos al lienzo.



Primero conectamos el nodo izquierdo superior de este con el módulo de **Train Matchbox Recommendation** y alguno de los nodos izquierdos superiores restantes con el nodo inferior derecho de **Split Data**.



Y dentro de sus propiedades ponemos las siguientes características:

- **Recommender prediction kind:** Item Recommendation. En este caso, queremos recomendarles películas a nuestros clientes.
- **Recommender item selection:** From Rated Items. Seleccionamos esta opción ya que se está desarrollando o probando un modelo. Esta opción habilita el modo de evaluación y el módulo sólo hace recomendaciones de los elementos del conjunto de datos de entrada que se han clasificado.
- **Maximum number of items to recommend to a user:** 5. El número de artículos que se le va a recomendar a cada usuario.
- **Minimum size of the recommendation pool for a single user:** 2. Indica el número de cuántas recomendaciones anteriores se requieren. De forma predeterminada, este parámetro se establece en 2, lo que significa que el elemento debe haber sido recomendado por al menos otros dos usuarios. Esta opción sólo debe utilizarse si está anotando en el modo de evaluación. La opción no está disponible si selecciona **From all items** en **Recommender item selection**.

De esta forma las propiedades quedarían así:

Properties Project

▲ Score Matchbox Recommender

Recommender prediction kind
Item Recommendation ▼

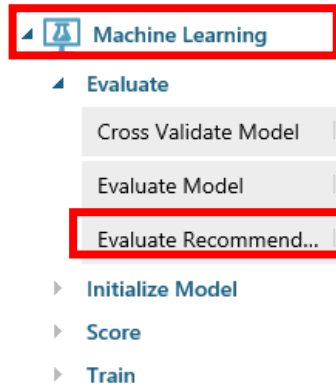
Recommended item selection
From Rated Items (for mod ▼

Maximum number of ite...
5

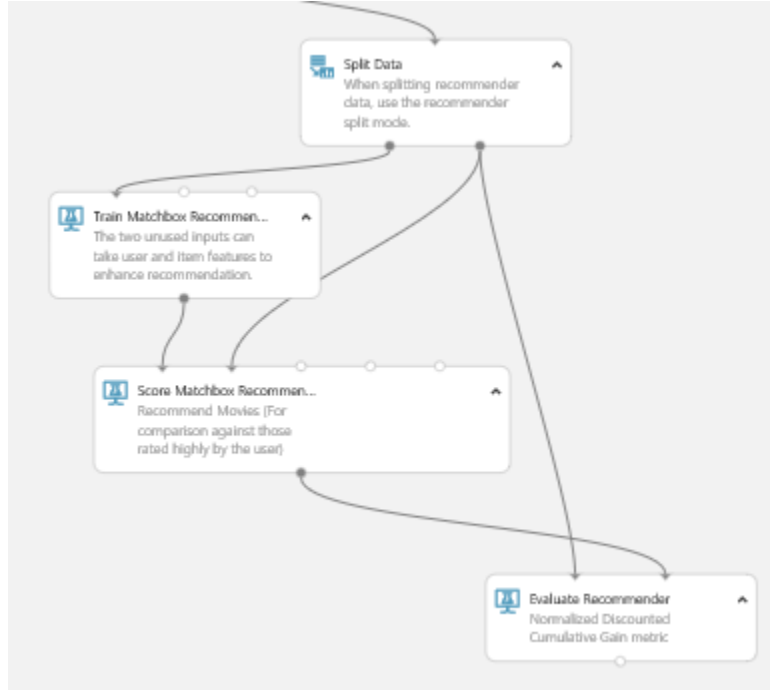
Minimum size of the reco...
2

☐ Whether to return th...

2. Encontramos la herramienta **Evaluate Recommender** dentro del grupo *Machine Learning* en la opción **Evaluate**, la arrastramos al lienzo.



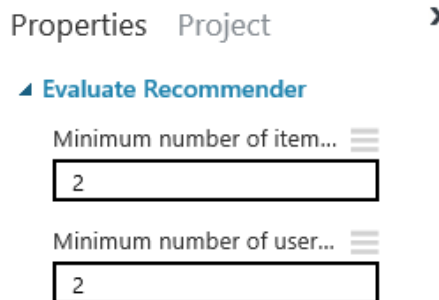
Y conectamos el nodo superior derecho con el nodo inferior derecho de **Score Matchbox Recommender** y el nodo superior izquierdo con el nodo que resta de **Split Data**, que es el nodo donde saldrá la base de entrenamiento.



Dentro de sus propiedades ponemos las siguientes características:

- **Minimum number of items that the query user and the related user must have rated in common:** 2. El número mínimo de artículos que tuvieron que ser calificados por el usuario al que le vamos a recomendar y el usuario relacionado.
- **Minimum number of users that the query item and the related item must have been rated by in common:** 2. El número mínimo de usuarios que tuvieron que calificar el artículo que vamos a recomendar y el artículo relacionado.

Así deben quedar las propiedades del módulo **Evaluate Recommender**:



The screenshot shows a software interface with a tabbed view. The 'Properties' tab is selected, and within it, the 'Evaluate Recommender' section is expanded. Two input fields are visible, both containing the value '2'. The first field is labeled 'Minimum number of item...' and the second is labeled 'Minimum number of user...'. Each label has a small icon of three horizontal lines to its right.

Property	Value
Minimum number of item...	2
Minimum number of user...	2

3. Damos clic en **RUN** que está en la parte superior de la página y esperamos un momento a que corra nuestro experimento.

4. Para visualizar las películas recomendadas damos clic en el nodo inferior de **Score Matchbox Recommender** y seleccionamos **Visualize**. Podemos observar la matriz de recomendaciones, donde cada usuario identificado por su Id tiene al menos 2 recomendaciones de películas.

User	Item 1	Item 2	Item 3	Item 4	Item 5
14887	Inception (2010)	One Flew Over the Cuckoos Nest (1975)	Star Wars Episode V - The Empire Strikes Back (1980)	The Raid 2 Berandal (2014)	There Will Be Blood (2007)
19075	Star Trek Into Darkness (2013)	No se Aceptan Devoluciones (2013)	Man of Steel (2013)	Elysium (2013)	
24752	Argo (2012)	My Week with Marilyn (2011)	The Necessary Death of Charlie Countryman (2013)	The Last Waltz (1978)	Our Idiot Brother (2011)
21707	Django Unchained (2012)	The Wolf of Wall Street (2013)	Capote (2005)	Behind the Candelabra (2013)	
14038	Moonrise Kingdom (2012)	Grizzly Man (2005) Biography	Police Academy Mission to Moscow (1994)	På rymmen med Pippi Långstrump (1970)	Hamilton Men inte om det gäller din dotter (2012)
4072	Beasts of the Southern Wild (2012)	Seeking a Friend for the End of the World (2012)	Step Brothers (2008)		
25065	The Great Gatsby (2013)	Spring Breakers (2012)			
14927	Now You See Me (2013)	Man of Steel (2013)	Oblivion (2013)	A Good Day to Die Hard (2013)	
12564	Red (2010)	Drinking Buddies (2013)	Dark Shadows (2012)	Valentines Day (2010)	New Years Eve (2011)
24871	Exit Through the Gift Shop (2010)	Misery (1990)	Thor The Dark World (2013)	This Is the End (2013)	Running Scared (2006)
20245	Life of Pi (2012)	American Hustle (2013)	Dans la maison (2012)	50/50 (2011)	Todo sobre mi madre (1999)
25743	Gravity (2013)	American Hustle (2013)	Autoreiji Biyondo (2012)		
20132	Oblivion (2013)	Bad Boys II (2003)	Arachnophobia (1990)	Ginger Snaps (2000)	Green Lantern (2011)
26216	Exit Through the Gift Shop (2010)	The Town (2010)	The House I Live In (2012)		

5. Para visualizar las métricas de la evaluación del modelo, damos clic en el nodo superior de **Evaluate Recommender** y seleccionamos a opción **Visualize**. Nos da una métrica llamada por sus siglas en inglés NDCG (normalized discounted cumulative gain) ganancia acumulada descontada normalizada. Con esto se puede medir la calidad del sistema de recomendación. Está basada en las distancias Manhattan y Euclidianas. Es calculada como el promedio de la similitud entre el usuario al que le vamos a dar recomendaciones y un usuario relacionado. En este caso buscamos un número cercano a 1.

NDCG



0.954984