Algoritmo para predecir calificaciones asignadas por usuarios a películas de la plataforma Netflix

Esteban Suárez Restrepo

Bancolombia

Sección Certificación y Calidad de Accesos

Medellín

Febrero 2021

Requisitos para ejecutar el modelo:

El software fue desarrollado con Python 3.9 y el IDE Pycharm 2020.3.3. Para ejecutar correctamente el código este debe ser clonado en el disco D de nuestro equipo (Ejecutar el archivo main.py) y se deben instalar las siguientes librerías:

- pandas
- numpy
- openpyxl
- matplotlib
- sklearn
- glob2

Decidí usar python para este proyecto, ya que es uno de los lenguajes más utilizados para el análisis y minería de datos. Como ventajas de python tenemos la posibilidad de utiilizar librerías especializadas en este campo como lo son Numpy, Pandas, Mlpy y Matploit.

Alcance del proyecto y datos elegidos:

El proyecto está basado en una competencia realizada por Netflix en el año 2007 y cuyo objetivo fue el desarrollo del mejor algoritmo para predecir calificaciones asignadas por usuarios a películas del catálogo de la compañía.

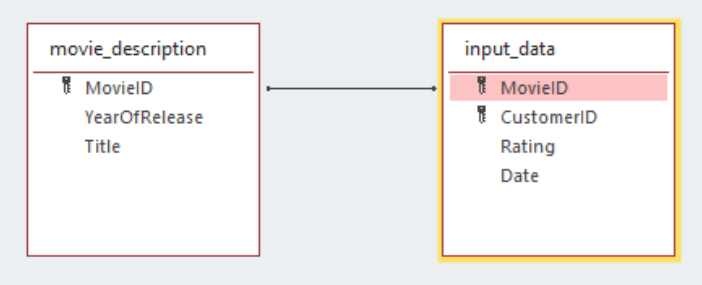
Para abordar el problema anterior utilizaremos un algoritmo de Regresión Logística de la librería sklearn. La Regresión Logística es un algoritmo supervisado y cuyo objetivo es indicar la tendencia de un conjunto de datos discretos, en nuestro caso queremos predecir la posible clasificación asignada a una película Y por un usuario X. Para el correcto funcionamiento del modelo es necesario dividir nuestro Dataset en dos partes, los datos de entrenamiento y los datos de prueba.

La regresión logística es empleada frecuentemente por los científicos de datos, debido a su eficacia, simplicidad y a la fácil interpretación de sus resultados, además no necesita de grandes recursos computacionales.

Datos de entrenamiento			
Campo	**Tipo**	**Tamaño**	**Descripción**
MovieID	Int	Int64	Identificación única de la película
CustomerID	Int	Int64	Identificación única del usuario
Rating	Int	Int64	Calificación asignada por el usuario (1-5)
Date	Datetime	Datetime	Fecha en la que se calificó la pelicula (YYYY-MM-DD)

Nombre de las películas			
Campo	**Tipo**	**Tamaño**	**Descripción**
MovieID	Int	Int64	Identificación única de la película
YearOfRelease	Int	Int64	Año en el que se estrenó la película en DVD
Title	String	String	Nombre de la película

Si analizamos nuestras fuentes de información como una Base de Datos obtendríamos la siguiente relación:



Explorar y evaluar los datos:

El primer paso es aplicar un proceso ETL para la compilación de datos a partir de tres fuentes de información. En este paso utilizaremos la librería "Pandas" para cargar nuestra información en DataFrames. Los datos de entrada se encuentran separados en dos archivos .txt (input_data_*.txt), los cuales concatenamos utilizando la librería "Glob2"

Una vez las fuentes de información estén cargadas, procedemos a estructurar los datos para ser procesados posteriormente por nuestro algoritmo predictivo.

- Separamos por el delimitador ":" empezando por la derecha, reemplazamos valores en blanco por N/A para posteriormente ser filtrados.

```
1:
1488844,3,2005-09-06
822109,5,2005-05-13
885013,4,2005-10-19
30878,4,2005-12-26
823519,3,2004-05-03
```

	0	1	2
1	1	:	1488844,3,2005-09-06
2	1	:	822109,5,2005-05-13
3	1	:	885013,4,2005-10-19
4	1	:	30878,4,2005-12-26
5	1	:	823519,3,2004-05-03

- Separamos por el delimitador "," y modificamos el formato de cada una de las columnas, debido a que el modelo solo permite valores númericos.

	0	1	2
1	1	:	1488844,3,2005-09-06
2	1	:	822109,5,2005-05-13
3	1	:	885013,4,2005-10-19
4	1	:	30878,4,2005-12-26
5	1	:	823519,3,2004-05-03

	MovieID	CustomerID	Rating	Date
1	1	1488844	3	20050906
2	1	822109	5	20050513
3	1	885013	4	20051019
4	1	30878	4	20051226
5	1	823519	3	20040503

Ejecutar ETL para modelar los datos:

Luego de tener los datos procesados, debemos definir las variables de entrada y la variable objetivo de nuestro modelo de predicción.

Variables de entrada = MovieID y CustomerID

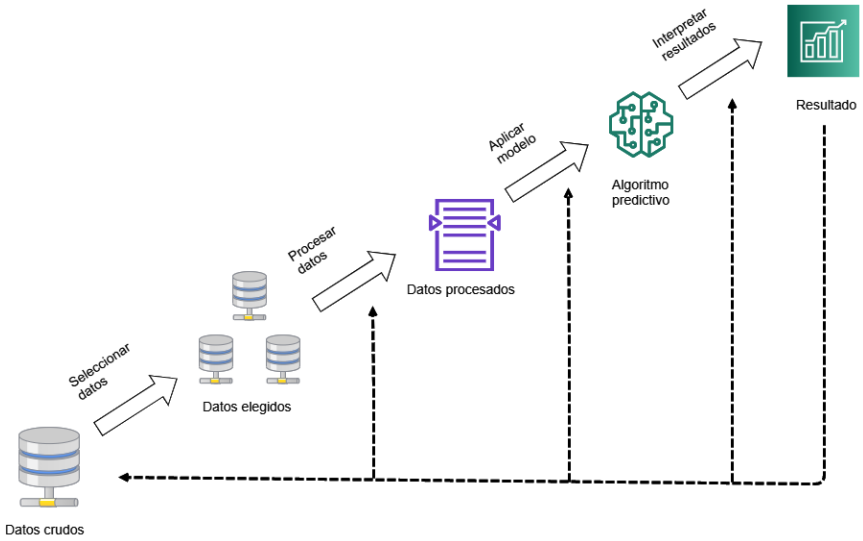
Variable objetivo = Rating

Con el método "train_test_split" de la librería sklearn podremos dividir nuestro dataset en un conjunto de datos de entrenamiento y en un conjunto de datos de prueba. Con este método podemos definir el porcentaje del dataset que utilizaremos para entrenar nuestro modelo y el

porcentaje que utilizaremos para probarlo. Para entrenar nuestro modelo utilizaremos el método “fit()” de la librería sklearn.

Una vez el modelo esté entrenado, podremos utilizar nuestro conjunto de datos de prueba para predecir la calificación que un usuario X asigna a una película Y, usando el método “predict()”. Por último realizamos un “Left Join” con la tabla “movie_titles” para obtener el nombre de cada una de las películas.

	MovieID	CustomerID	Predicted_Rating	Title
0	30	79	5	Something's Gotta Give
1	175	6	4	Reservoir Dogs
2	191	87	5	X2: X-Men United
3	157	6	4	Laird: White Knuckle Extreme
4	83	97	5	Silkwood
5	30	59	5	Something's Gotta Give
6	241	6	4	North by Northwest
7	28	7	4	Lilo and Stitch
8	77	134	5	Congo
9	241	131	5	North by Northwest
10	299	79	4	Bridget Jones's Diary



<u>Completar la redacción del proyecto:</u>

Incluya una descripción de cómo abordaría el problema de manera diferente en los siguientes escenarios:

- Si los datos se incrementaran en 100x.
  R//: Utilizaría la zona de procesos de la LZ, para aprovechar el procesamiento que está ofrece.
- Si las tuberías se ejecutaran diariamente a las 7 de la mañana.
  R//: Podría calendarizar la rutina de Python en la LZ, definiendo la hora y la frecuencia de ejecución y almacenando este resultado en la zona de resultados de la LZ. Otra opción sería utilizar el Programador de tareas de Windows para ejecutar este script en el día y la fecha indicada.
- Si la base de datos necesitara ser accedida por más de 100 personas.
  R//: Validar la posibilidad de ingestión de esta información en la zona de datos crudos de la LZ.

Referencias:

- Amat, J. (noviembre 2020). Regresión logística con Python. https://www.cienciadedatos.net/documentos/py17-regresion-logistica-python.html
- Dhiraj, K. (enero 9 del 2020). Logistic Regression in Python Using Scikit-learn. https://heartbeat.fritz.ai/logistic-regression-in-python-using-scikit-learn-d34e882eebb1
- Galarnyk, M. (septiembre 13 del 2017). Logistic Regression using Python (scikit-learn). https://towardsdatascience.com/logistic-regression-using-python-sklearn-numpy-mnist-handwriting-recognition-matplotlib-a6b31e2b166a