Clasificación con Regresión Logística

Alejandra Verónica López Chiquito[a]

[a]*Universidad Autónoma de Querétaro, Querétraro, México*

Abstract

EL nombre del este modelo de machine learning, regresión logística, puede confundir a muchas personas insinuando que se quiere encontrar relaciones entre variables, sin embargo en este ejercicio se podrá demostrar que este modelo es muy útil para tareas de clasificación, en este caso de un dataset con datos de pacientes con diabetes.

Keywords: Python, Regresión logística, Acurracy, Clasificación

1. Introducción

A diferencia de la regresión lineal, que se utiliza para predecir valores numéricos continuos, la regresión logística se aplica principalmente a problemas de clasificación, donde el objetivo es predecir a qué clase o categoría pertenece una observación. Este modelo es muy utilizado para encontrar la relación entre una variable dependiente binaria (que puede tomar dos valores, generalmente 0 y 1) y una o más variables independientes.

2. Metodología

Así como en la mayoría de los proyectos de machine learning se deberá realizar un pequeño proceso de visualización de los datos, seguido del preprocesamiento del dataset y creación de las variables necesarias, después el entrenamiento del modelo, a continucación la predicción para terminar la evaluación del modelo. En general, será el proceso que se seguirá en este ejercicio.

3. Definiciones

3.1. Regresión

Se puede definir como un método de análisis que se utiliza en estadísticas y en el campo del aprendizaje automático. El objetivo es encontrar relaciones entre variables.

3.2. Clasificación

Proceso para encontrar la clase o categoría a la que pertenece un elemento dadas sus características.

3.3. Regresión logística

Proceso que ayuda a resolver problemas de clasificación binaria.

3.4. Accurracy

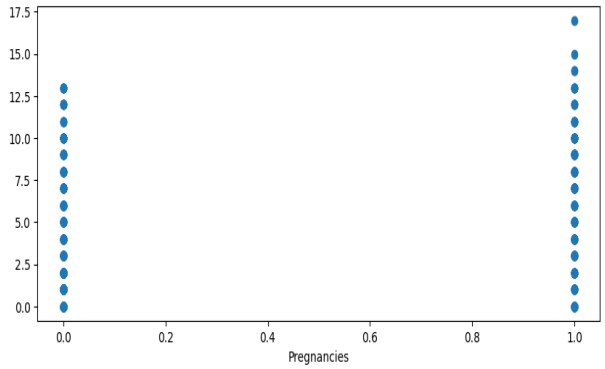
Es una métrica utilizada para evaluar el rendimiento de un modelo de clasificación en aprendizaje automático. Su formula es simple: número de predicciones correctas entre total de predicciones.

4. Diabetes

Como se comento al inicio de este paper, la visualización de los datos se toma como primer paso, ya que obtener las gráficas puede ser de mayor utilidad al querer plantear un objetivo o confirmar una hipotesis.

El dataset cuenta con 8 variables independientes ('Pregnancies', 'Glucose', 'BloodPressure', 'SkinThickness', 'Insulin', 'BMI', 'DiabetesPedigreeFunction', 'Age') y 1 columna como variable dependiente ('Outcome')

Al gráficas cada una de las VI contra la VD, se obtiene la misma forma en todas las gráficas y esto se debe a que la columna objetivo, es decir, Outcome, únicamente cuenta con 2 valores, 0 y 1. En este ejercicio el significado es 0 para paciente sin diabetes y 1 para paciente con diabetes.



Para este tipo de ejercicios de clasificación se requiere de una función de activación, por ejemplo sigmoide. Esta función ayuda al modelo a escalar el resultado de la predicción dentro de un rango entre 0 y 1. Con una simple condición se puede separar las 2 tipos de clases, es decir, si el valor es más cercano a 0, el paciente no tiene diabetes, si el resultado es más cercano a 1, el paciente padece diabetes.

Dividir el dataset en grupos para train y test, hace que el desarrollo de los modelos sea más ligera al momento de entrenar y evaluar los resultados del modelo. Una de las librerias más usadas es la train test split.

Para empezar el training se añade una columna de unos en la

posición 0 de nuestra variable X, es decir, todas nuestras VI.

La función más importante de este modelo es la de entrenamiento, al llamarla es donde se definen el número de iteraciones o épocas, el learning rate, se pasan los datos con los que el modelo aprenderá, es decir un grupo de datos que fue dividido del data set original para que el modelo encuentre los thetas correspondientes para lograr clasificar los resultados. Es aquí donde la función sigmoide se aplica para encoentrar la clase de cada conjunto de datos.

Como último paso se requiere evaluar el modelo. Para esto usa el otro grupo de datos que se dividió, es decir, Xtest y Ytest. Con Xtest se hace el mismo proceso de multiplicación por los thetas, se aplica la función sigmiode y se definde su clase dependiendo de a qué valor está más cercano entre 0 y 1. Este proceso se hace para todos los datos dentro de la variable Xtest.

Una vez terminado se define la formula del acurracy, definida anteriormente en este paper y con este valor, se puede encontrar que tan bueno es nuestro modelo.

Los primeros valores obtenido como acurracy fueron cerca de 0.60. Con el arreglo del learning rate desde 0.01 a 0.000017, las epocas fueron desde 500 hasta 100,000, y el cambio en el valor de los thetas iniciales, de aleatorios a ceros, el acurracy subió a 0.74.

5. Conclusiones

Este dataset es claro ejemplo de que no se depende de la cantidad de datos, sino de la calidad. Los datos dentro de nuestro dataset proporcionaban infomación muy concreta y útil.

El ejercicio de clasificación fue exitoso ya que se logró implementar el modelo matemático de la regresión logística a este conjunto de datos. Seguramente existe técnicas más complejas para mejorar el accurracy de nuestro modelo.