PROYECTO FINAL: MODELO DE CLASIFICACIÓN CON MACHINE LEARNING Y DEEP LEARNING

Maria Paula Alba Gomez (mariap.alba@usa.edu.co)
Maestría en Inteligencia Artificial

Abstract

Se presenta un modelo de clasificación con el dataset Studens in Performace Exams implementando los algoritmos clásicos de machine learning K-Nearest Neighbors (KNN), Support Vector Machine (SVM) y Random Forest (RF) además se crean diferentes modelos de redes neuronales variando el numero de neuronas, optimizador, función de perdida y numero épocas. Se muestran excelentes resultados con cada uno de los modelos de Machine Learning y Deep Learning concluyendo que el modelo predice correctamente con un porcentaje de error bajo.

Introducción

El dataset Students Performance in Exams consiste en las calificaciones que obtuvieron los estudiantes en las materias de matemáticas, lectura y escritura. Este dataset pretende entender la influencia de los antecedentes de los padres, el género, alimentación y la preparación para los exámenes en el desempeño de los estudiantes [1].

A partir de este dataset se realizaron análisis y exploración de los datos donde observan y concluyen que las calificaciones de cada materia tienen una relación lineal entre si [2]. Además, se han realizado modelos de regresión con Random Forest el cual la variable a predecir es el promedio de las calificaciones de las tres materias obteniendo un error de generalización del 1% [3]. Igualmente, para predecir el promedio de las materias se realizó un modelo de regresión lineal donde se obtuvieron resultados favorables [4].

Este proyecto tiene como objetivo la implementación de un modelo de clasificación con el dataset Students Performance in Exams utilizando algoritmos clásicos de machine Learning y algoritmos de Deep Learning.

Problemática.

A partir de los datos que se encuentran en el dataset seleccionado, se pretende predecir si un estudiante puede aprobar un examen mediante un modelo de clasificación teniendo en cuenta la preparación que tuvo para el examen, la alimentación, el género, el nivel de estudios de los padres y el promedio de las notas obtenidas en matemáticas, lectura y escritura.

Exploración de los datos

El conjunto de datos cuenta con ocho características (gender, race/ethnicity, parental level of education, lunch, test preparation course, math score, writing score y Reading score) de las cuales cinco son datos categóricos y tres son datos números. Para el desarrollo de este modelo se adicionaron dos caracteristicas más, mean score que es el promedio de las tres materias y pass test que contiene

números binarios (1 si el promedio es mayor a 65 y 0 en caso contrario).

En la Figura 1 se observa en los histogramas como se encuentran distribuidos los datos numéricos.

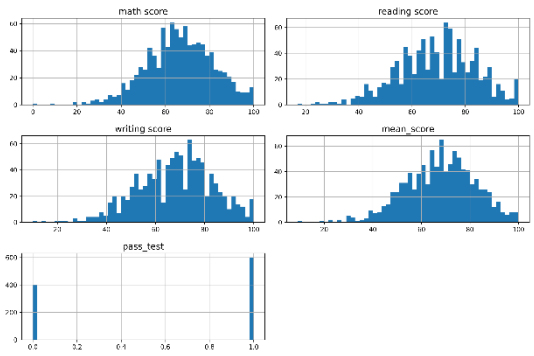

Figura 1. Histogramas datos numéricos.

En la Figura 2 se muestra la matriz de correlación entre las variables numéricas donde se concluye que tienen una relación directamente proporcional con la variable a predecir pass_test.

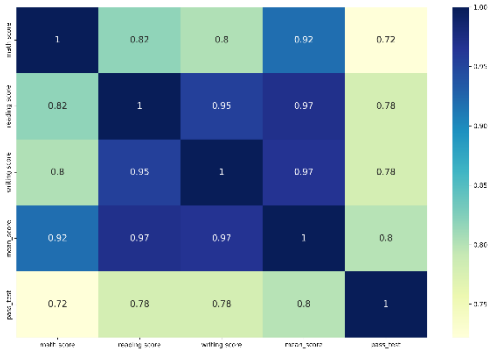

Figura 2. Matriz de Correlación.

A continuación el Figura 3 se confirma la relación directamente proporcional entre las variable numéricas con ayuda de la función *scatter_matrix()*.

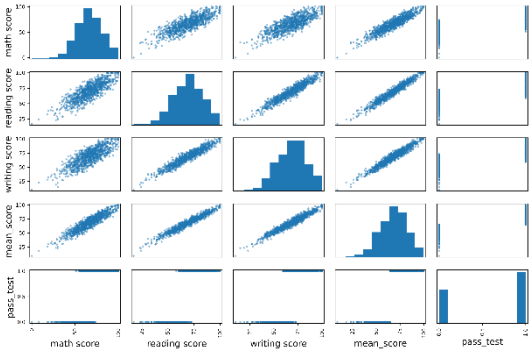

Figura 3. Resultado función *scatter_matrix()*

Preparación de los datos.

División de los datos

El conjuntó de datos se dividió 20% para datos de testeo y 80% para datos de entrenamiento. Al ser un problema de clasificación se estratificaron los datos de la variable a predecir. En la Figura 4 y 5 se observa la variable a predecir pass_test se dividió equitativamente entre los datos de entrenamiento y de testeo.

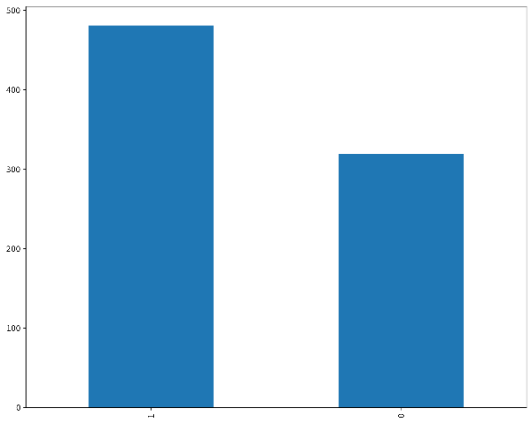

Figura 4. Variable a predecir datos de entrenamiento.

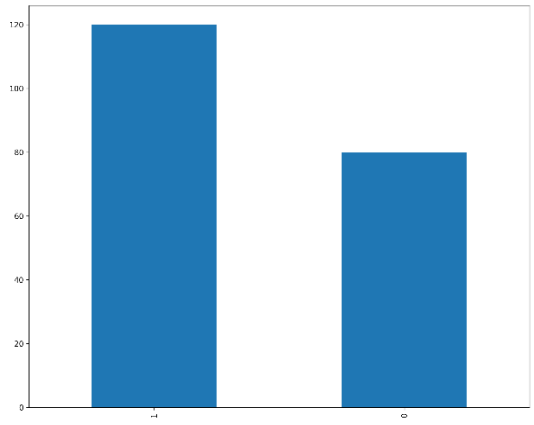

Figura 5. Variable a predecir datos de testeo.

Limpieza de datos

En este caso la limpieza de datos no fue necesaria ya que el dataset no tiene datos nulos. En la Figura 6 se muestra que no hay datos faltantes en el conjunto de datos.

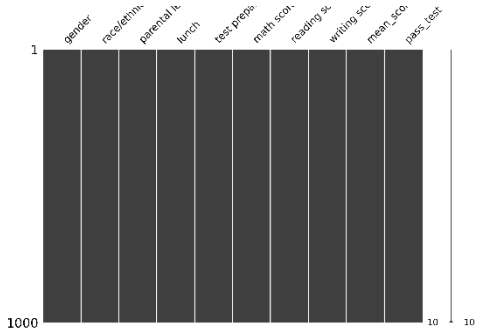

Figura 6. Datos faltantes.

Tratamiento de datos categóricos y Para la conversión de datos categóricos a datos numérico se implementó la estrategia OneHotEncoder que crea una columna nueva por cada categoría existente. Esta estrategia fue aplicada a las columnas gender, race/ethnicity, parental level of education, lunch, test preparation course.

Para normalizar los datos numéricos de aplicó la estrategia StandarScaler a las columnas 'math score','reading score','writing score'.

Modelo Machine Learning.

Entrenamiento

Se entrenó el modelo con los algoritmos de Support Vector Machine (SVM) con kernel gaussiano y polinomial, K-Nearest Neighbors (KNN) y Random Forest (RF) seleccionando este ultimo ya que cuenta con un error de generalización del 2,5%.

La tabla 1 muestra el score obtenido con los datos de testeo en cada modelo.

Algoritmo ML	Score
SVM polinomial	0.97
SVM gaussiano	0.965
KNN	0.965
RF	0.975

Tabla 1. Score en cada modelo

Se realizó una búsqueda de grilla con el algoritmo Random Forest, ya que obtuvo mayor score, para identificar los mejor hiperparametros que permiten disminuir el error de generalización.

Evaluación.

A partir de la búsqueda de grilla se identificó que el mejor modelo es Random Forest con los hiperparametros max_depth = 6 y n_estimators = 100. Al evaluar el modelo con los datos de testeo se obtuvo un Accuracy de 98%.

En la Figura 7 se muestra los resultados de la matriz de confusión.

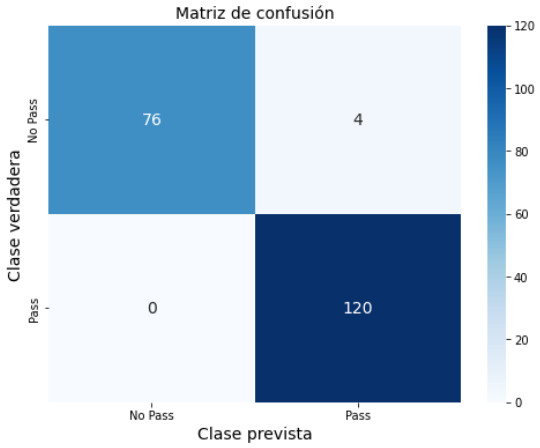

Figura 7. Matriz de confusión.

Modelo Deep Learning

Se probaron varios modelos de redes neuronales con diferentes capas ocultas, variando la función de perdida y el número de épocas.

En la tabla 2 se muestra el Accuracy obtenido en cada uno de los modelos de redes neuronales creados.

Modelo DL	Accuracy
Capas Ocultas: 2, Neuronas: 4, Optimizador: adam, Épocas: 60	99.5%
Capas Ocultas: 1, Neuronas: 32, Optimizador: sgd, Épocas: 20	97.5%
Capas Ocultas: 1, Neuronas: 32, Optimizador: sgd, Épocas: 50	98.5%
Capas Ocultas: 1, Neuronas: 4, Optimizador: adam, Épocas: 60	99%

Tabla 2. Accuracy redes neuronales.

El modelo que presentó mejor resultados se muestra en la Figura 8. Este modelo cuenta con dos capas ocultas con cuatro neuronas, función de activación *relu*, optimizador *adam* y 60 epocas de entrenamiento arrojando un error de generalización del 0.5% obteniendo mejores resultados que en el modelo Machine Learning.

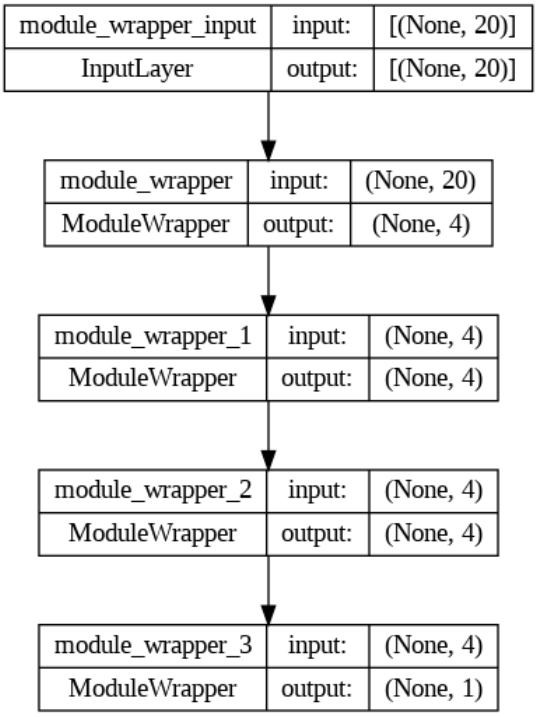

Figura 8. Mejor red neuronal.

La Tabla 3 muestra la comparación de error de generalización del modelo Machine Learning y el modelo Deep Learning.

Modelo	Error de generalización
Machine Learning	2%
Deep Learning	0.5%

Tabla 3. Comparación modelo ML y DL

Conclusión.

El modelo de clasificación tanto en machine Learning como en Deep Learning arrojando muy buenos resultados ya que en ambos las métricas superan el 95% de éxito, es decir que el

modelo se equivoca muy pocas en predecir si un estudiante va a aprobar o reprobar un examen a partir del género, la alimentación, el nivel de estudio de los padres, la preparación para un examen y el promedio de las notas obtenidas en matemáticas, lectura y escritura.

Comparando ambos modelos se concluye que es mejor el modelo con redes neuronales ya que tiene un error de generalización menor que el de Random Forest.

Referencias

1. https://www.kaggle.com/datasets/spscientist/students-performance-in-exams
2. https://www.kaggle.com/code/harshitkumar0240/student-s-performance-e-d-a
3. https://www.kaggle.com/code/maysoon/students-performance-in-exams
4. https://www.kaggle.com/code/tanersekmen/eda-prediction-of-student-performance