Predizione sulla Valutazione Finale degli Studenti

Patrik Brighenti, 143838
Dipartimento di Ingegneria "Enzo Ferrari"
Università degli studi di Modena e Reggio Emilia

21/06/2022

Abstract

Questo progetto ha come obiettivo l'analisi di un dataset di attributi relativi a studenti di due scuole superiori portoghesi. In particolare, si vuole predire se lo studente avrà una votazione finale "bassa (low), media (average) o alta (high)". Per farlo sono stati analizzati ed applicati diversi algoritmi di machine learning, per ottenere una classificazione il più possibile corretta.

1. Introduzione

In questo elaborato, vengono analizzati due dataset distinti, il primo si riferisce alla materia matematica, mentre il secondo alla materia portoghese.
Gli attributi dei dati includono i voti degli studenti, le caratteristiche demografiche, sociali e scolastiche ed è stato raccolto utilizzando relazioni e questionari scolastici.

Uno stesso studente può essere presente in entrambi i set di dati. Questi studenti possono essere identificati cercando attributi identici che caratterizzano ciascuno studente.

2. Analisi dei Dati

Il primo passo per sviluppare un progetto di machine learning è sicuramente l'analisi dei dati. Ad esempio si usano i metodi info() e describe() della libreria Pandas, che permettono di avere una visione generale dei valori nulli, del tipo di dato e di alcune caratteristiche generali di ogni feature.
Successivamente si è passati ad un'analisi approfondita delle varie feature.

2.1 Analisi delle Feature

Il dataset uniti contengono 1046 sample (matematica – 396, portoghese 650) e 30 feature, di seguito saranno descritte:

1- school - scuola dello studente (binario: 'GP' - Gabriel Pereira o 'MS' - Mousinho da Silveira)
2- sex - sesso dello studente (binario: 'F' - femmina o 'M' - maschio)
3- age - età studente (numerico: da 15 a 22)

4- address - tipo di indirizzo di casa dello studente (binario: 'U' - urbano o 'R' - rurale)
5- famsize - dimensione della famiglia (binario: 'LE3' - minore o uguale a 3 o 'GT3' - maggiore di 3)
6- Pstatus - stato di convivenza dei genitori (binario: 'T' - convivente o 'A' - separato)
7- Medu - istruzione della madre (numerico: 0 - nessuno, 1 - istruzione primaria, 2 - istruzione secondaria di I grado, 3 - istruzione secondaria di II grado, 4 - istruzione universitaria)
8- Fedu - istruzione del padre (numerico: 0 - nessuno, 1 - istruzione primaria, 2 - istruzione secondaria di I grado, 3 - istruzione secondaria di II grado, 4 - istruzione universitaria)
9- Mjob - lavoro della madre (nominale: 'insegnante', 'sanitari', 'servizi' civili (es. amministrativo o di polizia), 'a_casa' o 'altro')
10- Fjob - lavoro del padre (nominale: 'insegnante', 'assistenza sanitaria', 'servizi' civili (es. amministrativo o di polizia), 'a_casa' o 'altro')
11- reason - motivo per scegliere questa scuola (nominale: vicino a "casa", "reputazione" della scuola, preferenza "corso" o "altro")
12- guardian - tutore dello studente (nominale: 'madre', 'padre' o 'altro')
13- traveltime - tempo di viaggio da casa a scuola (numerico: 1 - <15 min., 2 - 15 a 30 min., 3 - 30 min. a 1 ora, o 4 - >1 ora)
14- studytime - tempo di studio settimanale (numerico: 1 - <2 ore, 2 - 2 a 5 ore, 3 - 5 a 10 ore o 4 - >10 ore)
15- failures - numero bocciature passate (numerico: n)
16- schoolup - supporto educativo extra (binario: sì o no)
17- famsup - sostegno educativo familiare (binario: sì o no)
18- paid – lezioni extra a pagamento relative alla materia del corso (matematica o portoghese) (binario: sì o no)
19- activities - attività extracurriculari (binario: sì o no)
20- nursery - scuola materna frequentata (binario: sì o no)
21- higher - vuole frequentare un'istruzione superiore(università) (binario: sì o no)
22- internet - Accesso a Internet da casa (binario: sì o no)
23- romantic - con una relazione romantica (binario: sì o no)
24- famrel – qualità delle relazioni familiari (numerico:da 1 – molto male a 5 - eccellente)
25- freetime – tempo libero dopo scuola (numerico: da 1 – molto poco a 5 – molto)
26- goout – esce con gli amici (numerico: da 1 – molto poco a 5 - molto)
27- Dalc – consumo di alcol durante il giorno (numerico: da 1 – molto poco a 5 - molto)
28- Walc – consumo di alcol durante il weekend (numerico: da 1 – molto poco a 5 - molto)
29- health – stato di salute (numerico: da 1 – molto poco a 5 - molto)
30- absences – numero di assenze (numerico: from 0 to 93)

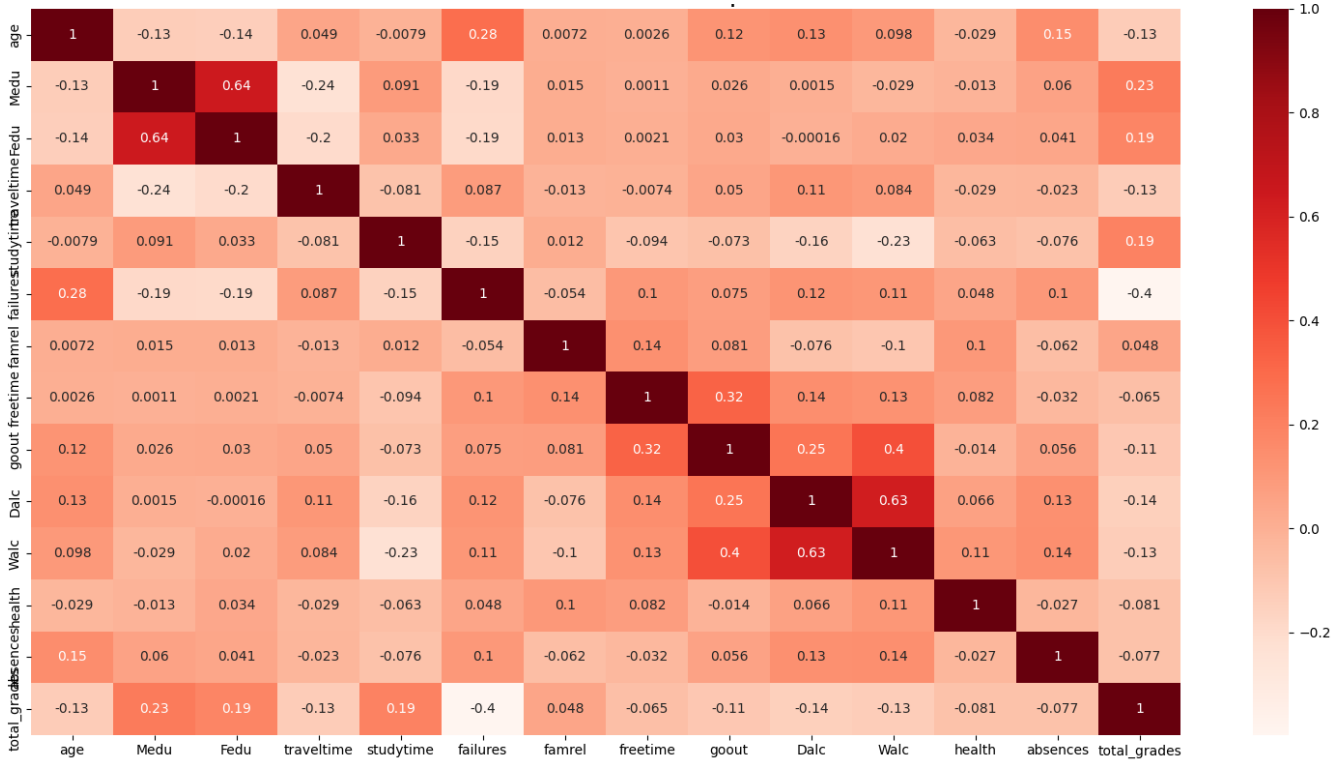
2.2 Analisi delle Label

G1 – Votazione nel I periodo (numerico)
G2 – Votazione nel II periodo (numerico)
G3 – Votazione nel III periodo (numerico)

2.3 Valori nulli

Nel dataset non erano presenti valori nulli.

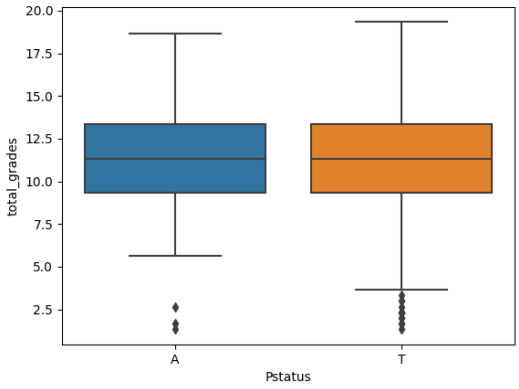
2.4 Matrice di Correlazione

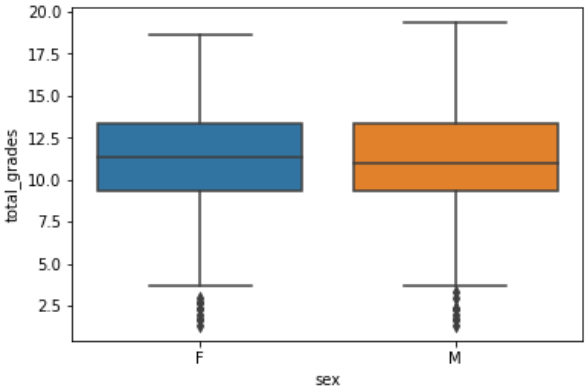


Notiamo un certo grado di correlazione tra le feature Fedu e Medu e tra le feature Dalc e Walc. Anche tra le features Walc e gout abbiamo una certa correlazione.
Decidiamo quindi di mantenere solo le features Fedu e Dalc, andando ad eliminare le features Medu e Walc.

Abbiamo, invero effettuato una feature selection utilizzando un filter method. Abbiamo scelto questo metodo perché indipendente dal modello predittivo scelto.

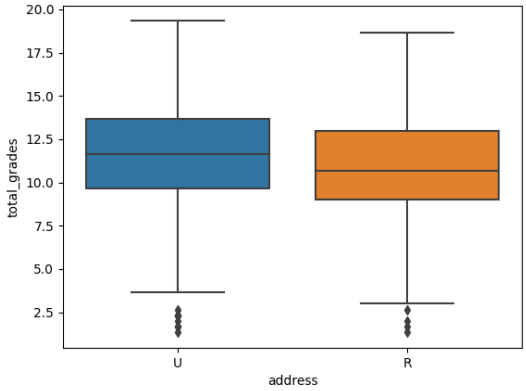
2.5 Correlazione Features e Votazione Finale

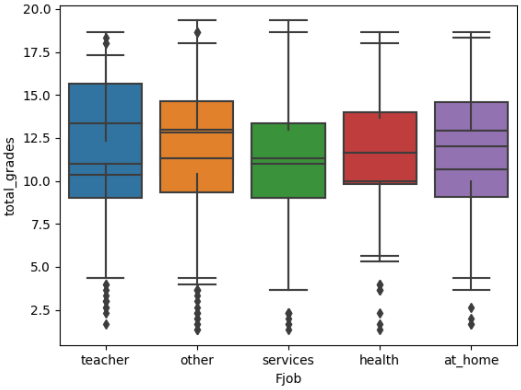


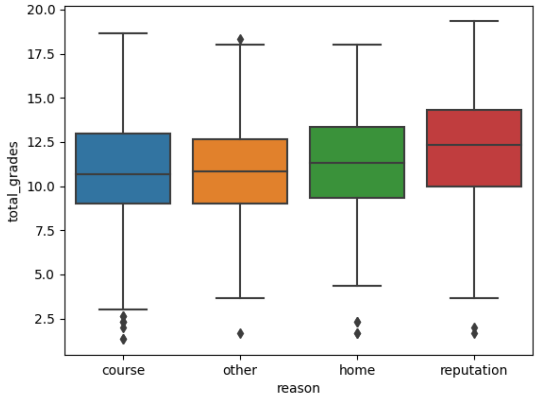


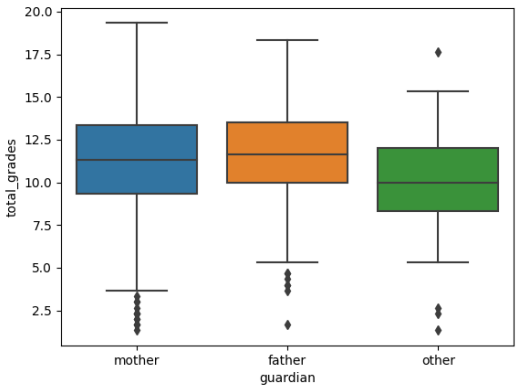
Le feature che non influenzano particolarmente sulla votazione finale sono la situazione dei genitori (convivente/separato) e il sesso dello studente. Decidiamo di eliminare queste features dal dataset.

Di seguito vengono riportate alcune features che incidono significativamente sulla votazione finale.









3. Processing dei Dati

3.1 Trasformazione Dominio delle Label

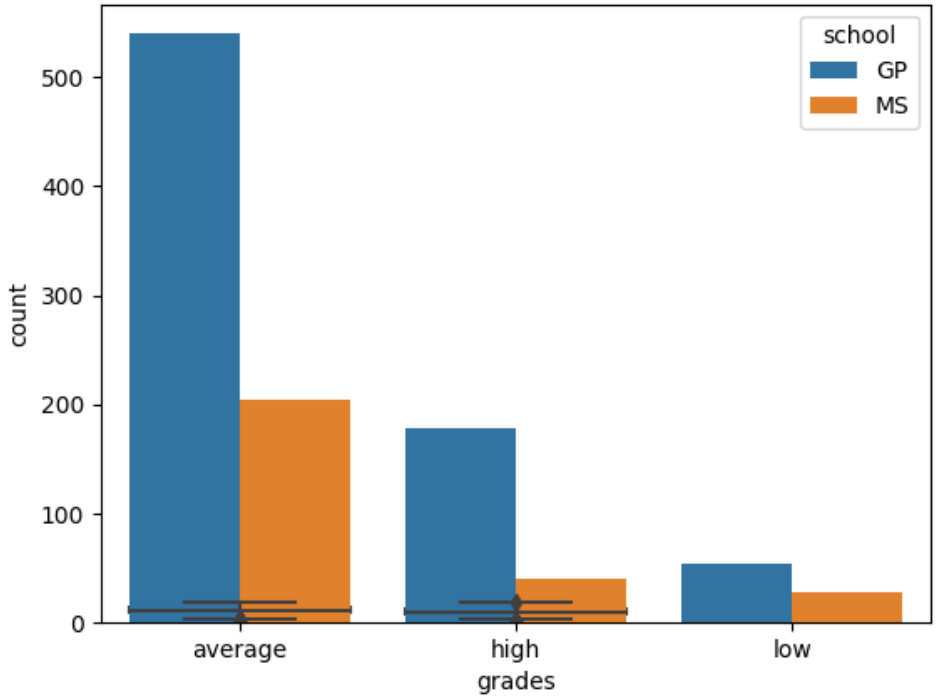
Dopo aver effettuato il merge dei due dataset per ottenerne uno unico, creo un'unica label, come media delle altre tre.

Successivamente trasformo la label da numerica a categorica, dividendola nelle tre classi:
- low (<7)
- average (>=7 & <17)
- high(>=14)

3.2 Trasformazione Features Categoriche

Il passo successivo è stato trasformare le features da categoriche a numeriche.

3.3 Criterio di Splitting



Si è deciso di dedicare l'80% dei sample per il training ed il restante 20% per il test. Per farlo si è utilizzato il metodo train_test_split della libreria sklearn. Visto che il dataset è sbilanciato, si è impostato il parametro stratify per garantire che train e test avessero un numero di sample proporzionato per ciascuna classe. Questa scelta ha alzato e reso più stabile le prestazioni dell'algoritmo. Inoltre, si è impostato il parametro random_state per garantire riproducibilità.

3.4 Scalamento dei dati

Sebbene le features numeriche non siano in range molto differenti tra loro, si è deciso di applicare la normalizzazione per aumentare la velocità di addestramento dei modelli.

4. Modelli

• **Softmax Regression**: Il punto di forza di questo algoritmo è che è facile da implementare ed efficiente in fase di training e predizione. L'inconveniente è la tendenza all'overfitting in caso di pochi sample, problema che abbiamo attenuato con la regolarizzazione.
• **K-Nearest Neighbors:** per l'efficienza in caso di pochi sample e poche features a disposizione, come nel nostro caso.
• **Support Vector Machine**
• **Decision Tree**

Gli ultimi due modelli sono stati inseriti per avere un ulteriore confronto, in modo tale da avere uno studio il più completo possibile.
Per identificare gli iperparametri migliori per ogni modello è stata effettuata una Grid Search, testando i modelli sul dataset di training.

4.1 Ensemble e Cross Validation

Si è deciso di utilizzare questo paradigma, implementato tramite la tecnica di stacking, per migliorare le prestazioni dell'algoritmo.

5. Risultati

Accuracy	0.9138755980861244
Precision	0.8466034438300433
Recall	0.9138755980861244
F1-Score	0.8779397244683953

References

[1] https://archive.ics.uci.edu/ml/datasets/Student+Performance#