GPA Genie:

Integrating Study Habits, Extracurricular Activities, and Parental Information for GPA Classification

Dhruv Chandna & Soham Jain
Dr. Yilmaz Period 1
TJ Machine Learning 1
10/23/2024

Project Overview

- Tool for students to optimize academic and extracurricular planning from an early age
- Enables students to weigh different study habits and activities to maximize academic success
- Our goal is to predict high school students’ GPA by examining relevant factors that can shape their educational outcomes

DATASET

Descriptive Attributes

- Students Performance Dataset
    - Contains demographics, study habits, and extracurricular information for 2,392 high school students
- Target is *GradeClass*, a quantitative discrete variable for GPA classification:
    - **0:** GPA >= 3.5
    - **1:** 3.0 <= GPA < 3.5
    - **2:** 2.5 <= GPA < 3.0
    - **3:** 2.0 <= GPA < 2.5
    - **4:** GPA < 2.0

Attributes

- Students Performance Dataset contains 14 attributes

1. **Student ID** (1001-3392)
2. **Age** (15-18)
3. **Gender**
   - 0: Male
   - 1: Female
4. **Ethnicity**
   - 0: Caucasian
   - 1: African American
   - 2: Asian
   - 3: Other
5. **ParentalEducation**
   - 0: None
   - 1: High School
   - 2: Some College
   - 3: Bachelor's
   - 4: Higher
6. **StudyTimeWeekly** (0.0-20.0)
   - Study time in hours
7. **Absences** (0-30)
8. **Tutoring**
   - 0: No tutoring
   - 1: Receives tutoring
9. **ParentalSupport** (self-evaluated)
   - 0: None
   - 1: Low
   - 2: Moderate
   - 3: High
   - 4: Very High
10. **Extracurricular**
    - 0: No participation
    - 1: Participation
11. **Sports**
    - 0: Does not play sport
    - 1: Plays sport
12. **Music**
    - 0: No music activities
    - 1: Music activities
13. **Volunteering**
    - 0: No music activities
    - 1: Music activities
14. **GPA** (2.0-4.0)

PREPROCESSING

Data Cleaning

- We removed the StudentID and GPA attributes
  - Student ID - no predictive power
  - GPA - class variable is just GPA discretized

Viewer

Relation: Student_Performance_Data

No.	1: StudentID Numeric	2: Age Numeric	3: Gender Numeric	4: Ethnicity Numeric	5: ParentalEducation Numeric	6: StudyTimeWeekly Numeric	7: Absences Numeric	8: Tutoring Numeric	9: ParentalSupport Numeric	10: Extracurricular Numeric	11: Sports Numeric	12: Music Numeric	13: Volunteering Numeric	14: GPA Numeric	15: **GradeClass** Numeric
1	1001.0	17.0	1.0	0.0	2.0	19.833722807854713	7.0	1.0	2.0	0.0	0.0	1.0	0.0	2.9291...	2.0
2	1002.0	18.0	0.0	0.0	1.0	15.40875605584674	0.0	0.0	1.0	0.0	0.0	0.0	0.0	3.0429...	1.0
3	1003.0	15.0	0.0	2.0	3.0	4.21056976881226	26.0	0.0	2.0	0.0	0.0	0.0	0.0	0.1126...	4.0
4	1004.0	17.0	1.0	0.0	3.0	10.028829473958215	14.0	0.0	3.0	1.0	0.0	0.0	0.0	2.0542...	3.0
5	1005.0	17.0	1.0	0.0	2.0	4.6724952729713305	17.0	1.0	3.0	0.0	0.0	0.0	0.0	1.2880...	4.0
6	1006.0	18.0	0.0	0.0	1.0	8.191218545250186	0.0	0.0	1.0	1.0	0.0	0.0	0.0	3.0841...	1.0
7	1007.0	15.0	0.0	1.0	1.0	15.601680474699295	10.0	0.0	3.0	0.0	1.0	0.0	0.0	2.7482...	2.0
8	1008.0	15.0	1.0	1.0	4.0	15.424496305808074	22.0	1.0	1.0	1.0	0.0	0.0	0.0	1.3601...	4.0
9	1009.0	17.0	0.0	0.0	0.0	4.562007558047703	1.0	0.0	2.0	0.0	1.0	0.0	1.0	2.8968...	2.0
10	1010.0	16.0	1.0	0.0	1.0	18.444466363097202	0.0	0.0	3.0	1.0	0.0	0.0	0.0	3.5734...	0.0
11	1011.0	17.0	0.0	0.0	1.0	11.851363655296536	11.0	0.0	1.0	0.0	0.0	0.0	0.0	2.1471...	3.0
12	1012.0	17.0	0.0	0.0	1.0	7.59848581924029	15.0	0.0	2.0	0.0	0.0	0.0	1.0	1.5595...	4.0
13	1013.0	17.0	0.0	1.0	1.0	10.038711615617213	21.0	0.0	3.0	1.0	0.0	0.0	0.0	1.5200...	4.0
14	1014.0	17.0	0.0	1.0	2.0	12.101425068754875	21.0	0.0	4.0	0.0	1.0	0.0	0.0	1.7515...	4.0
15	1015.0	18.0	1.0	0.0	1.0	11.197810636915708	9.0	1.0	2.0	0.0	0.0	0.0	0.0	2.3967...	3.0
16	1016.0	15.0	0.0	0.0	2.0	9.728100710723563	17.0	1.0	0.0	0.0	1.0	0.0	0.0	1.3415...	4.0
17	1017.0	18.0	0.0	3.0	1.0	10.098656081788002	14.0	0.0	2.0	1.0	1.0	0.0	0.0	2.2321...	3.0
18	1018.0	18.0	1.0	0.0	0.0	3.5282382085577235	16.0	1.0	2.0	0.0	0.0	0.0	0.0	1.3844...	4.0
19	1019.0	18.0	0.0	1.0	3.0	16.25465808609359	29.0	0.0	2.0	1.0	0.0	0.0	1.0	0.4695...	4.0

Add instance | Undo | OK | Cance

Normalization

- Min max normalization to ensure all values range between 0 and 1

Before:

Age	Gender	Ethnicity	ParentalEducation	StudyTimeWeekly	Absences	Tutoring	ParentalSupport	Extracurricular	Sports	Music	Volunteering
17	1	0	2	19.833723	7	1	2	0	0	1	0
18	0	0	1	15.408756	0	0	1	0	0	0	0
15	0	2	3	4.210570	26	0	2	0	0	0	0
17	1	0	3	10.028829	14	0	3	1	0	0	0

After:

Age	Gender	Ethnicity	ParentalEducation	StudyTimeWeekly	Absences	Tutoring	ParentalSupport	Extracurricular	Sports	Music	Volunteering
0.666667	1.0	0.000000	0.50	0.992773	0.241379	1.0	0.50	0.0	0.0	1.0	0.0
1.000000	0.0	0.000000	0.25	0.771270	0.000000	0.0	0.25	0.0	0.0	0.0	0.0
0.000000	0.0	0.666667	0.75	0.210718	0.896552	0.0	0.50	0.0	0.0	0.0	0.0
0.666667	1.0	0.000000	0.75	0.501965	0.482759	0.0	0.75	1.0	0.0	0.0	0.0

ATTRIBUTE SELECTION

Method 1: **Ranker + CorrelationAttributeEval**

Threshold : 0.01

Features:
Music, Ethnicity, Age, ParentalEducation, StudyTimeWeekly, Absences, Tutoring

```
Ranked attributes:
 0.017224   11 Music
 0.013468    3 Ethnicity
 0.013074    1 Age
 0.01196     4 ParentalEducation
-0.0002      2 Gender
-0.002799   10 Sports
-0.006036    8 ParentalSupport
-0.007427    9 Extracurricular
-0.016604    5 StudyTimeWeekly
-0.018528    6 Absences
-0.050898    7 Tutoring
```

$$r = \frac{\sum(x_i - \bar{x})(y_i - \bar{y})}{\sqrt{\sum(x_i - \bar{x})^2 \sum(y_i - \bar{y})^2}},$$

Method 2: **Ranker + ReliefFAttributeEval**

Threshold: 0.0015

Features:
Absences, Extracurricular, Music, Ethnicity, Age, StudyTimeWeekly

Evaluates attributes by sampling an instance and the value for the nearest instance of the same and different class

```
Ranked attributes:
 0.005512094    6 Absences
 0.002713589    9 Extracurricular
 0.001505272   11 Music
 0.00049426    10 Sports
 0.000000356    2 Gender
-0.000436036    7 Tutoring
-0.001040747    8 ParentalSupport
-0.001403292    4 ParentalEducation
-0.006471977    3 Ethnicity
-0.006616634    1 Age
-0.006968669    5 StudyTimeWeekly
```

Method 3: **GreedyStepwise + CfsSubsetEval**

Threshold: N/A

Features:
Age, Ethnicity, ParentalEducation, StudyTimeWeekly, Absences, Tutoring, Music

Creates a subset of features that are highly correlated with the class while having low redundancy between them.

```
Selected attributes:
1,3,4,5,6,7,11 : 7
Age
Ethnicity
ParentalEducation
StudyTimeWeekly
Absences
Tutoring
Music
```

Method 4: **Ranker + PrincipalComponents**

Threshold: 0.7

Features:
PCA1 -0.451**A**+0.451**T**-0.365**G**-0.339**V**-0.31**M**
PCA2 -0.482**PS**-0.478**STW**-0.368**T**+0.305**PE**+0.295**V**
PCA3 -0.643**A**-0.471**S**-0.327**STW**-0.299**PE**-0.246**G**

Age, **T**utoring, **G**ender, **V**olunteering, **M**usic, **P**arental**S**upport, **S**tudy**T**ime**W**eekly, **P**arental**E**ducation, **S**ports

```
Ranked attributes:
 0.9067    1 -0.451Age+(
 0.8168    2 -0.482Paren
 0.728     3 -0.643Absen
 0.6412    4 0.64 Ethnic
 0.556     5 0.656Music-
 0.4716    6 0.497Ethnic
 0.389     7 -0.62Extra
 0.3083    8 0.54 Parent
 0.2283    9 0.598StudyT
 0.1494   10 0.578Sports
 0.0727   11 0.518Tutor
 0          12 0.474Age-0
```

Method 5: **AllRetained**

Threshold: N/A

Features:
Age, Gender, Ethnicity, ParentalEducation, StudyTimeWeekly, Absences, Tutoring, ParentalSupport, Extracurricular, Sport, Music, Volunteering

1 ☐ Age
2 ☐ Gender
3 ☐ Ethnicity
4 ☐ ParentalEducation
5 ☐ StudyTimeWeekly
6 ☐ Absences
7 ☐ Tutoring
8 ☐ ParentalSupport
9 ☐ Extracurricular
10 ☐ Sports
11 ☐ Music
12 ☐ Volunteering

TRAIN-VALIDATION-TEST SPLIT

Train-Validation-Test Split

- First, we mapped class values from quantitative to qualitative variables for classification

```
num_to_word = {0: 'zero', 1: 'one', 2: 'two', 3: 'three', 4: 'four'}
```

- Then, we used the train_test_split method from scikit-learn with the 'stratify' parameter to ensure class distributions accurately reflected the original dataset

```
def split_df(df, target_column, train_split=0.8, test_split=0.1, val_split=0.1):
    train_df, temp_df = train_test_split(df, test_size=(1 - train_split), stratify=df[target_column], random_state=42)
    test_df, val_df = train_test_split(temp_df, test_size=(val_split / (test_split + val_split)), stratify=temp_df[target_column], random_state=42)
```

CLASSIFIERS

Classifiers

- **Logistic**
  - Statistical model that uses a logistic function to map predicted values to probabilities, allowing for class predictions
- **MultilayerPerceptron**
  - Neural network that processes input data through hidden layers to produce class predictions
- **Bagging**
  - Ensemble learning method that combines predictions from multiple models trained on random subsets
- **Logistic Model Trees (LMT)**
  - Combines decision trees with logistic regression for classification

RESULTS & DISCUSSION

Highest Accuracy

Achieved highest accuracy with **Logistic** classifier and **AllRetained** attribute selection

- **Accuracy:** 75.7%
- **Recall:** 0.757
- **Precision:** 0.869
- **AUC:** 0.913
- **F1-score:** 0.809

```
=== Summary ===

Correctly Classified Instances         181               75.7322 %
Incorrectly Classified Instances        58               24.2678 %
Kappa statistic                          0.6254
Mean absolute error                      0.1644
Root mean squared error                  0.2802
Relative absolute error                 61.0763 %
Root relative squared error             76.3767 %
Total Number of Instances              239

=== Detailed Accuracy By Class ===

                 TP Rate  FP Rate  Precision  Recall   F-Measure  MCC      ROC Area  PRC Area  Class
                 0.950    0.169    0.852      0.950    0.898      0.788    0.950     0.933     four
                 0.463    0.056    0.633      0.463    0.535      0.464    0.854     0.585     three
                 0.000    0.000    ?          0.000    ?          ?        0.815     0.477     zero
                 0.630    0.052    0.607      0.630    0.618      0.569    0.881     0.576     one
                 0.769    0.080    0.652      0.769    0.706      0.646    0.911     0.672     two
Weighted Avg.    0.757    0.114    ?          0.757    ?          ?        0.913     0.769

=== Confusion Matrix ===

   a   b   c   d   e   <-- classified as
 115   6   0   0   0 |   a = four
  12  19   0   0  10 |   b = three
   3   0   0   7   1 |   c = zero
   4   1   0  17   5 |   d = one
   1   4   0   4  30 |   e = two
```

Confusion Matrix

Actual \ Predicted	0	1	2	3	4
0	0	7	1	0	3
1	0	17	5	1	4
2	0	4	30	4	1
3	0	0	10	19	12
4	0	0	0	6	115

Analysis

- Logistic model captured predictive relationships between input features and the categorical target variable
- Small input volume made *AllRetained* optimal attribute selection method
- Confounding variables (i.e. school, state, etc.) taint results

CONCLUSION

Conclusion & Future Work

- Incorporate features like socioeconomic status, school, sleep patterns, etc.
- Hyperparameter tuning to optimize performance
- Explore tailored frameworks using Google Colab/Jupyter

Thanks!

Any Questions?