IIM BANGALORE'S VISTA
THE INTERNATIONAL BUSINESS SUMMIT

Link to Kaggle Notebook

Vista 2024
Data Beyond Boundaries

Predicting Course Completion for Vidya Vigyan

TEAM - THE MAESTRO

SUBHASHREE KEDIA
SHIVANSH GUPTA
MUSKAN

Exploratory Data Analysis Summary

SURVIVAL ANALYSIS

- This analysis helps predict the likelihood of a candidate completing the course based on their dedication.
- The graph clearly shows that the probability of students completing the course increases with engagement time exceeding 20 hours.

FEATURE INDEPENDENCE

- All columns are **mutually exclusive and collectively exhaustive**, meaning they cover all possible categories without an overlap.
- There is **no multicollinearity** between features which is a good sign for us.

PARTIAL DEPENDENCE PLOTS (PDP)

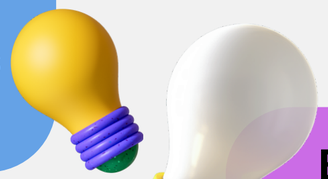
- PDP analysis helps **identify the thresholds for each feature** that contribute to success, refining our understanding of what drives performance.

DATA CONSISTENCY

- The data is **uniform** and doesn't follow a **normal distribution**.
- The data exhibits **high variability** indicating that the data is quite spread out.

ANOVA RESULTS

- **Analysis of Variance (ANOVA)** shows that the features are significantly different for people completing and not completing the course



ACCESS MODE & PATHWAY TYPES

- Surprisingly, the mode of access and the type of learning pathway don't significantly impact the course success, which is very counter-intuitive.

BIVARIATE THRESHOLDS

- Students who engage with the course for **35 to 45 hours** and consume **7 to 10 units of content are more likely to be successful.**
- **Higher content consumption ranges (16 to 95)** & **engagement hours above 95** correlate with success.
- Students who take **more than 4 assessments** and have a **performance metric above 75** also show higher success rates.
- The mode of assessments taken in **Health sector and Arts Sector** shows more dedication towards the course.

Explanation of New Features with Respect to Indian Context

InvolvementMetric (ContentConsumed * AssessmentsTaken)

- **Description:** It reflects the overall involvement of the learner in the course.
- **Indian Context:** Highlights dedication in managing multiple commitments, crucial in competitive educational settings.

- **Description:** It measures the learner's proficiency and consistency in exams.
- **Indian Context:** Reflects readiness and understanding in the academic-centric Indian education system.

ExamProficiencyIndex (AssessmentsTaken * PerformanceMetric)

EngagementIndex (AssessmentsTaken * ProgressPercentage)

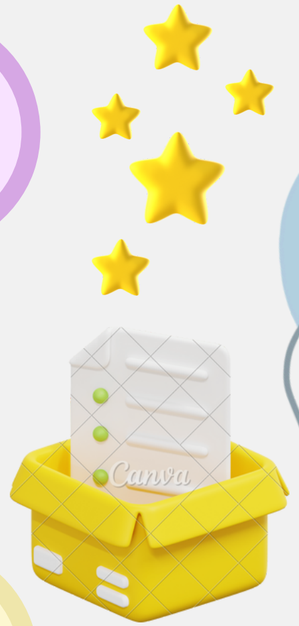
- **Description:** It gauges the learner's engagement level.
- **Indian Context:** Identifies actively progressing learners, crucial for retention and commitment.

- **Description:** It measures overall academic achievement.
- **Indian Context:** Highlights high achievers focusing on grades and completion rates.

AcademicAchievement Factor (PerformanceMetric * ProgressPercentage)

StudyDedicationIndex (ProgressPercentage * EngagementHours)

- **Description:** It assesses the learner's dedication to studying.
- **Indian Context:** Reveals persistence despite external pressures, reflecting commitment to learning.



APPROACH FOR FEATURE ENGN.

- We have validated our generated features by **correlation matrix, and eta square**.
- We have **applied transformation** on the features given to make it smoother for model training.
- We saw no relevance of **AccessMode** as a feature with respect to course success and hence we dropped it.
- We dropped **ExamProficiency** and **AcademicAchievementIndex** to avoid **multicollinearity** in the model training

- In enhancing Vidya Vigyan's predictive model for course completion, we faced challenges due to the **uniformity of the dataset**. This uniformity prevented the creation of an **efficiency feature**, as the **lack of variability** and **correlation among features** made advanced techniques like the **chain rule ineffective**.
- Additionally, attempts to **link access mode and pathway type with engagement hours** didn't yield significant correlations, indicating these factors **didn't influence** engagement as initially hypothesized.

Model Selection

Algorithms Applied

We initially applied seven algorithms: Logistic Regression, Random Forest, Decision Tree, XGBoost, SVM, Neural Network, and Gaussian Naive Bayes, and mapped their accuracy

Performance Challenges

Gaussian Naive Bayes performed poorly due to the lack of a normal distribution in the data, while Logistic Regression also struggled for the same reason. SVM's performance was suboptimal because of issues revealed by t-SNE analysis.

Top Performers

Among these, Random Forest, Decision Tree, XGBoost, and Neural Network demonstrated excellent accuracy in predicting course completion rates.

Ensemble Optimization

To further optimize the model, we employed ensemble methods: a Voting Classifier, a Stacking Classifier with Logistic Regression as the meta-model, and Bagging with XGBoost.

Final Model

Ultimately, the Stacking Classifier was chosen as the final model, as it incorporated features from all top-performing models, ensuring robust and accurate predictions.

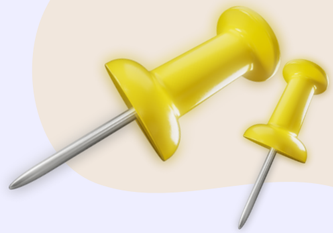
Model Trained on Binary Data

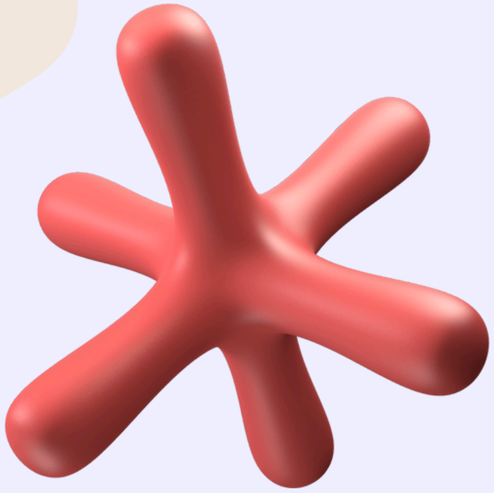
We transfered all the features to **binary features** using the thresholds calculated in PDPs, and then trained **Bernoulli Naive Bayes**, and got a very good accuracy.

Evaluation Metrics

We evaluated our final stacking model using classification report and confusion matrix.We tried ROC AUC but the results were’nt satisfactory.
Accuracy of our model is around 96% which implies it captures the behavior almost perfectly

Problems





Completion Incentive

A significant number of users, despite investing 80-90 hours, fail to complete the course, indicating a potential lack of incentive to finish.

Interaction Deficit

After reaching 60% progress, users might struggle to succeed due to insufficient interaction with the portal.

Course Depth

The courses are not exhaustive enough by Survival analysis, we can say even with engagement hours less than 20, people can complete the course.

Assessment Motivation

Users in Science & Programming are not motivated enough to give assessments.

Challenge Level

The courses might be challenging as we can deduce from average PerformanceMetric, which can result in potentially demotivating users.

Recommendations

GAMIFICATION

Increased Incentives for Key Milestones: Since reaching 50% progress is critical for users, additional incentives should be provided at this point to encourage further engagement.

Performance-Based Leaderboards: Implement leaderboards based on performance metrics. Enhancing competitiveness with rewards and points specifically tied to progress percentage and assessments can significantly boost motivation, as these metrics have shown high relevance according to the correlation matrix.

Threshold-Based Gaming Incentives: Utilize partial dependence plots to identify key thresholds. Crossing these thresholds should unlock gaming incentives, promoting continued user engagement.

UI/UX AND COURSE CONTENT

Interactive Platform: Both the website and mobile apps should be made more interactive to enhance user engagement.

Comprehensive Course Content: Ensure that the course content is exhaustive and closely aligned with the assessments to provide relevant and cohesive learning experiences.

Community Building: Establish a community feature, including a chat box, to foster loyalty and motivation among users.

Course Feedback System: Implement a feedback mechanism to continuously improve the overall quality of the courses.

TARGETED INTERVENTIONS

Behavioral Insights for Personalized Support: Use behavioral data predicted by our model to identify users who are likely to succeed and those who might struggle. Personally contact and motivate the latter group to improve their chances of success.

Personalized Motivational Emails: Send personalized emails to encourage students to complete their courses. These emails should be tailored using insights from a predictive model based on data from previous users.

Appendix

A. Acronyms and Abbreviations

- EDA: Exploratory Data Analysis
- ANOVA: Analysis of Variance
- SVM: Support Vector Machine
- XGBoost: Extreme Gradient Boosting
- η2\eta^2η2: Eta Squared (a measure of effect size)

B. Methodology Descriptions

B.1. Statistical Analysis

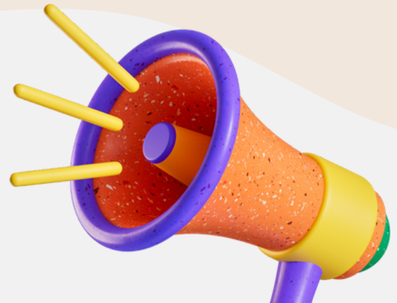
- ANOVA Analysis: Used to compare means among different groups and determine if at least one of them differs significantly.
- Survival Analysis: A set of statistical approaches used to determine the time it takes for an event of interest to occur.

B.2. Feature Engineering Techniques

- Interaction Method: Involves creating features by combining two or more existing features to capture interactions between them.
- Square Root Method: Applies a square root transformation to features to reduce skewness and stabilize variance.
- Squaring and Cubing Methods: These transformations are used to emphasize the differences among feature values.

C. Tools and Libraries

- **Python Libraries:**
  - NumPy: Used for numerical operations and handling arrays.
  - Pandas: A data manipulation and analysis library.
  - Matplotlib & Seaborn: Libraries for data visualization.
  - Scikit-learn: Provides simple and efficient tools for data mining and data analysis.
  - XGBoost: An optimized gradient boosting library designed to be efficient and flexible.



Appendix

D. Model Descriptions

D.1. Machine Learning Models

- Logistic Regression: A regression model used for binary classification tasks.
- Decision Tree: A model that splits the data into branches to make decisions based on input features.
- Random Forest: An ensemble method using multiple decision trees to improve classification accuracy.
- SVM: A supervised learning model that finds the optimal hyperplane to separate classes.
- XGBoost: A scalable and accurate implementation of gradient boosting.
- Gaussian Naive Bayes: A probabilistic classifier based on applying Bayes' theorem with Gaussian assumptions.
- Artificial Neural Network: A computational model inspired by the way biological neural networks work.

D.2. Ensemble Methods

- Voting Classifier: Combines multiple models and makes predictions based on majority voting.
- Stacking: An ensemble learning technique that combines multiple models to improve prediction accuracy.
- Bagging: An ensemble method that improves the stability and accuracy of machine learning algorithms by training multiple versions of the same model on random subsets of the data.

Thank You!