Discussions- After collecting the data we preprocessed it in the excel itself. We then used Jupyter Notebook for running this model. We analyzed our dataset using Describe function to remove any unwanted , NA values or Empty Spaces. We then used One Hot Encoding, Binary Encoding and Label Encoding on the Categorical Variables Dataset to make the Dataset consistent for processing. We then renamed the columns to make the code cleaner. The Dataset was then split into 25 percent test and 75 percent train. Further The fundamental Naive Bayes algorithm was applied to our dataset with the assumption that each feature has an independent and equal outcome contribution. Each variable is taken to be equally contributing to the output while processing making all the edges of the variables in the graph pointing directly towards the outcome variable i.e. (whether the course will be successful?).

We then Created a Gaussian Classifier with-model = GaussianNB(), then trained the model using the training sets and the model created above.

Following that we checked whether any of the element is NaN, and not whether the return value of the any function is a number to clean the dataset of nan, Inf, and missing cells (for skewed datasets). The Dimensions of the input array also were skewed, as the input csv had empty spaces. Finally, a conversion of data frames X and Y into matrices was required. To compare our final output the predict function for target values of X was used, which returned a matrix of predicted values to be compared against with the ground truth labels that is the y\_test and hence, the final accuracy score measured. We then wanted to increase the predicted score, so we then generated a correlation matrix for fine tuning our hyperparameters and get the optimal accuracy for our model.

Challenges Faced

* The data generated was raw and required a lot of preprocessing and had to be made of a similar type i.e.(“*ValueError: Input contains NaN, infinity or a value too large for dtype ('float64').)*”, “*AttributeError: 'numpy.ndarray' object has no attribute 'values'.*”)
* The sum of the C.D.F. of all the variables were not 1 so directly applying the probabilities was difficult.
* The columns and data that were unwanted or were wrongly filled had to be dropped and only the data that was accurately filled had to be taken.
* At the end, the accuracy was coming out to be 60 percent we had to do a lot of fine tuning to find out the optimal values of the hyperparameters.

Innovations

We had assumed the input variables to be independent in the model, but we can make interconnected and accordingly take the weights of one variable on another and find their interdependency relations.

Results-

The Final Accuracy achieved after hyperparameter tuning was found out be 72.22%.

The Figure 1 given below shows how the input variables are inter-related and their effect on the output.

Chart, treemap chart

Description automatically generated

Fig-1 Correlation Matrix

Given Below are the Encoded names of the columns in the Correlation Matrix-

Columns = { ' Comm\_Platform\_Satisfaction(1-Yes, 0-No)':'A', 'Pref\_Live\_Lectures':'B', 'Pref\_Audio\_Lectures':'C', 'Pref\_Recorded\_Content':'D', 'Pref\_PowerPoint\_presentation':'E', 'Pref\_Interactive\_Sessions':'F', 'PercentileAbleTo\_Score':'G', 'Question\_Difficulty':'H', 'Curr\_Live\_Lectures':'I', 'Curr\_Audio\_Lectures':'J', 'Curr\_Recorded\_Content':'K', 'Curr\_PowerPoint\_presentation':'L', 'Curr\_Interactive\_Sessions':'M', 'PlatformAccessEase(1-Easy, 2-Medium, 3-Hard)':'N', 'platform\_UI\_intuitive \_easy-to-use':'O', 'E-Learning\_feature\_incorporate':'P', 'THA':'Q', 'PracticeExercises':'R', 'ReadingMaterials':'S', 'QuizSolutions':'T', 'TextBooks':'U', 'Research Papers':'V', 'Materials\_provision\_Platform':'W', 'Tests\_Assignments\_in\_course':'X', 'Result\_Show\_Time':'Y', 'AvgDur\_Tests':'Z', 'WeeklyHours\_Browse\_Platform':'AA', 'Portal\_Login\_Freq':'AB', 'CourseProgress\_Satisfaction\_x':'AC', 'Prim\_Comm\_Platform':'AD'}, inplace = True)

In the above correlation matrix, a warm-cool color scheme has been used where the warmth of the color increases the positive correlation between the 2 variables. The number inside that column is the impact of the increase in 1 input variable on another input variable.

As the color scheme turns towards dark blue it gives us the negative correlation, and the number inside the box denotes that amount of decrement of one variable due to increment of another variable.

The current ways used by the instructor to deliver the lectures i.e.('I', 'J', 'K', 'L', 'M', 'X', 'Y', 'Z') had zero correlation and impact on the other input variables and output, hence could easily be dropped.

The variable that is the percentage scored by the student was found to decrease with an increase in the Take-Home Assignments provided, and found to increase with the amount of time spent by the student in browsing the E-Learning Platform 'AA'.

We further found out that the ease of accessing the platform ‘N’ was increasing as the Platform got a more intuitive User Interface ’O’

The provision of Take-Home Assignment 'Q' decreased the browsing time of the student on the platform.

It was found out that when the Reading Materials ’S’ are provided then mostly the Quiz Solutions ‘T’ are also provided along with that.

Our final results was that the decision variable was most positively affected by the Percentile Score of the Student 'G'(0.5) and most negatively affected by the number of Take-Home Assignments present in the course 'Q', The Number of Reading Materials 'S', and the provision of Quiz Solutions 'T' (0.54, 0.52, 0.59). The variables like Communication Platform Satisfaction, 'A', Amount of Preference towards Live Lectures 'B', The Ease Of Accessing the Platform 'N', The provision of Practice Exercises ‘R’ and Research Papers ‘V’ on the platform had a minimal effect on the output.