When I was in my final year of B.Tech, as a part of academics, I had to do my "Major Project" which contains half of the semester's credit points. So I chosen a topic on "Predictive analytics of Student academic Performance in STEM Education".

This Project is related to machine learning. In this, we create a dataframe, from the input of a table(CSV file) containing 1000 datasets, to predict a student's academic performance (marks grade) , which depends on some 8 attributes : math\_score, reading\_score, writing score, gender, parental level education, test preparation course, lunch type, ethnicity group.

We feature engineer the attributes math\_socre, reading \_score, writing\_score...That is, we set a pass criteria of 40 marks for these subjects and define new attributes Pass\_math, Pass\_read, Pass\_write ...(pass/fail status for each subject). We also derive the total\_score, percentage, status (overall pass/fail), grade using the same.

Now the dataframe has 15 attributes in which some are non numeric. So we label encode all the attributes (convert them to numeric) to ensure the dataframe ready as a ML model.

We assign x component as dataframe with 14 attributes...all columns except the marks grade.

We assign y component as dataframe with 1 colum...marks grade.

Now, we take 750 rows and apply the x component and corresponding y component...we call it as x\_train and y\_train. The selection of rows is done randomly by train\_test\_split.

As we see, the nature of x\_train.. that has 14 columns and 750 rows. And the values are numeric with different ranges. To improve the performance of model we normalize the x\_component values using min-max scaler....( To transform features to be on a similar scale we do normalization. This improves the performance and training stability of the model.)

We apply Principal component analysis (PCA) which simplifies the complexity in high-dimensional data while retaining trends and patterns. It does this by transforming the data into fewer dimensions, which act as summaries of features.

We fit the x\_train and y\_train to data models obtained by ML classification algorithms . Now , we get y\_pred (Predictive y\_test values) if we input x\_test to the model, which was trained by x\_train and y\_train.

Then, we compare the accuracy of y\_pred with the actual y\_test ( remaining 250 rows with single colum which is grade of student).

The confusion matrix tells the grade mark predicted Vs actual grade of student.

I used Logistic Regression, Random Forest, Decision Forest, Support Vector Machine as classification algorithms.Out of these, the Decision Forest algorithm gave 100% accuracy in predicting the grade attribute.

This model can also be used to predict performance of candidates in various organisations, with custom attributes. This project, can be extended to real time exams and predict performances.