International Journal of Scientific Research in Engineering and Management (IJSREM)
Volume: 04 Issue: 08 | August -2020 ISSN: 2582-3930

Machine Learning Techniques for Intervention Prediction and Progressive Learning

Anand M, Asst. Professor, Dept of Information Science & Engineering,

GSSSIETW, Mysuru

Abstract-Machine learning algorithms have many applications in supporting target intervention approaches. The primary goals of this project are to determine the impact of process-level information, predict the final grade of the students based on the early tests, illustrate the concept of progressive learning, and to establish a perfect prediction of intervention assessment using state of the art techniques. Interventions mean the meetings between student and instructor, recommendation of study techniques and other extra instructions on particular topics. In order to assess this intervention time accurately, and to make a considerable impact in the module of progressive learning, we consider a specific course and its rubric, i.e., the traditional teaching methods, like class tests and cumulative grades and we merge them with process-level information such as grade weightage for online quizzes. Using some intrinsic techniques in machine learning, we eliminate the data which are unsuitable for prediction analysis. Since we have a target variable to predict the data, we use supervised learning. The various techniques that are used for these predictions are univariate linear regression, multivariate linear regression, and ridge regression. We make two types of predictions namely post-hoc and temporal and we show that process-level information has no considerable effects on post-hoc predictions while can be a deciding factor in temporal predictions. The aim of the project is to show that simple machine learning methods like linear regression are enough to generate a great deal of advancement in the concept of progressive learning as we can predict the final grade on or before the two-thirds of the course. This ties up to the concept of intervention and enables the student to improve further when his/her marks in the assessments fall below the bridge. Though some of the test post-hoc models may not perform as expected, but with large scale implementation of the course structure, and with the increase in data, these models can be bettered.

1. INTRODUCTION

Machine Learning

Machine learning is a subfield of computer science that evolved from the study of pattern recognition and computational learning theory in artificial intelligence. Machine learning explores the study and construction of algorithms that can learn from and make predictions on data. Such algorithms operate by building a model from example inputs in order to make data-driven predictions or decisions, rather than following strictly static program instructions. Machine learning is closely related to computational statistics; a discipline that aims at the design of algorithm for implementing statistical methods on computers. It has strong ties to mathematical optimization, which delivers methods, theory and application domains to the field. Machine learning is employed in a range of computing tasks where designing and programming explicit algorithms is infeasible.

Related Works

Here we are proposing a project to predict the accurate time for intervention and the final course grade using regression and classification prediction algorithms during a particular course progression in a university setting. Interventions mean the meetings between student and instructor, recommendation of study techniques and other extra instructions on particular topics

Methods Used for Predictions

Firstly, this work can be divided into two categories on the basis of techniques used: Regression models in order to predict the course grade based on a suitable set of features, and Classification models to derive temporal and post-hoc interventions. A synthetic data set containing 60 students is taken for analysis. Then, this data set is cleaned using Microsoft Excel, i.e. to check for any missing values. Since this is a complete data set, those problems never did arise. Now, according to our fundamental analysis, single linear regression is done on three different variables: Average of all the quizzes, average of all the others excluding the quizzes, and average of high-error quizzes.

Since it is a pretty raw dataset, we should find the high-error quizzes. To calculate them, the mean of each quiz was taken, and the Sum Squared Error (SSE) was done. The quizzes with a higher SSE were considered as high-error quizzes because they deviate more from the mean.

Next, the above three variables are used to compute univariate linear regression. Following that, multiple variables are used to improve our accuracy over the test data. To establish suitable weightages to the features taken, a considerable L1 penalty is used to create a ridge regression model. Finally, more features are taken, like all the quizzes individually as a group, and a model is constructed. With or without an L1 penalty, this model is a real over-fit because it crosses the $N > 2^d$ boundary (N - Number of Training examples, d – Total Number of Features or Dimensions). A model with the lowest RSS (Residual Sum of Squares) and relatively low error rate is chosen for predicting a particular student's final marks.

Now, the next module is intervention assessment (or) prediction. An intervention helps the student to perform better in the future throughout the course. The first method goes temporally and tries to check after every quiz, and conduct an intervention. The problem with the method is that it can lead to a lot of intervention session in a semester, and it is wastage of time for the class instructor. Alternative approach would be to predict and hold intervention at the correct interval of time. Three warnings are given before a

session with the instructor is held, and succeeding the third warning, the intervention is held.

The final module is for assessing final grade based on temporal projections. For every instance of tests throughout the curriculum, the final grade is predicted. This is done through all the instances, thereby, increasing the number of coefficients as the course progresses. That is one of the reasons why this module is named as Progressive Learning. The progress of a student is predicted, and therefore changes, after every instance of assessment. This gives us a much better idea on by which class can we predicted the final grade with the least error.

Course Description

For this project, we take a university level course. This course contains 33 classes with 22 online quizzes, 2 case studies, 3 assignments and 3 mid-term examinations which can be used for our prediction for final course grade and intervention timing. We have 31 instances for a course over the period of its completion. If we exclude the final grade, we have 30 of them. We have 22 quizzes which can be a main factor in the prediction of our final course grade. If we consider a real life scenario, professors say that if you study everything in a day wise schedule, then you would be the forerunner for being the top student of your class. Taking the same concept into machine learning, we are going to show that day-to-day improvement in quizzes of the students would enable them to receive a higher course grade than without day to day assessments. Since we have 22 quizzes, factoring the different between top students and average students would be a prolonged task. Therefore, in order to correct his situation, we create another feature set that has only limited number of quizzes. These quizzes are selected based on their relative difficulty based on the performance of the students. If there is a large variability in the scores of the quiz for all the students that quiz, is considered to be a tough quiz and these type of quizzes are collected into a different feature set. Along with these 30 features, we derive 5 agglomerative features. From the rubric above, these are quiz average, case study average, assignment average, mid-term tests average and high-error quizzes average. These agglomerative features are also used for predicting the course grade.

2. LITERATURE SURVEY

The Elements of Statistical Learning (Data mining, Inference and Prediction)

Authors: Trevor Hastie, Robert Tibshirani, and Jerome Friedman.

Supervised Learning:

For each there is a set of variables that might be denoted as inputs, which are measured or preset. These have some influence on one or more outputs. For each example the goal is to use the inputs to predict the values of the outputs. This exercise is called supervised learning.

We have used the more modern language of machine learning. In the statistical literature the inputs are often called the predictors, a term we will use interchangeably with inputs, and more classically the independent variables. In the pattern recognition literature the term features is preferred, which we use as well. The outputs are called the responses, or classically the dependent variables.

Regression:

A linear regression model assumes that the regression function E(Y |X) is linear in the inputs $X_1, \ldots, X_p$. For prediction purposes they can sometimes outperform fancier nonlinear models, especially in situations with small numbers of training cases, low signal-to-noise ratio or sparse data. Finally, linear methods can be applied to transformations of the inputs and this considerably expands their scope.

The most popular estimation method is least **squares, in which we pick the coefficients β = (β0, β1, . . . , βp)^T** to minimize the residual sum of squares. RSS $(\beta) = \sum_{i=1}^{N} (y_i - f(x)_i)^2$

Classification:

The logistic regression model arises from the desire to model the posterior probabilities of the K classes via linear functions in x, while at the same time ensuring that they sum to one and remain in [0, 1]. The model has the form

$$\log \frac{Pr(G=k\,1|X=s)}{Pr(G=k|X=s)} = (K\,1)0 + {}_{KT\ 1}{}^{X}$$

The model is specified in terms of K − 1 log-odds or logit transformations (reflecting the constraint that the probabilities sum to one). Although the model uses the last class as the denominator in the odds-ratios, the choice of denominator is arbitrary in that the estimates are equi-variant under this choice.

Assessing Intervention Timing in Computer-Based Education Using Machine Learning Algorithms

Authors: Alexander J Stimpson and Mary L. Cummings.

The use of computer-based and online education systems has made new data available that can describe the temporal and process-level progression of learning. To date, machine learning research has not considered the impacts of these properties on the machine learning prediction task in educational settings. Machine learning algorithms may have applications in supporting targeted intervention approaches. The goals of this paper are to: 1) determine the impact of process-level information on machine learning prediction results and 2) establish the effect of type of machine learning algorithm used on prediction results. Data were collected from a university level course in human factors engineering (N = 35), which included both traditional classroom assessment and computer-based assessment methods. A set of common regression and classification algorithms were applied to the data to predict final course score. The overall prediction accuracy as well as the chronological progression of prediction accuracy was analyzed for each algorithm. Simple machine learning algorithms (linear regression, logistic regression) had comparable performance with more complex methods (support vector machines, artificial neural networks). Process-level information was not useful in post-hoc predictions, but contributed significantly to allowing for accurate predictions to be made earlier in the

course. Process level information provides useful prediction features for development of targeted intervention techniques, as it allows more accurate predictions to be made earlier in the course. For small course data sets, the prediction accuracy and simplicity of linear regression and logistic regression make these methods preferable to more complex algorithms.

3. System Analysis

Existing System

Existing systems generally predict only the final course grade. To do so, they use complex machine learning methods like Support Vector Machines or Artificial Neural Networks. Though these methods are feasible, for a small data set such as a class of students, these methods do not prove beneficial. The idea of intervention assessment is usually done manually and doing so leads to improper predictions in the timing. This wrong prediction may lead to the drastic degradation in the final grade of a student. Almost all existing systems do not use the concept of machine learning to predict this intervention timing.

Drawbacks of Existing System

- Existing systems use complex machine learning methods for small data sets.
- Interventions are usually done manually, but not by machine learning algorithms.
- The existing system is only good for predicting the final grade of a course in most of the cases, but the concept of intervention is negligible.
- Existing systems have singular goals that are done according to the instructions given.
- Even final grade prediction is done using a wide range of features, which would lead to over-fit.
- The basic concept of Progressive Learning is existent, but not yet implemented over a complete course.

Proposed System:

The proposed system for **"**Intervention prediction using Machine Learning techniques**"** takes the rubric of a course along with the data set of the students in order to assess the final grades based on their progress throughout the course, i.e., in various quizzes, class tests, problem sets, case studies, etc. At regular intervals of the course, the system evaluates the information of each student, predicting the final grade as well as assessing the perfect time at which an intervention could more effective.

Advantages of Proposed System:

- This system uses simple machine learning methods such as linear and logistic regression.
- Through its up to the mark predictions, the variations between using complex methods are compared to show that simple methods prove to be the best approach for small data sets.
- The error rate also decreases as the course progresses because we use Sum Squared Error (SSE) for linear.
- The intervention assessment module is very effective because it gives the teacher the right time to interact with the student for improving upon his/her progress in/during the course.
- To show that simple methods are better than complex methods, all the primary machine learning methods are used and the comparison is made among them.
- The concept of Progressive Learning helps both the student and teacher to evaluate the situation at the end of the course.

4. System Requirements

Functional Requirements:

Functional requirements specify which outputs should be produced from the given inputs. They describe the relationship between the input and output of the system, and for each functional requirement, a detailed description of all data inputs and their source and the range of valid outputs must be specified.

Input:

1) A course rubric:
   The course rubric consists of the total features that we are willing to provide to our system. In this case, a set of online quizzes, class room tests, problem sets, case studies etc. would be rubric. Use these features, we are going to assess and create a perfect feature set that allows us to make accurate predictions.
2) Students Data Set:
   A data set in this case would be the details of the entire class strength starting from their primary details like name and ID extending up to marks in all the different quizzes and tests. This data set is going to be a raw data set, i.e., a data set which hasn't been preprocessed in given as an input to the system.
3) Feature set:
   After applying some machine learning strategies to find the best cost function and best fit required for the feature set, we take the features or combination of features that accumulate for the most weightage required for accurate predictions. This feature set (all quizzes) is used throughout the system in order to predict the intervention time.
4) Queries:
   The queries based on the quizzes or the details of the students are taken in order to predict the final grade using the best machine learning fit or to predict the timing of the intervention.

Output:

1) Preprocessed Data Set:
   Since the raw data set is given as an input, it contains more unwanted data which may hinder the analysis process. Therefore, data preprocessing techniques are applied by the system return data that can be used clearly for prediction analysis. For example, if there are missing values in the mark fields, regression analysis may not work because missing values. So, based on these exceptions, and using the concept of excused and unexcused absences, we render the data to be more suitable for analysis.
2) The final course grade:

From the first quiz, the system computes and stores the predicted results in the database. Because of the large data set and high range feature set, the prediction of the final grade after the first quiz may have a very high error value. But as the course progresses, this error is reduced and our main goal is to predict this accurately by have of the tests are done (exactly at the half way point of the course.).

3) Intervention prediction:
The main aim of our project is to predict a perfect timing for intervention in order to improve the scores in quizzes and other assessments. Since we calculate both post-hoc (cumulative) and temporal (time - series) results with regression and classification, our prediction must be able to generate perfect timing for interventions. Given the student details and based on his recent marks in the recent quiz or test, the prediction method various. Though the method may vary, prediction is given for sure.
4) High-Error Data:
This is one of the outputs generated from the feature set. We take some of the most difficult quizzes by computing the error between them. If there is a high error in a particular quiz for all the students, it means that the quiz is difficult for average students. These quizzes can be used as another feature for predicting the course grade and intervention timing. Since we present the student data, the error of all the quizzes in computed individually and those which high error rates are taken as a separate feature set.
5) Analysis of students:
If the operator of the system queries the name of a student to view his performance, this query input will result in a graphical analysis as well as the tabular analysis of that student.

Non- Functional Requirements:

Non-functional requirements describe the user visible aspects of the system that are not directly related with the functional aspects of the system. The non-functional requirements:

1) The information retrieved should be changed periodically depending on the progress of the course.
2) User Interface: Being a prototype model, the interface is just made to be the runtime environment.
3) Exception Handling: An exception is raised when user tires to display information regarding a particular student whose data is not available, and that error or warning is taken care with suitable adjustments.

Tools and Functions used:

GraphLab Create

GraphLab Create™ is a machine learning platform that enables data scientists and app developers to easily create intelligent apps at scale. Building an intelligent, predictive application involves iterating over multiple steps: cleaning the data, developing features, training a model, and creating and maintaining a predictive service. GraphLab **Create™ does all of this in** one platform. It is easy to use, fast, and powerful.

Python

Python is a widely used general-purpose, high level programming language. Its design philosophy emphasizes code readability, and its syntax allows programmers to express concepts in fewer lines of code than would be possible in languages such as C++ or Java. The language provides constructs intended to enable clear programs on both a small and large scale. Since Python can integrate easily with GraphLab Create, we use this language for our coding and implementation.

The IPython Notebook

The IPython Notebook is an interactive computational environment, in which you can combine code execution, rich text, mathematics, plots, and rich media. It aims to be an agile tool for both exploratory computation and data analysis, and provides a platform to support reproducible research, since all inputs and outputs may be stored in a one-to-one way in notebook documents.

5. SYSTEM DESIGN

The Course Rubric

Metric	Description	Number of instances in the course	Total contribution to the final grade
Daily Quizzes	Multiple choice questions tested by day-to-day coursework	22	12%
Case Studies	Case studies that encourage students to apply and study the real-world practical applications of the subject	2	30%
Problem Sets (or) Assignments	Homework problem sets	3	18%
Mid-term tests	Cumulative examinations covering all prior course material	3	40%
Final Grade	Calculated based on the contributions of other metric.	1	N/A

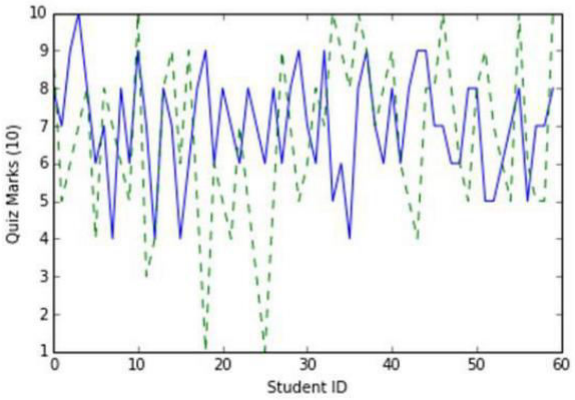
Distribution of Instances over Classes

Class Number	Quizzes	Case Studies	Assignments	Mid-term tests
1	0	0	0	0
2	1	0	0	0
3	2	0	0	0
4	3	0	0	0
5	4	0	0	0
6	5	0	0	0
7	6	0	1	0
8	6	0	1	0
9	7	0	1	0
10	8	0	1	0
11	9	1	1	0
12	10	1	1	0
13	10	1	1	1
14	11	1	1	1
15	12	1	1	1
16	13	1	1	1
17	13	1	2	1
18	13	1	2	1
19	14	1	2	1
20	15	1	2	1
21	15	1	2	2
22	15	1	2	2
23	16	1	2	2
24	17	1	2	2
25	18	1	2	2
26	18	2	2	2
27	18	2	2	2
28	18	2	3	2
29	19	2	3	2
30	20	2	3	2
31	21	2	3	2
32	22	2	3	2
33	22	2	3	3

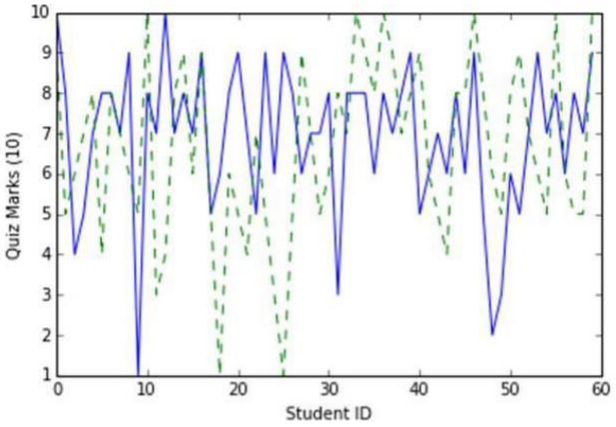
6. GRAPHICAL INTERPRETATIONS

High Error Quiz vs. Normal Quiz

As per the records in the dataset, the calculated Sum Squared Error (SSE) is the highest for Quiz 11 (4.76), and the least for Quiz 20 (2.05).



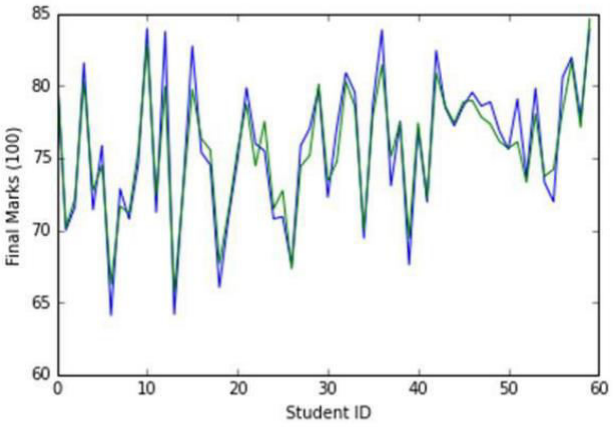
Another example, the difference between Quiz 11 (4.76) and Quiz 5 (3.41)



The blue line shows the line graph for quiz 20/ quiz 5, while the dashed green line shows the variation in quiz 11.

Normal Regression vs. Ridge Regression

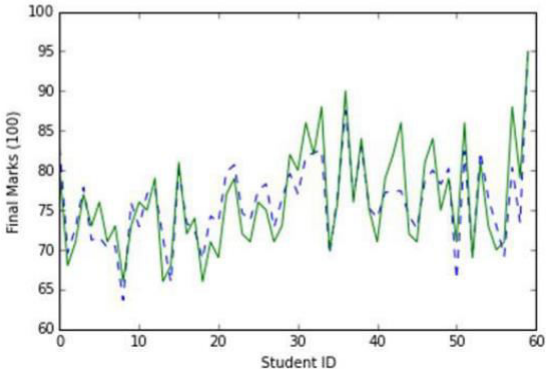
Here, all the quizzes are taken as parameters and two models are constructed. One follows a normal regression, while the other model is developed with a L1 penalty of 0.5e1. The difference is shown in the graph below.



Good Models

Case Studies, Assignments and Mid Exams (Agglomerative) with Ridge

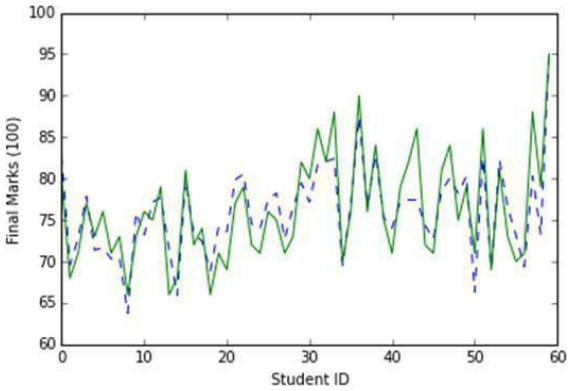
The training error of this model is 0.035 and the test error is The reflection of this model on the original grade is shown in the graph below.



The blue dashed line is the predicted grade from the model, and the green line is the original final grade.

Case Studies, Assignments and Mid Exams (Agglomerative)

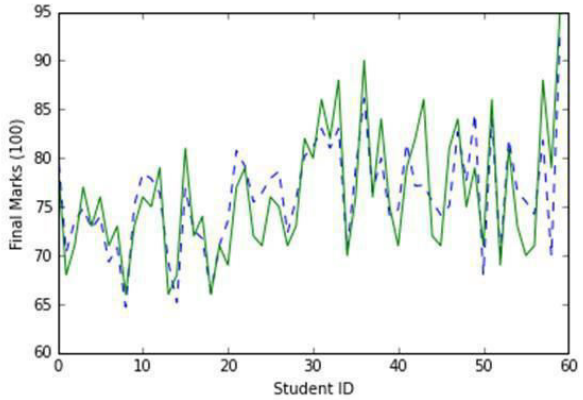
The training error of this model is 0.035 and the test error is The reflection of this model on the original grade is shown in the graph below.



The blue dashed line is the predicted grade from the model, and the green line is the original final grade.

Mid exams and Case Studies (Not Agglomerative)

The training error of this model is 0.038 and the test error is 0.034. The reflection of this model on the original grade is shown in the graph below.

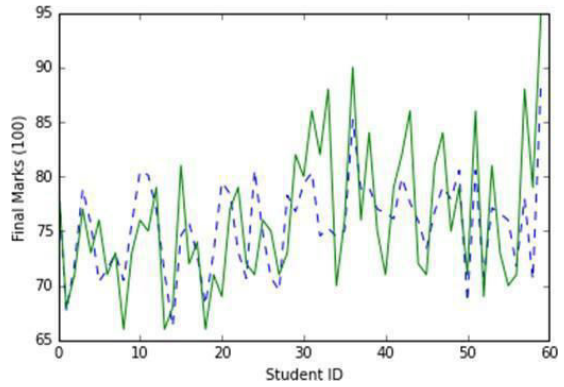


The blue dashed line is the predicted grade from the model, and the green line is the original final grade.

Average Models

Quiz Average and Mid Average

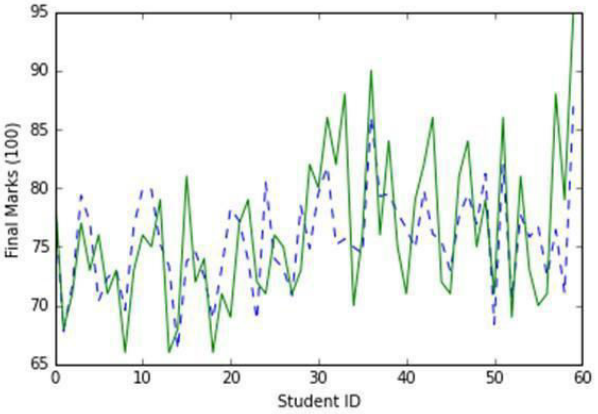
The training error of this model is 0.051 and the test error is 0.049. The reflection of this model on the original grade is shown in the graph below.



The blue dashed line is the predicted grade from the model, and the green line is the original final grade.

High-Error Quizzes Average, Mid Average and the marks of the quiz with the highest Sum Squared Error

The training error of this model is 0.052 and the test error is 0.051. The reflection of this model on the original grade is shown in the graph below.

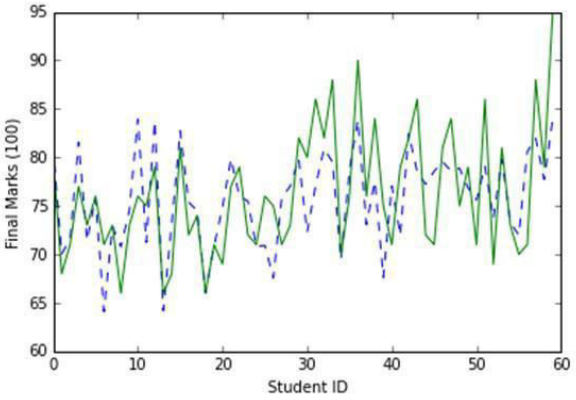


The blue dashed line is the predicted grade from the model, and the green line is the original final grade.

Over-fit Models

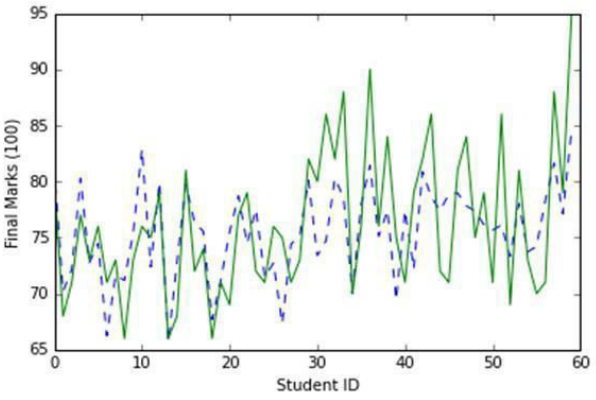
All Quizzes Model (Not Agglomerative) with Ridge

The training error of this model is 0.046 and the test error is 0.065. The reflection of this model on the original grade is shown in the graph below.



All Quizzes Model (Not Agglomerative) without Ridge

The training error of this model is 0.044 and the test error is 0.072. The reflection of this model on the original grade is shown in the graph below.

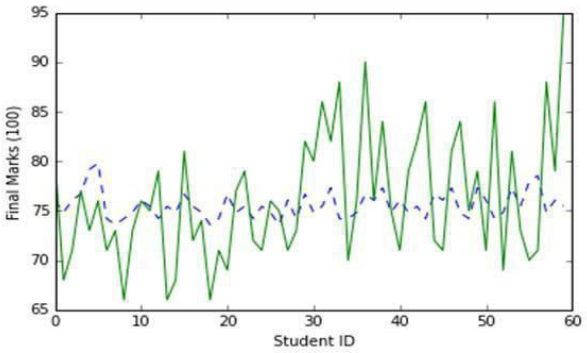


Progressive Learning Models

Temporal Models

Model 1 vs. Final Grade

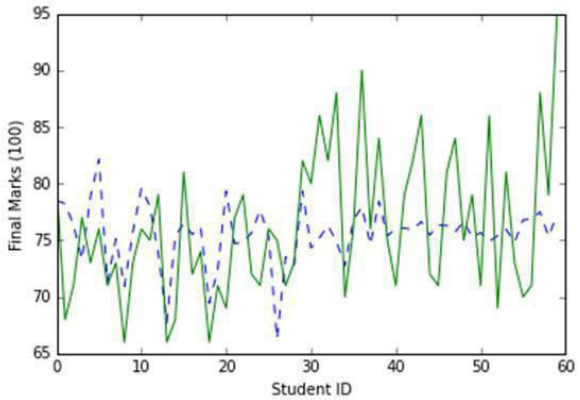
The dashed lines are the predictions for Model 1 (Only the first assessment). The green wiggly line shows the original grade.



The error rate (Residual Sum of Squares) for training data in model 1 is 0.066 (1820.33) and for test data is 0.072 (687.5).

Model 3 vs. Final Grade

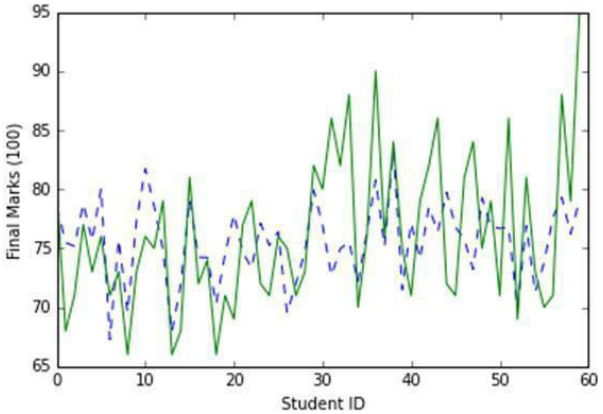
The dashed lines are the predictions for Model 1 (First three assessments). The green wiggly line shows the original grade.



The error rate (Residual Sum of Squares) for training data in model 3 is 0.064 (1594.18) and for test data is 0.073 (678.19).

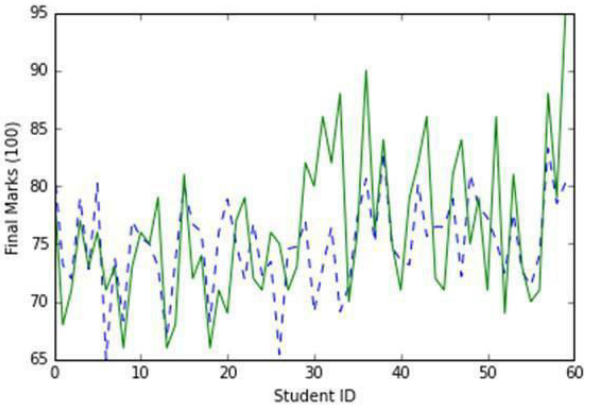
Model 5 vs. Final Grade

The error rate (Residual Sum of Squares) for training data in model 5 is 0.058 (1351.69) and for test data is 0.067 (573.83).



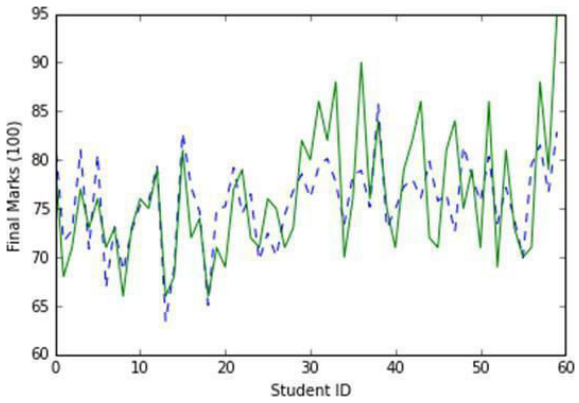
Model 7 vs. Final Grade

The error rate (Residual Sum of Squares) for training data in model 7 is 0.05 (1155.7) and for test data is 0.084 (973.86).



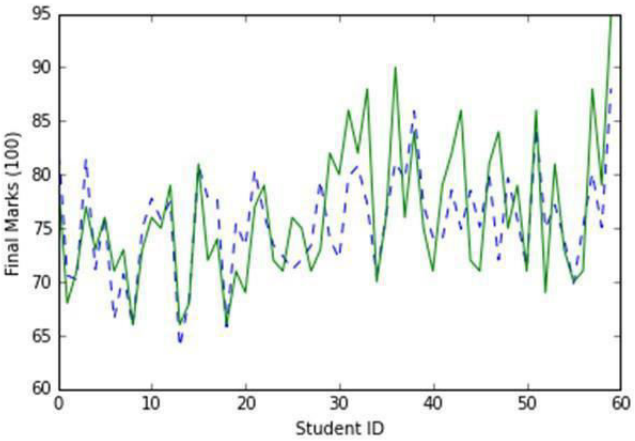
Model 10 vs. Final Grade

The error rate (Residual Sum of Squares) for training data in model 10 is 0.048 (938.02) and for test data is 0.056 (448.09).



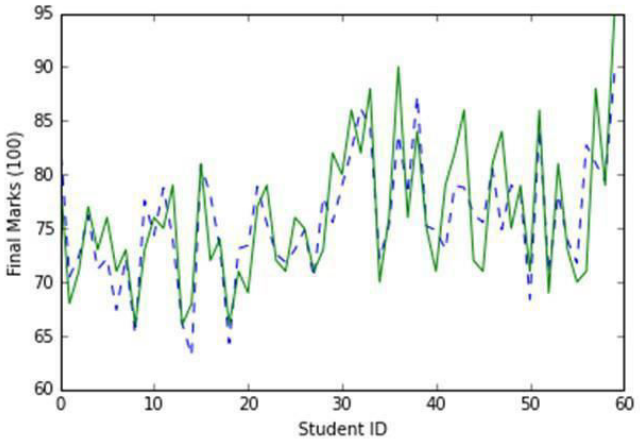
Model 15 vs. Final Grade

The error rate (Residual Sum of Squares) for training data in model 15 is 0.042 (735.88) and for test data is 0.06 (520.98).



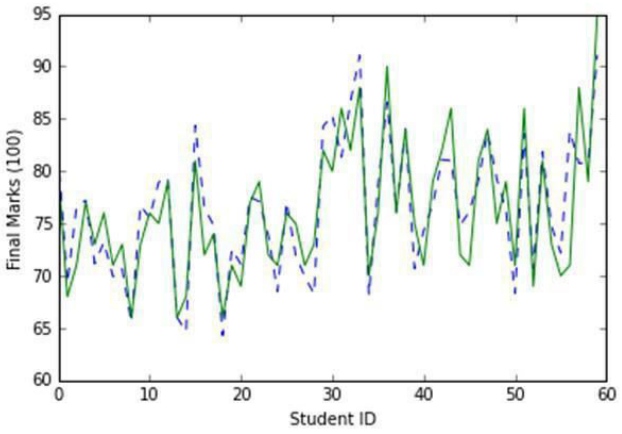
Model 20 vs. Final Grade

The error rate (Residual Sum of Squares) for training data in model 20 is 0.035 (482.39) and for test data is 0.052 (414.11).



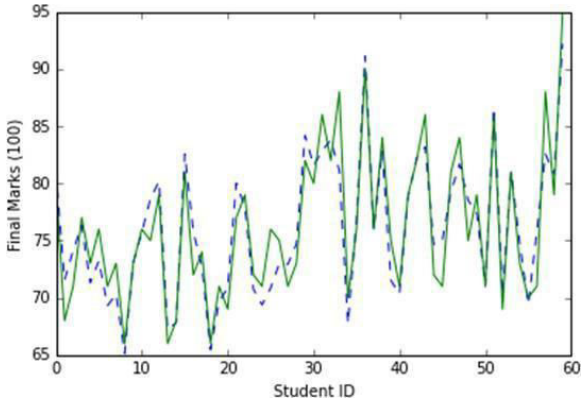
Model 25 vs. Final Grade

The error rate (Residual Sum of Squares) for training data in model 25 is 0.03 (342.57) and for test data is 0.046 (334.37).



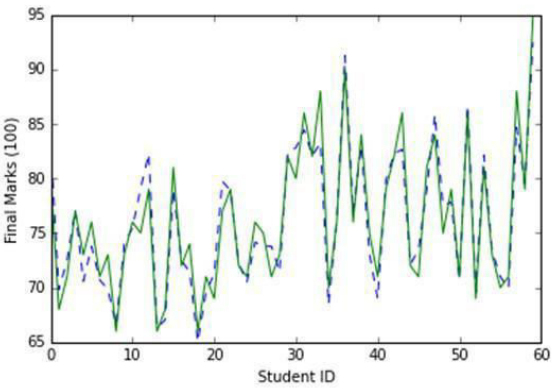
6.6.1.10. Model 28 vs. Final Grade

The error rate (Residual Sum of Squares) for training data in model 28 is 0.023 (206.76) and for test data is 0.031 (143.23).



7.6.1.10. Model 29 vs. Final Grade

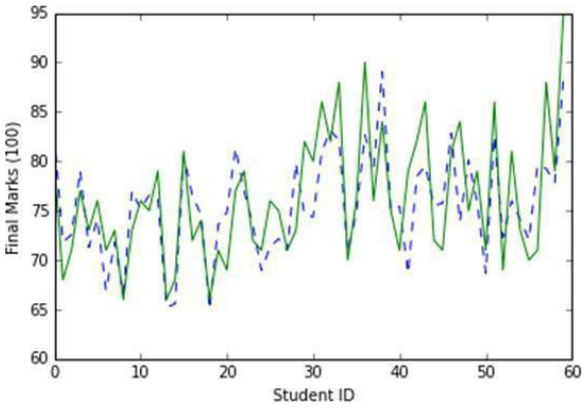
The error rate (Residual Sum of Squares) for training data in model 29 is 0.018 (140.54) and for test data is 0.023 (68.89).



Mean and Median Models

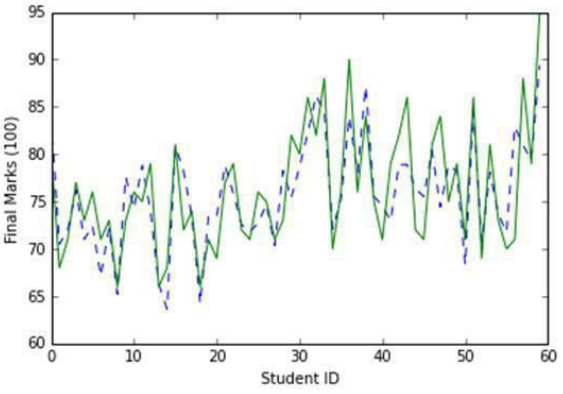
Prediction Model Based on Mean

The error rate (Residual Sum of Squares) for training data in the mean model (Model 18) is 0.039 (578.95) and for test data is 0.065 (528.75).



Prediction Model Based on Median

The error rate (Residual Sum of Squares) for training data in the median model (Model 21) is 0.035 (481.58) and for test data is 0.053 (426.86).



Final Grade Predictions (In decreasing order of error in test data)

Model(Features)	Error Rates(training error, test error)
All Quizzes Model (Not Agglomerative)	(0.044, 0.072)
All Quizzes Model (Not Agglomerative) with Ridge (L1 penalty = 0.5e1)	(0.046, 0.065)
High-Error Quizzes (Agglomerative)	(0.068, 0.062)
All Quizzes (Agglomerative)	(0.066, 0.061)
High-Error Quizzes Average, Mid Average and the marks of the quiz with the highest Sum Squared Error	(0.052, 0.051)
Quiz Average and Mid Average	(0.051, 0.049)
Mid exams and Case Studies (Not Agglomerative)	(0.038, 0.034)
Excluding Quizzes (Agglomerative)	(0.034, 0.033)
Case Study Average, Assignment Average, and Mid Average	(0.035, 0.032)
Case Study Average, Assignment Average, and Mid Average with Ridge (L1 penalty = 0.01e1)	(0.035, 0.031)

Final Grade Predictions(In order of method complexity)

Model (Features)	Error Rate (Training Error, Test Error)
Simple Linear Regression	
All Quizzes (Agglomerative)	(0.066, 0.061)
Excluding Quizzes (Agglomerative)	(0.034, 0.033)
High-Error Quizzes (Agglomerative)	(0.068, 0.062)
Multiple Linear Regression	
Quiz Average and Mid Average	(0.051, 0.049)
High-Error Quizzes Average, Mid Average and the marks of the quiz with the highest Sum Squared Error	(0.052, 0.051)
Case Study Average, Assignment Average, and Mid Average	(0.035, 0.032)
All Quizzes Model (Not Agglomerative)	(0.044, 0.072)
Mid exams and Case Studies (Not Agglomerative)	(0.038, 0.034)
Ridge Regression	
All Quizzes Model (Not Agglomerative) with Ridge (L1 penalty = 0.5e1)	(0.046, 0.065)
Case Study Average, Assignment Average, and Mid Average with Ridge (L1 penalty = 0.01e1)	(0.035, 0.031)

7. RESULTS

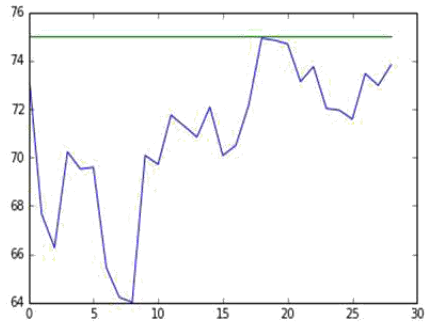
Interpreting the Complete Report

- The first line displays the original grade of the student along with his ID.
- Then, temporal projections calculated from Quiz 1 (Model 1) to Mid 3 (Model 29) are displayed chronologically in a tabular form. Next, details about interventions are displayed, if any.
- We only chose to reveal post-hoc intervention assessments rather than temporal intervention predictions.
- Now the summary according to temporal models.
  - The best estimated/projected value
  - The assessment after which the value was best estimated.
  - Difference between original value and the projected value
  - Estimation based on mean model
  - Estimation based on median model
- We tested some post-hoc models, and the results according to those are displayed here. Quiz average and Mid Average, All Quizzes, to name a few.
- The graphical representation of projections across various models to that of the final grade.
  - Thecurved line shows the mapping/projections of marks at regular instances of evaluation.
  - The straight line describes the final grade.
  - The place these two first meet is the point at which the error is minimum.

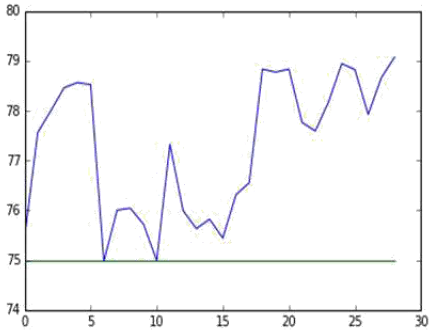
Total Report Sample Output 1

```
Enter Student ID: 27
Final Grade Achieved (Original) :75

Temporal Predictions

Assessment   Marks
    1        73.58
    2        67.67
    3        66.27
    4        70.23
    5        69.52
    6        69.59
    7        65.44
    8        64.22
    9        64.01
    10        70.08
    11        69.71
    12        71.75
    13        71.3
    14        70.84
    15        72.08
    16        70.08
    17        70.49
    18        72.16
    19        74.92
    20        74.84
    21        74.69
    22        73.13
    23        73.75
    24        72.02
    25        71.95
    26        71.58
    27        73.46
    28        72.97
    29        73.83
```

```
Interventions (if any)

Intervention Warning 1 quiz 3
Intervention Warning 2 quiz 4
Intervention Warning 3 quiz 13
Intervention scheduled after quiz 14 and after class number 17
Intervention scheduled after quiz 15 and after class number 18
Intervention scheduled after quiz 16 and after class number 21

Models Derived From Temporal Predictions

Best Estimated Value: 74.92
Best Estimated Value Was Addressed After Assessment Number: 19
Difference Between Original and Best Predicted Value: 0.08
Estimated Final Marks Based on Mean Assessment Indexes: 72.16
Estimated Final Marks Based on Median Assessment Indexes: 74.69


Post Hoc Predictions

Quiz Average and Mid Average: 70.74
High Error Quiz Average, Mid Average, and Quiz With Highest Error Rate: 73.15
Case Study Average, Mid Average, Assignment Average: 78.24
Case Study Average, Mid Average, Assignment Average (Adjusted): 78.3
All Quizzes: 67.57
All Quizzes (Adjusted): 67.33
Mid Exams and Case Studies: 78.64

     Graphical Representation of Data

     Blue Line Shows The Variability Across Temporal Predictions
     Green Line Shows the Final (Original) Grade
```




```
Interventions (if any)

Intervention Warning 1 quiz 12
Intervention Warning 2 quiz 13
No Intervention Required

Models Derived From Temporal Predictions

Best Estimated Value: 74.99
Best Estimated Value Was Addressed After Assessment Number: 11
Difference Between Original and Best Predicted Value: 0.01
Estimated Final Marks Based on Mean Assessment Indexes: 76.55
Estimated Final Marks Based on Median Assessment Indexes: 78.83


Post Hoc Predictions

Quiz Average and Mid Average: 80.13
High Error Quiz Average, Mid Average, and Quiz With Highest Error Rate: 79.88
Case Study Average, Mid Average, Assignment Average: 77.15
Case Study Average, Mid Average, Assignment Average (Adjusted): 77.02
All Quizzes: 71.24
All Quizzes (Adjusted): 72.38
Mid Exams and Case Studies: 77.86

 Graphical Representation of Data

 Blue Line Shows The Variability Across Temporal Predictions
 Green Line Shows the Final (Original) Grade
```



Total Report Sample Output 2

```
Enter Student ID: 12
Final Grade Achieved (Original) :75

Temporal Predictions

Assessment   Marks
    1        75.45
    2        77.56
    3        78.0
 e  4        78.46
    5        78.56
    6        78.52
    7        74.97
 t  8        76.0
    9        76.04
    10        75.72
    11        74.99
    12        77.32
    13        75.99
    14        75.63
    15        75.82
    16        75.44
    17        76.31
    18        76.55
    19        78.83
    20        78.77
    21        78.83
    22        77.76
    23        77.59
    24        78.17
    25        78.94
    26        78.82
    27        77.92
    28        78.66
    29        79.07
```

9. CONCLUSION

Through this project, we were able to su
implement some of the underdeveloped metho
field of machine learning such as progressive lea
we were able to accurately classify intervention
the process-level data. The course rubric consider
project, along with accurate temporal predict
helped in creating some post-hoc models tha
improved through large scale execution of this
Though the structure of the course may be rathe
on the student as well as the teacher, the outpu
and the learning models help in the developme
organizing interventions, if needed, at perfect ir
time. One of the major problems of implementing
is the scale of the data set. This rubric, being exe
rather small data set of just a one course and a
students, the results may seem to differ. But whe
a large scale, say on a single course across 30
students, the models tend to smoothen with tha
As said earlier, with the increase in data, we
improve the post-hoc models, and we can also de
models by creating new feature sets. With the
progress in the field of machine learning, the da
before we can create machines that can act as tea

We would like to leave with this work, and future amends can be made to this project by using complex methods such as Artificial Neural Networks (ANN). It is unnecessary for a small data set of 60 students to process through a refined neural network and would cause very critical errors. Therefore, given our scale and the environment, we were able to develop and create an environment, though labor-intensive, both the faculty and students can know the accurate position of the student at a given time in the semester.

REFERENCES

1. Alexander J Stimpson and Mary L. Cummings. **"**Assessing Intervention TiminginComputer-BasedEducationUsing Machine Learning Algorithms**"**, IEEE Access, Volume 2, 27 February 2016.
2. Carlos Guestrin et al. **"**GraphLab Create – Sophisticated Machine Learning as Easy as **'**Hello World**'"**, Dato.com, 27 February 2016.
3. Grady Booch. **"**Principles of Design**"**, the Unified Modeling Language User Guide, 27 February 2016.
4. Hunter J. D. **"**Mathplotlib: A 2D Graphics Environment**"**, Computing in Science and Engineering, IEEE Computer Society, Volume 9, 27 February 2016.
5. Trevor Hastie et al. **"**Supervised Learning**"** The Elements of Statistical Learning (Data mining, Inference and Prediction), 27 February 2016.