Modelling Student Performance in Large Online Classes.

Joseph Okonda L.

Problem

Personalized learning systems promise to significantly transform the education industry by tailoring content to learners' needs and offering by offering `Just in Time` assistance. Such systems need a way of 'knowing' when a given student needs assistance before they can intervene by, for instance, directing the student to relevant study material.

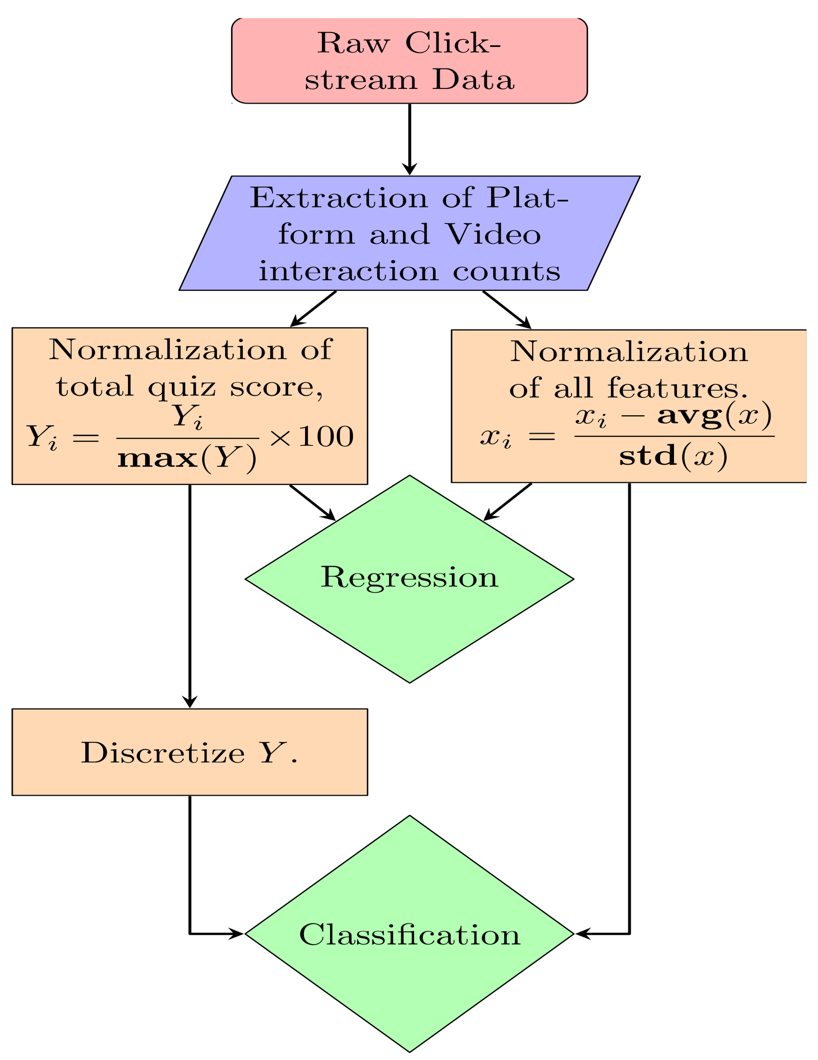
Dataset

The Project used clickstream data collected from 4 MOOCs offered through Stanford's OpenEdx Learning Platform. The data catalogues each student's interaction with the learning platform. For training, we only consider students who attempted at least one quiz.

Course	Total Enrollment	Training Data
MMDS	13 818	430
Algorithms	18 358	333
Compilers	30 485	2200
Stat. Learning	125 105	39616

Feature Extraction

From the raw data, we extract the following features: Video interaction counts (number of plays, pauses, seeks, stops, speed change etc.), Platform interaction counts(forum, transcript, download).



Models

- **L_2 Regularized Linear Regression.**

We first fit Ridge regression to the data. We use Mean Square Error as the loss function. We also use RMSE to measure performance on a hold-out cross-validation set.

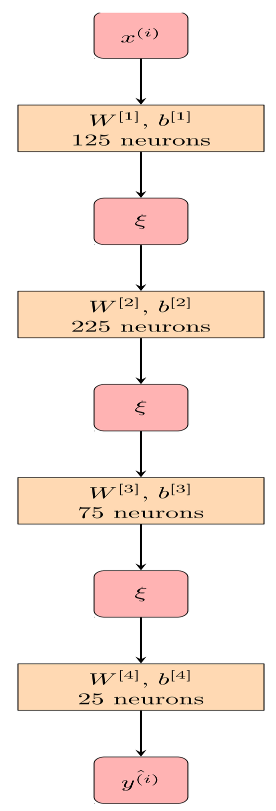
$$\ell(\theta) = \frac{1}{m}\sum_{i=1}^{m}(\theta^T x^{(i)} - y^{(i)})^2 + \frac{1}{\lambda} \parallel \theta \parallel^2$$

- **K-Nearest Neighbors Regression.**

K-NN is a non-parametric model that makes predictions by taking a weighted average of the K data-points closest to the point we want to estimate. This project uses K=10 and Euclidean Distance as the distance measure. As with Ridge Regression, we use RMSE to measure performance.

- **Neural Network Regression.**

We fit a fully connected 4-Layer Feed Forward NN with ReLU nonlinearities.



- **Multinomial Logistic Regression.**

We turn the problem into a classification task by discretizing the response variable into quartiles.

On the discretized data, we first fit softmax regression with K = 4 classes i.e. the 4 quartiles between 1 and 100. We evaluate the model by measuring its F1 score on a hold out test set.

- **Support Vector Classification.**

We also fit (on the discretized data sets) a support vector classifier using the Radial Basis Kernel.

$$K(X, X') = \exp\left(-\frac{||x - x'||2}{2\sigma^2}\right)$$

Results & Analysis

- **Regression**

First, we evaluate our models on the 4 individual courses.

	MMDS	Compilers	Algorithms	Stats.	All.
KNN	23.50	20.02	20.03	13.22	13.76
Ridge Reg.	23.60	19.50	**19.33**	13.77	**13.44**
NN Reg.	**23.42**	**19.32**	19.54	**12.79**	13.45

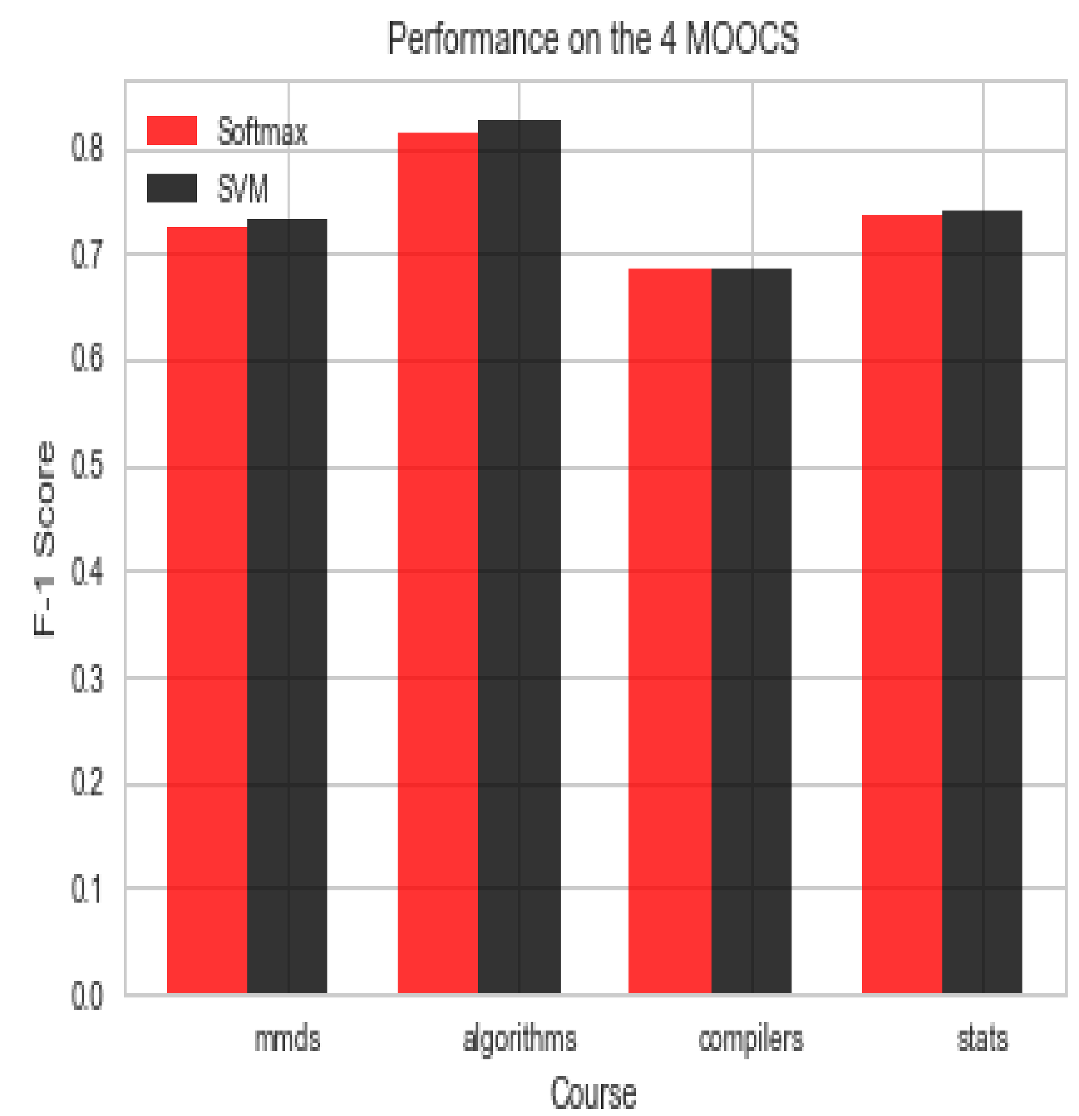
The table above shows the RMS errors for the three regression models on the 4 MOOCs. In addition, the last column shows the performance of the models on the combined dataset. On average, of the three models, the NN has the least RMS error.

Next, we focus on the NN and measure how well it generalizes to `unrelated` courses. Specifically, we train the model on course A and test it on course B where $A \neq B$. The columns of the table below show the data that the model was trained on and the rows indicate the data on which it was tested. The values are the RMSEs

	MMDS	Compilers	Algorithms	Stats.
MMDS	23.42	24.66	24.92	24.97
Compilers	17.74	19.32	17.34	17.36
Algorithms	17.74	16.89	19.54	16.82
Stats.	14.28	13.12	13.01	12.79

- **Classification.**

The chart below displays the performance of the classification models on the discretized data. We use F-1 as the performance metric because the discretized data, for all the courses, is highly unbalanced, with a majority of the labels being for the 1st quartile (0-25%)



Future Works

This project demonstrated the ability to use a learners' click behavior to predict their performance in an online class as measured by the number of points they accrue from the course's quizzes. This was done in the context of creating a system that can automatically flag when a student needs assistance. However, the project does not tell us in what areas the flagged student needs assistance in. Additionally, instead of using the features used in the project to predict a student's score at the end of the course, one could include a temporal component in the analysis and fit models such as Hidden Markov Models.

References

1. Pedregosa, F., Varoquaux, G., Gramfort, A. & Michel, V. (2011). Scikit-learn: Machine Learning in Python. *Journal of Machine Learning Research, 12*, 2825-2830.
2. Knowles, Jared (2014). Of Needles and Haystacks: Building an Accurate Statewide Dropout Early Warning System in Wisconsin. *JEDM - Journal of Educational Data Mining, 7*, 1-52.
3. Geigle, Chase & Zhai, ChengXiang (2017). Journal of Educational Data Mining JEDM.. *JEDM - Journal of Educational Data Mining, 9*, 1-24.
4. Casey, Kevin (2017). Using keystroke analytics to improve pass-fail classifiers. *Journal of Learning Analytics, 4*, 189-211.