Learning Analytics Results

Tasha Vincent

March 6, 2019

Introduction

This project leverages the data gathered from engineering students at the University of Genoa in Italy, and examines whether general engagement metrics such as number of activities completed, mouse movements, and keystrokes can be used to accurately predict outcomes on the final exam. I extract and transform the data using data wrangling methodologies, explore data visualization in R Studio, and develop a machine learning algorithm evaluated using RMSE. Although the results are not as dramatic as hoped, the data is fruitful for further exploration and the data is now in excellent shape to explore other metrics that may be more predictive of final results.

Methods and Analysis

I began by downloading the Educational Process Mining Learning Analytics Dataset from here https://archive.ics.uci.edu/ml/datasets/Educational+Process+Mining+(EPM)%3A+A+Learning+Analytics+Data+Set including data gathered from the engineering students at the University of Genoa in Italy. The datasets were originally organized by session, with separate files for intermediate grades and final grades.

Data wrangling

Step one was to consolidate all of the session level information into a single, tidy data set that can be analyzed by student and by session to build a picture of a given student's experience in the classroom. For each of the six sessions, data was collected in a single file per student ID containing a list of the activities completed, and the mouse movements, keystrokes, and start and end times for each activity. I used the datatable library to extract a list of the files into a consolidated dataframe per session, then used the bindrows function to generate a single data set across all sessions and students. The column heads for the session information were provided separately, so I used the colnames function to bind the column names to the data.

Step two was to consolidate the grades data. The intermediate grades file included scores for activities that students completed in sessions 2 through 6. Although the information in this file was readily scannable, it was not in a tidy format, so I used gather to create a new dataframe in a tidy format.

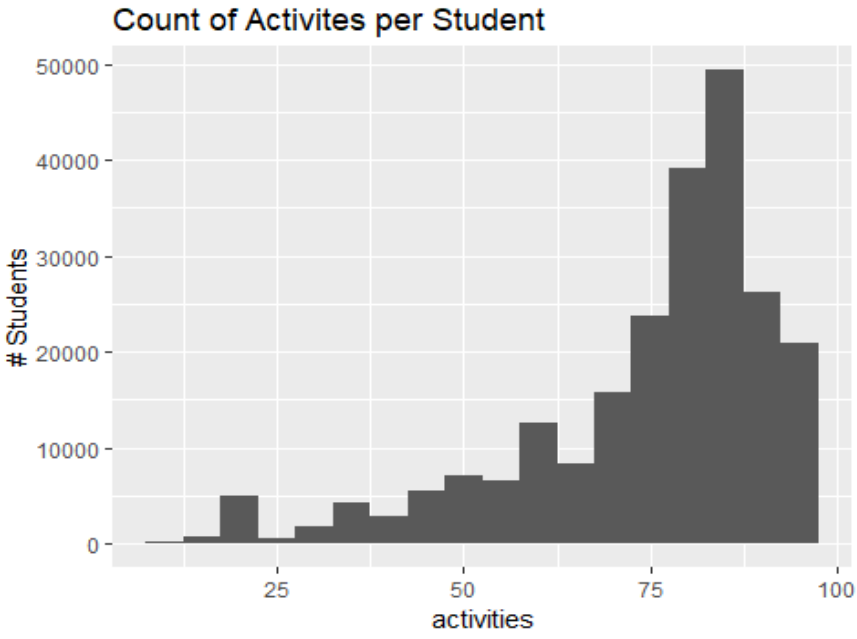
```
##    Student Id      Session 2       Session 3       Session 4
##  Min.   :  1.0   Min.   :0.000   Min.   :0.000   Min.   :0.000
##  1st Qu.: 29.5   1st Qu.:0.000   1st Qu.:0.500   1st Qu.:4.000
##  Median : 58.0   Median :3.500   Median :2.500   Median :4.500
##  Mean   : 58.0   Mean   :2.887   Mean   :2.135   Mean   :3.943
##  3rd Qu.: 86.5   3rd Qu.:4.500   3rd Qu.:3.500   3rd Qu.:5.000
##  Max.   :115.0   Max.   :6.000   Max.   :4.000   Max.   :5.000
##    Session 5      Session 6
##  Min.   :0.00   Min.   :0.000
##  1st Qu.:3.00   1st Qu.:0.125
##  Median :3.50   Median :2.000
##  Mean   :3.03   Mean   :1.696
##  3rd Qu.:4.00   3rd Qu.:2.750
##  Max.   :4.00   Max.   :4.000

## # A tibble: 6 x 3
##   StudentId Session  Score
##       <dbl> <fct>    <dbl>
## 1         1 Session2   5
## 2         2 Session2   4
## 3         3 Session2   3.5
## 4         4 Session2   6
## 5         5 Session2   5
## 6         6 Session2   5.5
```

The final exam grades were provided in two files, with students allowed to sit one or both sessions. I opted to use the students latest score as their final score in this case, and consolidated the files through join statements into a final dataframe.

The final exam data included the scores on session-specifc questions. To prepare this data for further analysis, I created an items dataset in tidy format.

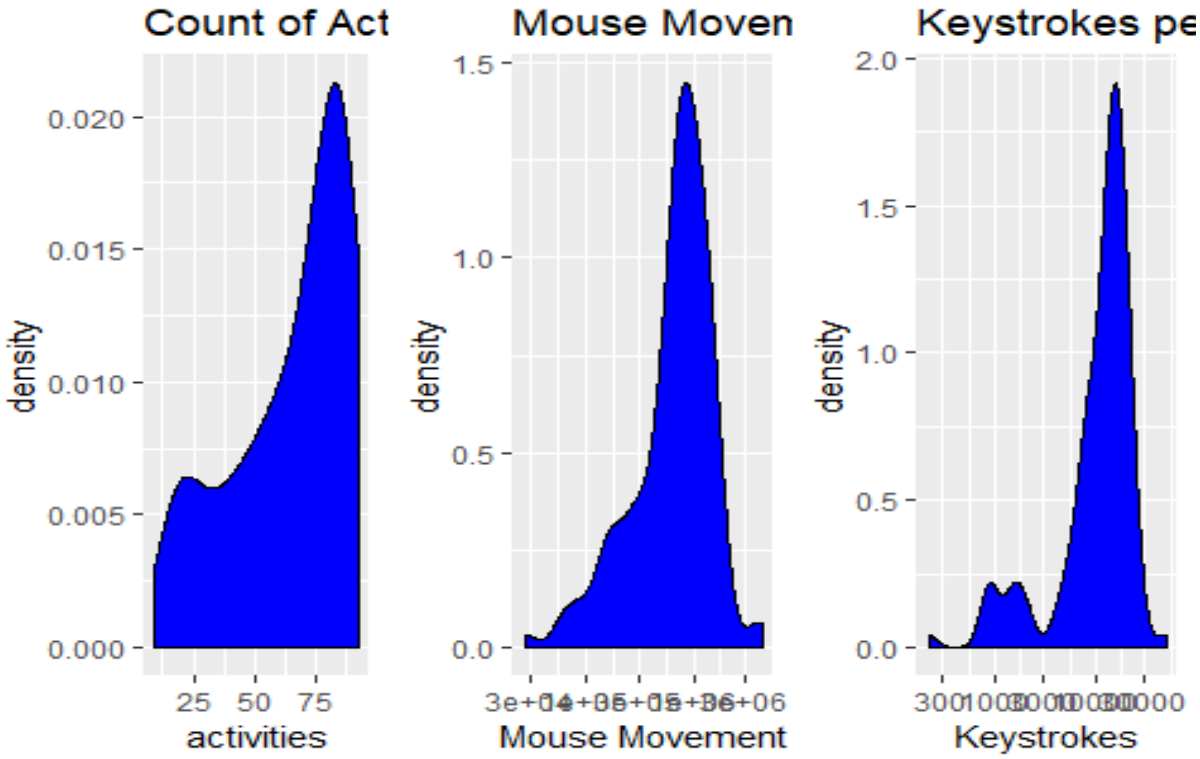
Data Visualizations

I began by creating a dataframe with just the metrics I wanted to explore, grouping the data by student ID. I then created a histogram to explore whether students usually completed the same number of activities.



Next, I created density plots with counts of activities for student, mouse movements

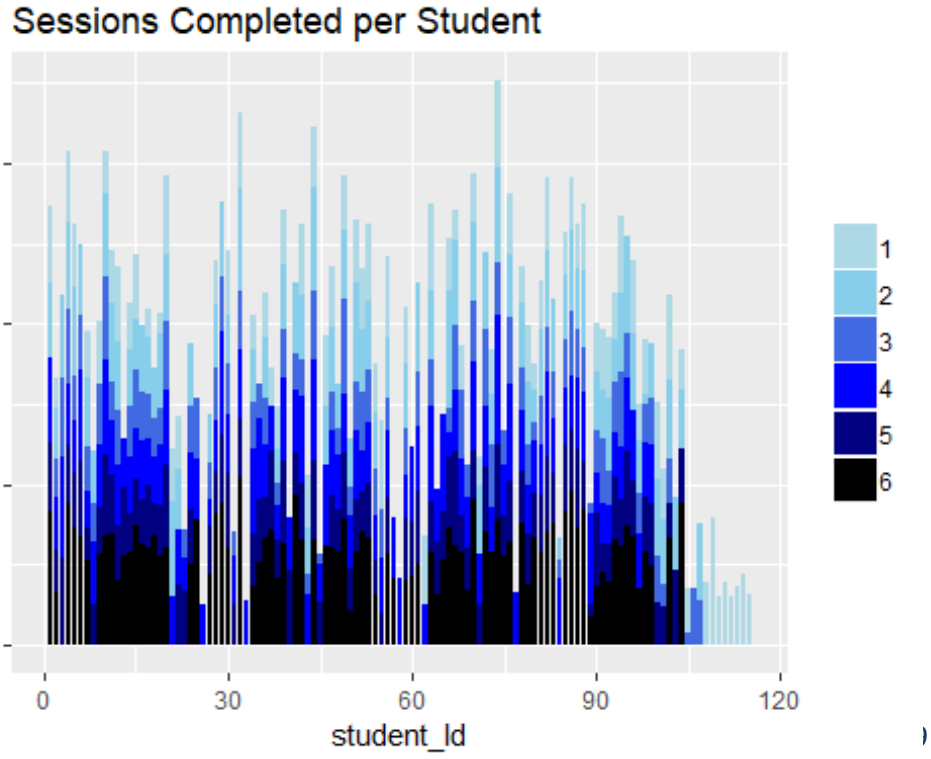
per student, and keystrokes per student. The number of activities shows a bimodal distribution, with peaks around 20 and 40 activities while the number of mouse movements and keystrokes are shown in gaussian distribution.



Next I began to explore the session-level information. Next I grouped by session as well as by student.

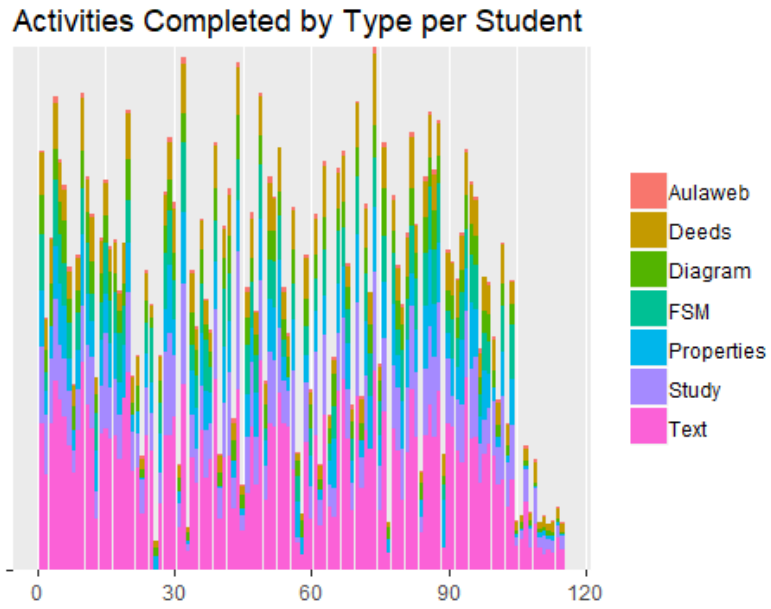
We can see that different students completed different sessions.

Examining the histograms at this level shows a

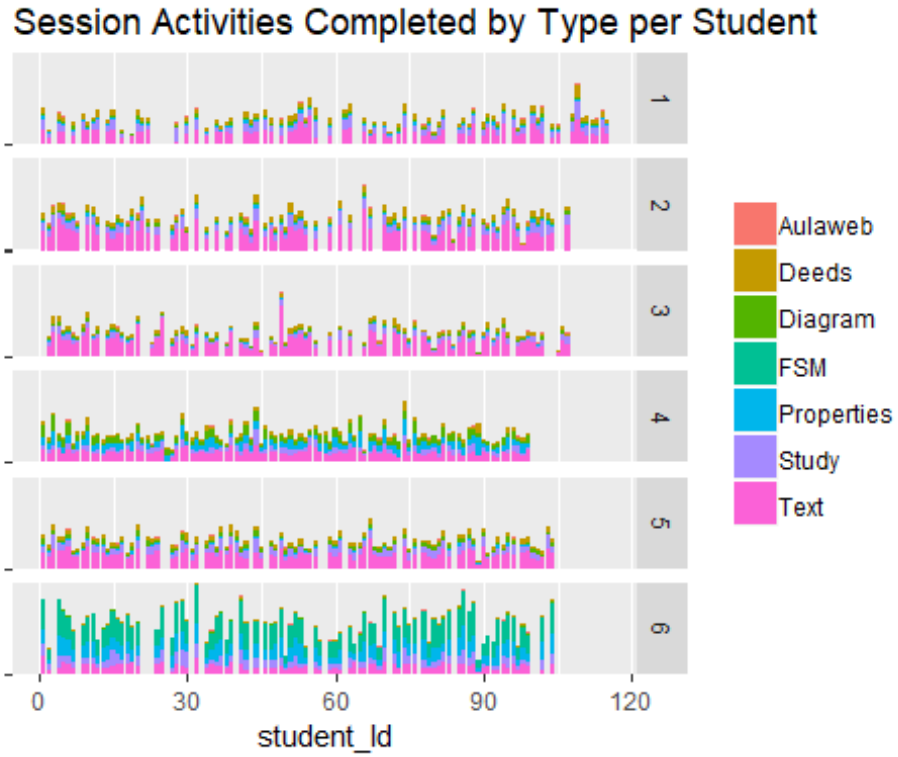


greater variety in student engagement. I also explored using box plots to summarize student mouse movements and keystrokes per session.

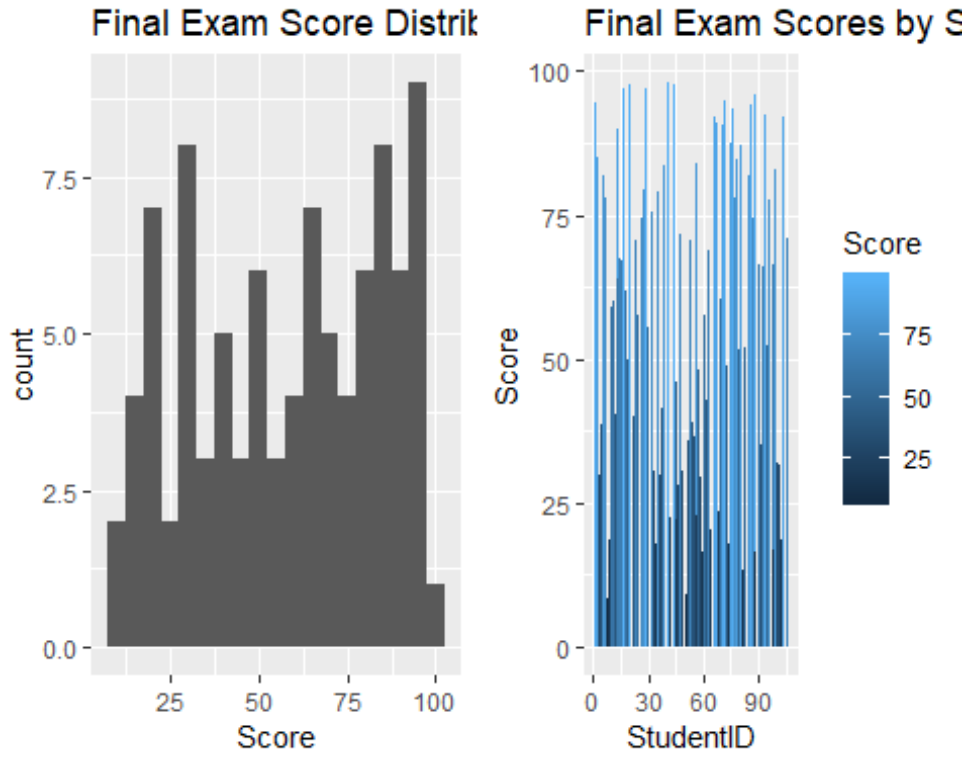
I examined the activities that users completed, by activity. The visualization generated was difficult to interpret however, so I further manipulated the data to consolidate activity types.



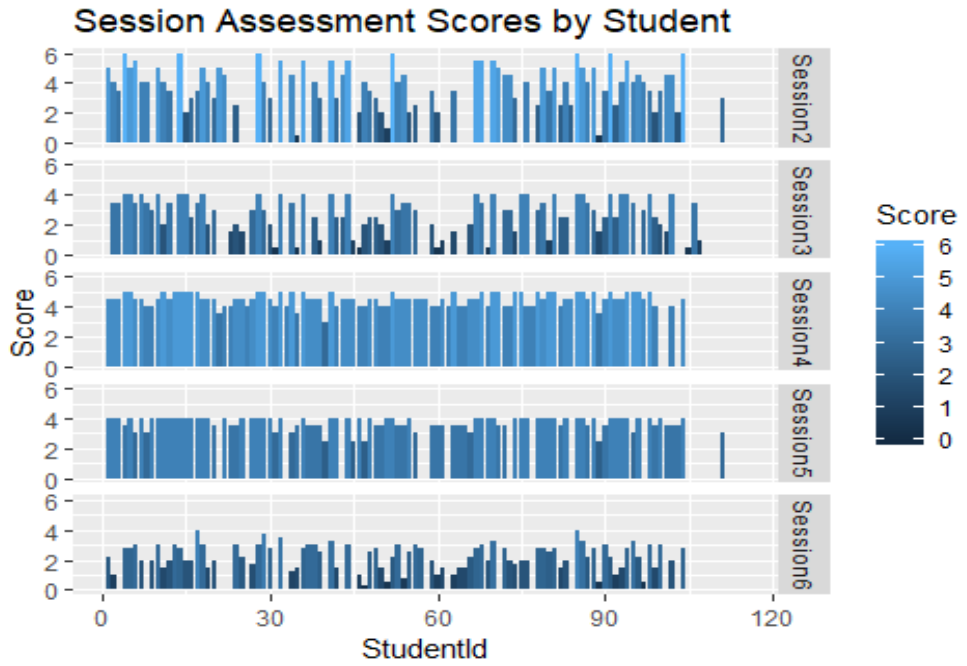
I then further explored this information by session.



Finally, I turned my attention to the scores. First I looked at the final exam scores, and noted a wide variability in outcomes.



Next I took a look at the session level activity scores, and observed wide variations in participation by students, as well as point values earned per student, per activity, and for the same student across activities.



Creating a predictive model

I began by combining all of the score values into a single data set. I used the caret package to partition the data set, using 30% of the data to test my model. I chose 30% because the relatively small number of students in the data set, 115, and felt that 10% would be too low, and 50% too high. To create test and training sets, I first needed to isolate consistent data. Not all students completed all six sessions, and not all the students who completed work in a session also took the final exam. There were only 93 students who took the final exam, so I used this as the basis for creating and index of 30% of the students. I then created a test set of exam scores for these students, and activity summary data for the students. I created training sets for the activities completed and exam scores, ensuring that the same student IDs existed in both, then joined the tables.

I opted to use the residual mean squares estimate as the definition of success for this model. I started by creating a model that simply predicted that the students would receive the average score on the final exam. I then evaluated this model using the RMSE function.

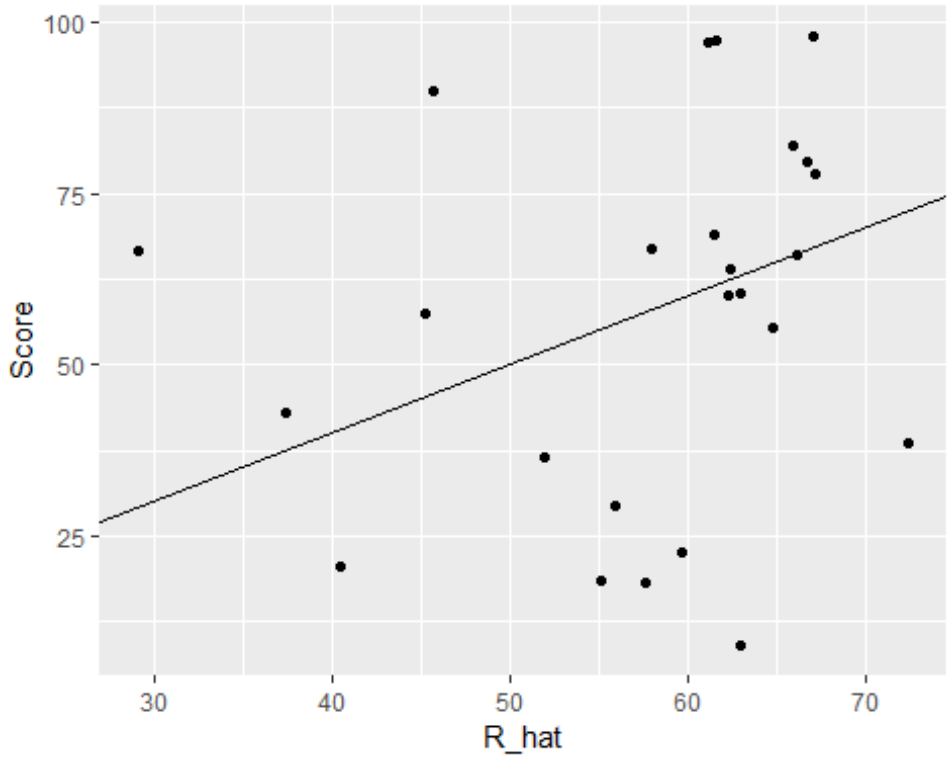
```
mu <- mean(train$Score)

avg_only <- RMSE(test$Score, mu)
avg_only
```

```
## [1] 26.63665
```

Next I used lm to see whether the number of activities, number of mouse movements, and number of keystrokes were correlated to the final exam scores for a given student. I created a simple plot of the results vs. the regression line, which implied that the correlation was weak.

```
##
## Call:
## lm(formula = Score ~ activities + mmove + keys, data = train)
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -50.705 -18.069   1.043  20.521  41.800
##
## Coefficients:
##              Estimate Std. Error t value Pr(>|t|)
## (Intercept) 2.365e+01  1.143e+01   2.069   0.0426 *
## activities  2.256e-01  2.136e-01   1.056   0.2949
## mmove       5.435e-06  5.043e-06   1.078   0.2852
## keys        9.876e-04  5.740e-04   1.721   0.0901 .
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 24.64 on 64 degrees of freedom
## Multiple R-squared:  0.2153, Adjusted R-squared:  0.1785
## F-statistic: 5.853 on 3 and 64 DF,  p-value: 0.001347
```

```
## # A tibble: 4 x 7
##   term          estimate  std.error statistic p.value   conf.low conf.high
##   <chr>            <dbl>      <dbl>     <dbl>   <dbl>      <dbl>     <dbl>
## 1 (Intercep~ 23.7           1.14e+1      2.07  0.0426    8.16e-1   4.65e+1
## 2 activities  0.226         2.14e-1      1.06  0.295    -2.01e-1   6.52e-1
## 3 mmove       0.00000543    5.04e-6      1.08  0.285    -4.64e-6   1.55e-5
## 4 keys        0.000988      5.74e-4      1.72  0.0901   -1.59e-4   2.13e-3
```



Nonetheless, I wanted to see if these measures could be used to improve the average only model, so I used the coefficients to create a model of effects of activity, mouse movements, and the number of keystrokes.

```
coefs <- tidy(fit, conf.int = TRUE)
coefs

## # A tibble: 4 x 7
##   term          estimate  std.error statistic p.value   conf.low conf.high
##   <chr>            <dbl>      <dbl>     <dbl>   <dbl>      <dbl>     <dbl>
## 1 (Intercep~ 23.7           1.14e+1      2.07  0.0426    8.16e-1   4.65e+1
## 2 activities  0.226         2.14e-1      1.06  0.295    -2.01e-1   6.52e-1
## 3 mmove       0.00000543    5.04e-6      1.08  0.285    -4.64e-6   1.55e-5
## 4 keys        0.000988      5.74e-4      1.72  0.0901   -1.59e-4   2.13e-3

predicted_score <- 23.7 + 0.226*test$activities +  0.00000543*test$mmove +
0.000988*test$keys
```

Results

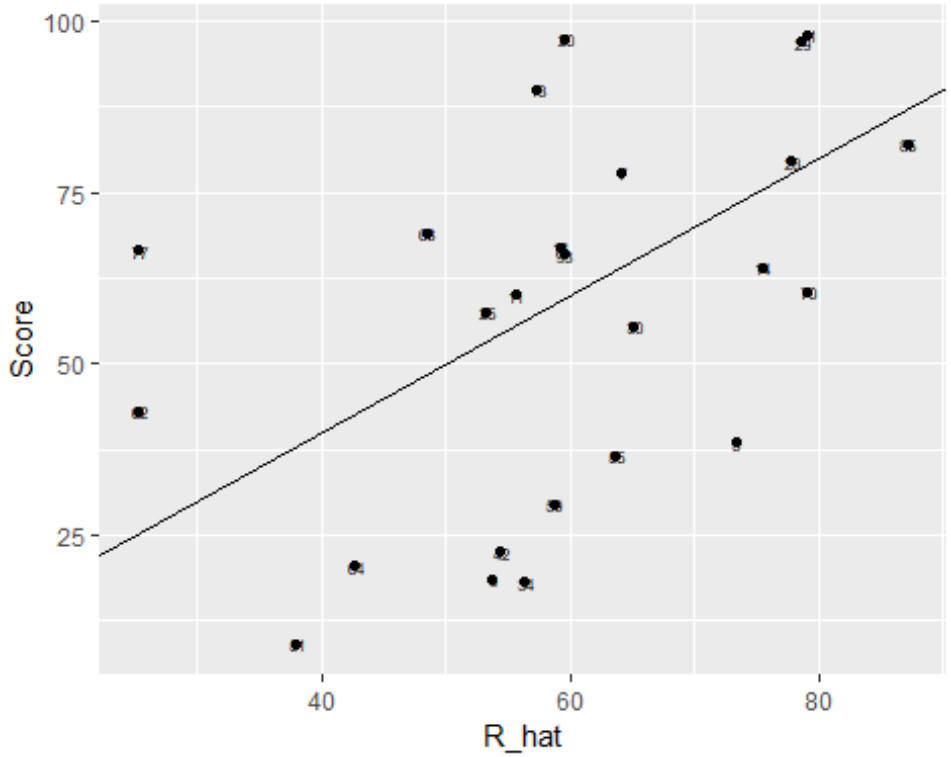
The results of the linear model based on engagement metrics are not very compelling. The average only approach has a rmse of over 26, which means it can only predict a student score within 26 points on a 100-point exam. Additional terms comparing number of activities, mouse movements, and keystrokes and their effect on the final score barely improved the model at all, with a combined effective less than 1 percentage point.

Model	RMSE
Simple average	26.63665
Linear model of engagement metrics	26.50072

Next, I created a model based on the scores on session activities.

```
## # A tibble: 6 x 7
##   term        estimate std.error statistic p.value conf.low conf.high
##   <chr>          <dbl>     <dbl>     <dbl>   <dbl>    <dbl>     <dbl>
## 1 (Intercept)  24.9        13.4     1.85   0.0691     -2.00     51.7
## 2 Session2      2.10        1.89    1.11   0.269      -1.67      5.88
## 3 Session3      1.96        2.37    0.827  0.411      -2.78      6.71
## 4 Session4      0.0884      2.84    0.0311 0.975      -5.59      5.77
## 5 Session5      0.980       3.22    0.304  0.762      -5.45      7.41
## 6 Session6      9.36        3.23    2.90   0.00523     2.90     15.8
```

I then plotted the results with a regression line.



These results weren't much better, improving RMSE by just over 2 percentage points.

Model	RMSE
Simple average	26.63665
Linear model of engagement metrics	26.50072
Linear Model of Session Scores	24.02432

Conclusion and next steps

While the outcome was not as dramatic as I had hoped, the methodology does seem promising, perhaps if applied to session-level variability. For next steps, I would explore whether completing certain activity types have a stronger effect on the final exam score, and perhaps even activity types by session. Finally, I would look to see whether the session level scores correlate to the questions on the final exam that come from each session. In summary, I think the data is ripe for generating a predictive model, provided I can find the right metrics to include in the model.

The data on which this project is based was originally published in the following publication.

[1] M. Vahdat, L. Oneto, D. Anguita, M. Funk, M. Rauterberg.: A learning analytics approach to correlate the academic achievements of students with interaction data from an educational simulator. In: G. Conole et al. (eds.): EC-TEL 2015, LNCS 9307, pp. 352-366. Springer (2015). DOI: 10.1007/978-3-319-24258-3 26