The University of Texas at Dallas

EPPS 6354 Information Management

Professor: Karl Ho

Student: Federico Ferrero

Final Paper

Academic Analytics to report student performances and prevent dropout at university courses

Academic Analytics to report student performances and prevent dropout at university courses

Introduction

In the current scenario of data proliferation in virtual educational environments -or what some authors have called “datafication” in education (Breiter, 2016; Selwyn, 2015; Van Dijck, 2014)- new descriptive and evaluative developments occur not only at systemic level but also at the level of the individual learning. In this context, the Learning Analytics (focused on individual learning) and the Academic Analytics (concentrated in the management of academic institutions) allow professors and administrators to use educational data in order to better understand their student’s learning processes and, thus, customize pedagogical interventions.

Just to mention an application example, the design and use of algorithms dedicated to the “prediction of students success” are increasingly frequent. Particularly, these developments assume as a starting point that the prediction of learning is a possible task and that it can be accurate as it is now feasible to apply certain statistical techniques on large amounts of data, “big data”, not previously available (Gandomi and Haider, 2015).

Concretely, there are several proposals for algorithms dedicated to prediction in education that has recently gained momentum at the university level (Arnold and Pistilli, 2012; Arnold, Tanes and King, 2010; Iten, Arnold and Pistilli, 2008; Jayaprakash, Moody, Lauria, Ragan and Baron, 2014; Gašević, Dawson, Rogers and Gasevic, 2016; Tanes, Arnold, King and Remnet, 2011; Sclater, Peasgood and Mullan, 2016; Pistilli and Arnold, 2010). They are the so-called “probability of success algorithms” or “student success prediction algorithms” from which the “risk of falling behind” is calculated for each student in the cohort. With these data, universities contact their students and recommend activities in order to mitigate the risks of desertion and abandonment.

In addition, exploratory analyzes also allow identifying relevant variables to predict some trends in student performance. These strategies are chosen rather than applying external models because there are some conceptual caveats about the impossibility to cheerfully extend algorithms created by other educational organizations (or what Cathy O'Neill (2016) calls the "scalability" problem).

On this occasion, we will work with this second approach and not with the application of student success prediction algorithms already developed by other organizations. As educational activities are "situational" by definition and because the generalization of research results is a significant challenge in this field (Gasevic, et al. 2016); the objective of this exercise is the exploration of educational data to identify trends in practices that may indicate academic desertion and dropout.

In this context, this project has two main objectives.

On the one hand, it is proposed the *construction and analysis of a relational database with data from university students that include their practices in the university Learning Management System, their scores and assessment results, and their socio-demographic characteristics*.

On the other hand, it will be created a web application prototype. The objective of this application is *to report to the institution administrators considering a description of the practices and educational performances as well as some predictive results that can justify specific interventions to mitigate abandonment and dropout*.

The actions performed in this project will be detailed below. First, we describe the characteristics of the data used. Second, the relational database design process will be detailed. Third, we will refer to the database creation and the data upload process using PostgreSQL pgAdmin. Fourth, we will display preliminary results of our exploratory data analysis using Falcon SQL to produce plots and charts. Fifth, the methodology used to produce the regression model will be presented, as well as the results of our prediction model of failing a course. Finally, we will describe the sequence of steps followed to develop the prototype of the web application intended to show university administrators the previously built report.

The data

In this project data coming from the Open University Learning Analytics Dataset (OULAD) will be used. It is important to mention that this British university has significant developments in the field of Educational Technology due to the majority of its students study off-campus (either in blended learning systems or directly in MOOCs).

Particularly, this dataset is composed of 7 relations saved in .csv format and available online (visit the webpage at https://analyse.kmi.open.ac.uk/open_dataset#about). This is open anonymized data and, as it is explained in the website, it contains information about courses, students, and their interactions with Virtual Learning Environment (VLE) for seven selected courses (called modules) as well as data about assessments and demographic information of students.

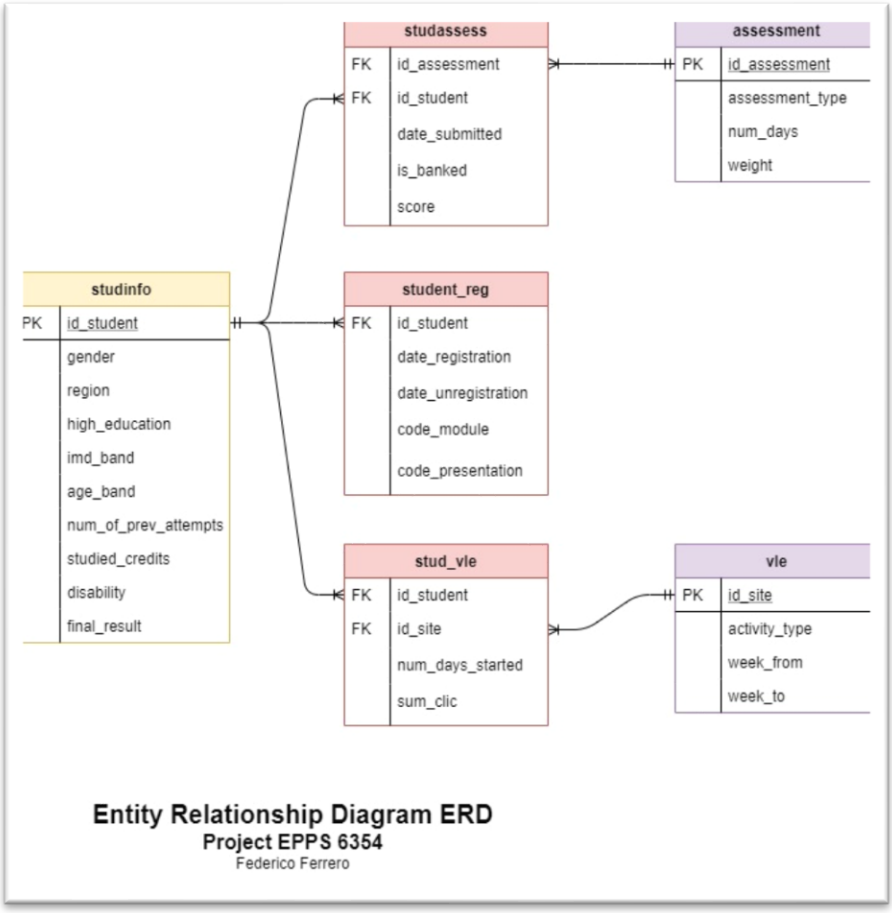
Having available these resources, a specific relational database design was considered taking into account the queries whose results will integrate the final report to the authorities of the educational institution. In the first moment, we focus on the treatment of raw data: an imputation method (using means of variables) was applied when values were missing, and subsequently, the elimination of certain variables not considered necessary for this project analysis was conducted.

The final database design considering the purposes of this project (see Figure 1) contains three big areas: Students' information; Students' practices (including practices at VLE, registration, and assessment); and description both of VLE and Assessments.

Furthermore, as can be observed in the Entity Relationship Diagram, considerations about cardinality can be made. Indeed, just to mention one example, the minimum number of assessments, registrations, and practices at VLE that a single student can have is one and the maximum is many; so, the link between the three relations about practices (in red) and the student information relation (in yellow)

is “one to many”. In turn, one and only one (different) student can have different assessments, registrations, and practices at VLE.

Figure 1: Entity Relationship Diagram



Creation of Open University database and population with data in pgAdmin

In third place, the database was created in PostgreSQL pgAdmin under the name of "Open University". Each one of the 6 relations was defined with its variable types and names as well as the respective primary keys and foreign keys (as indicated in ERD). After that, data was uploaded in pgAdmin for each one of the specified relations (see the example code for the "Student Info" relation in Figure 2 below).

Figure 2: Example of code to create the "Student Info" relation and to populate it with data

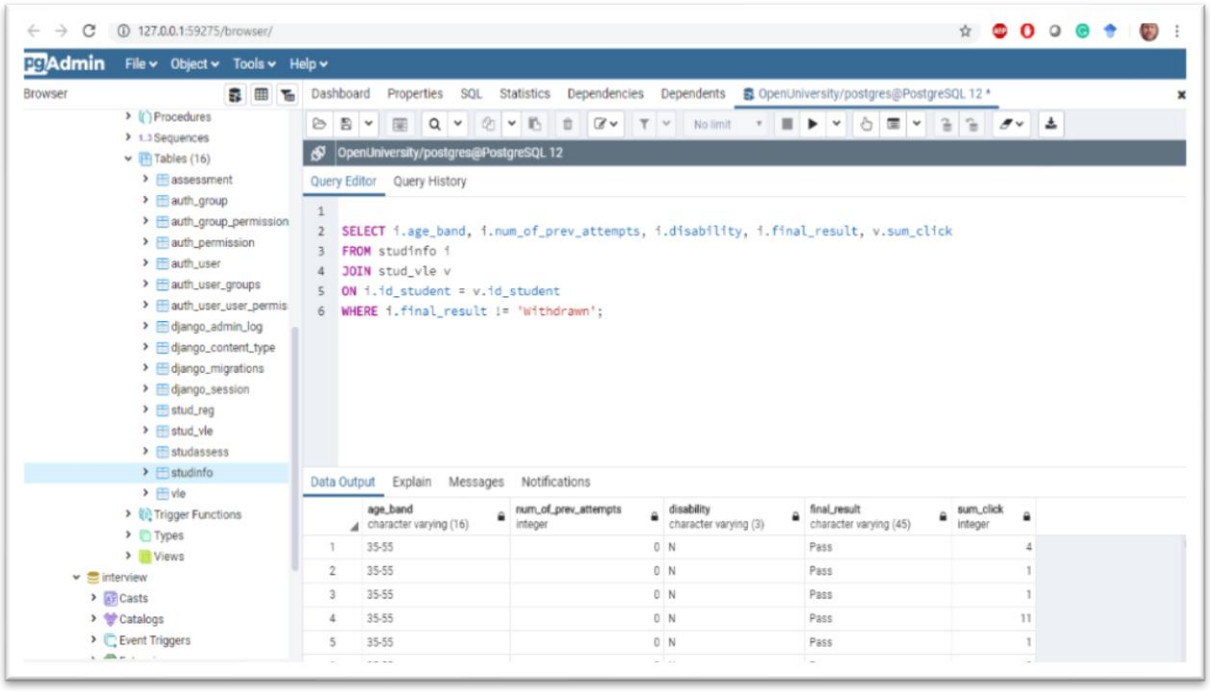
```
-- Create database
CREATE DATABASE "OpenUniversity"
WITH
OWNER = postgres
ENCODING = 'UTF8'
CONNECTION LIMIT = -1;

-- Create courses relation, import data and check the relation /
  CREATE TABLE studinfo
  (
      id_student integer NOT NULL,
      gender varchar(3),
      region varchar(45),
      highest_education varchar(45),
      imd_band varchar (16),
      age_band varchar(16),
      num_of_prev_attempts integer,
      studied_credits integer,
      disability varchar(3),
      final_result varchar(45),
    PRIMARY KEY (id_student)
  );


  COPY studinfo FROM 'C:\Users\feder\Desktop\SDAR\HO-SQL\OPEN UNIVERSITY DATA SET\studentInfo.csv' DELIMITER ',' CSV HEADER;
```

Once the database was completely created, linked, and loaded with the data, its correct operation was checked, as can be seen in Figure 3.

Figure 3: Example of query and outcome in PostgreSQL pgAdmin



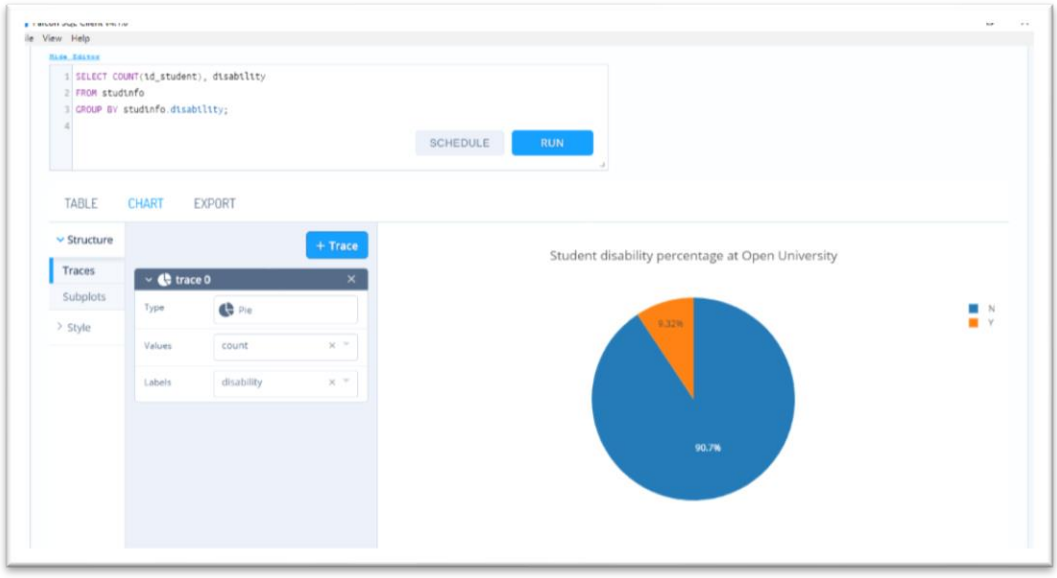
	age_band character varying (16)	num_of_prev_attempts integer	disability character varying (3)	final_result character varying (45)	sum_click integer
1	35-55	0	N	Pass	4
2	35-55	0	N	Pass	1
3	35-55	0	N	Pass	1
4	35-55	0	N	Pass	11
5	35-55	0	N	Pass	1

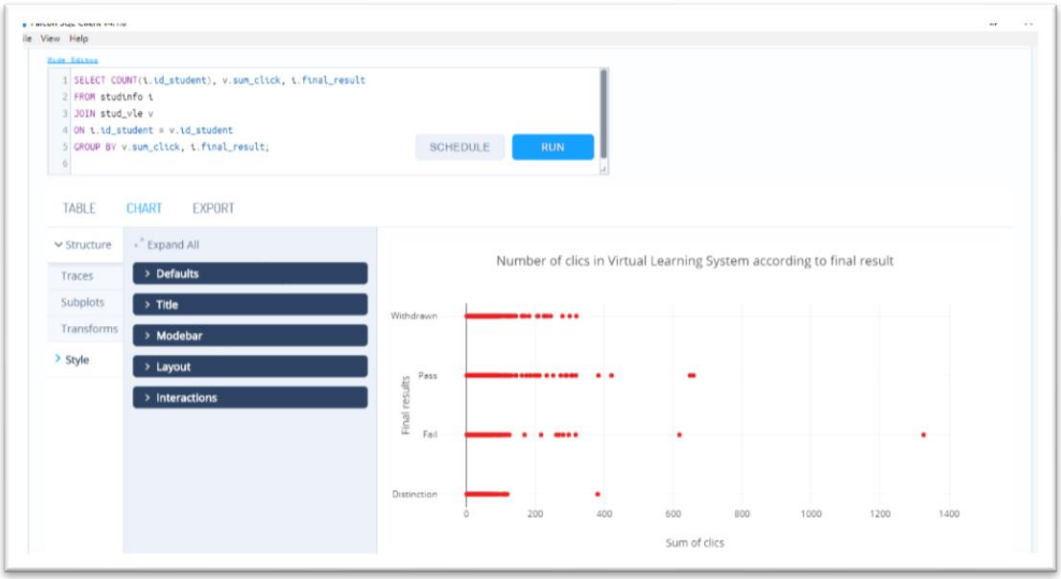
Results of the exploratory analysis using Falcon SQL

Authors like Jayaprakash, et al. (2014), Simpson (2006), and Arnold and Pistilli (2012) identify predictors of academic dropout. Just to mention some of them, in educational literature often it is considered the effects of disabilities, age group, time spent on Learning Management Systems, number of previous attempts before passing a course, among others.

In this case, our preliminary analysis will focus on these variables that usually predict academic dropout. Using Falcon SQL linked to our previously set pgAdmin database, simple queries were displayed to generate plots (see Figure 4).

Figure 4: Generation of chart and plots with Falcon SQL

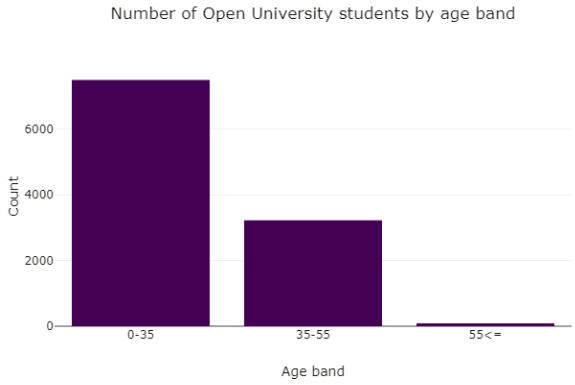




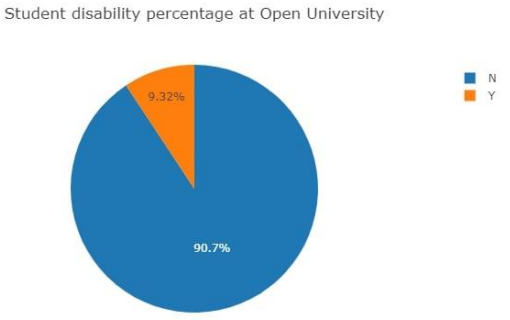
In light of the referenced predictors of dropout, the behavior of these variables can be explored in our database (see Figure 5).

In relation to age groups (diagram A), the majority of Open University students (more than 7000) are in the age range of 0-35 years, reaching around 3000 the number of students between 35 and 55 years. There is also a minimum proportion of students over 55 years old. On the other hand, regarding the presence of disabilities (diagram B), only 9.32% of registered students present them.

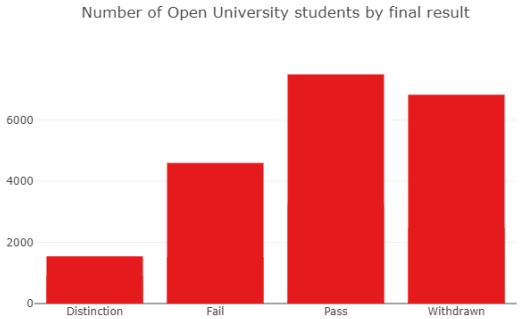
Figure 5: Graphics obtained with Falcon SQL



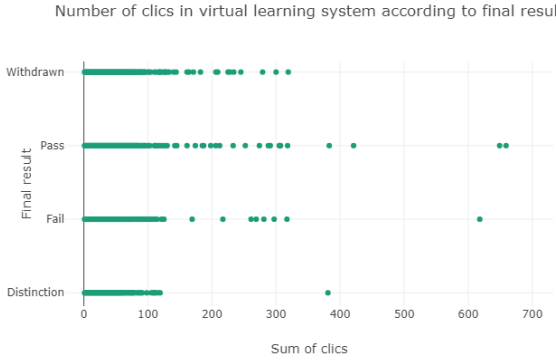
A



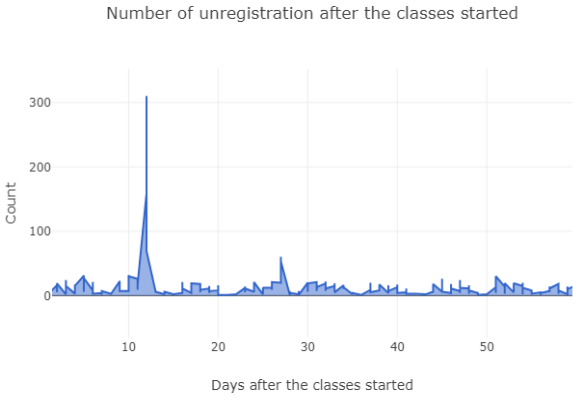
B



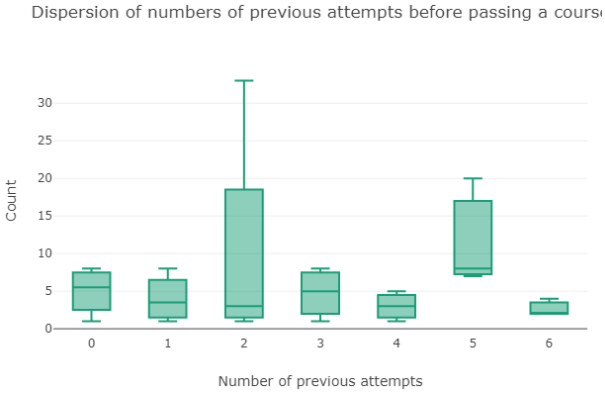
C



D



E



F

Considering now the number of students according to the final result obtained in each course (diagram C), we observe that the majority manages to pass (approximately 7000). In any case, the high number of withdrawn cases (around 6000) is noticeable, which indicates that it is a usual practice in the institution. It is followed in frequency level by the students who fail in their courses (they exceed 4000) for the period considered and those who obtain distinctions (approximately 1800 students).

Regarding the time dedicated to learning management systems, the number of clicks registered in the VLE can be taken here as an indication of the time spent on studying in the virtual environment (diagram D). Without pretending to be conclusive in this regard, if we analyze the number of clicks made by students according to the final results in the courses, we can observe that students who fail a course stop participating in the VLE at around 110 clicks. This finding can be suggestive considering the moment when the first signs of abandonment are seen. On the contrary, in the case of students who succeed in the course, the activity is more constant and more extended in its time.

In this sense, as diagram E shows, the number of unregistrations after the classes started are concentrated in two particular moments: around day 10 and later around day 25. Identifying these instances can be very helpful in achieving contact with students before they decide to unregister. Discussions in this regard are also addressed in Simpson (2006) since it is essential to identify the optimal date to produce pedagogical interventions that really avoid academic desertion.

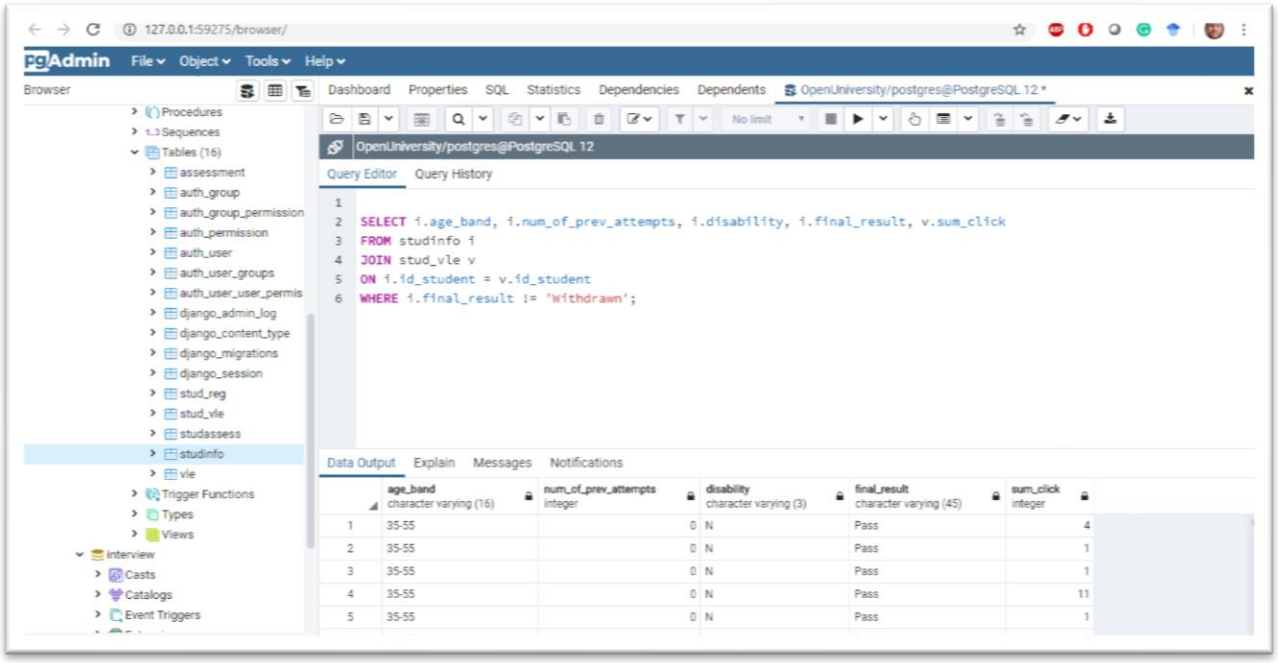
Regarding the dispersion of the number of previous attempts before passing a course we can see in diagram F a group of boxplots. Among them, the 2 previous attempts and the 5 previous attempts stand out as the quantities that bring together the most frequencies. The 2 previous attempts before passing a course show the highest number of cases as well as the highest dispersion, registering an interquartile range that extends from 2 to 18 students. Apparently 2 previous attempts seem to indicate the usual number required to pass a course after failing it.

Regression model: prediction of failing a course

With the previous results obtained from the data exploration, in this project, it has been decided to run a linear probability model (OLS) to predict failing a course according to our data.

To achieve this objective, a query in SQL was built to gather all the variables required in the model (Figure 6). Then the nominal variables were recoded to have all numerical values and the simple regression model was run by programming in R.

Figure 6: Extraction of dataset in pgAdmin to later run regression model at R Studio



The regression results are shown in Table 1. On the one hand, being part of the age group between 35-55 years and having a disability increases the likelihood of failing a course (these predictors are statistically significant). On the other hand, the number of previous attempts before passing a course and the sum of clicks at VLE are also statistically significant predictors but they reduce the likelihood of failing a course.

Table 1: Linear model outputs

	Dependent variable: **Failing a course**
Factor (35-55 years old)	0.11**
Factor (more than 55 years old)	0.06
Number of previous attempts before passing a course	-0.71*
Presence of disability	0.34*
Sum of clicks at VLE	-0.20*
Constant	2.413
Observations	*695,680*
R^2	*0.61*
Note: *p<0.1, **p<0.05, ***p<0.01	

These results can provide orientation to the university decision-makers about which are the practices and students' profiles that have effects on its academic success or failure.

Based on these results, groups of students under risk of dropping out can be detected and, subsequently contact them in order to provide adequate support. It is also important to highlight that, in addition to the predictors examined, the consideration of the average time of unregistration can be

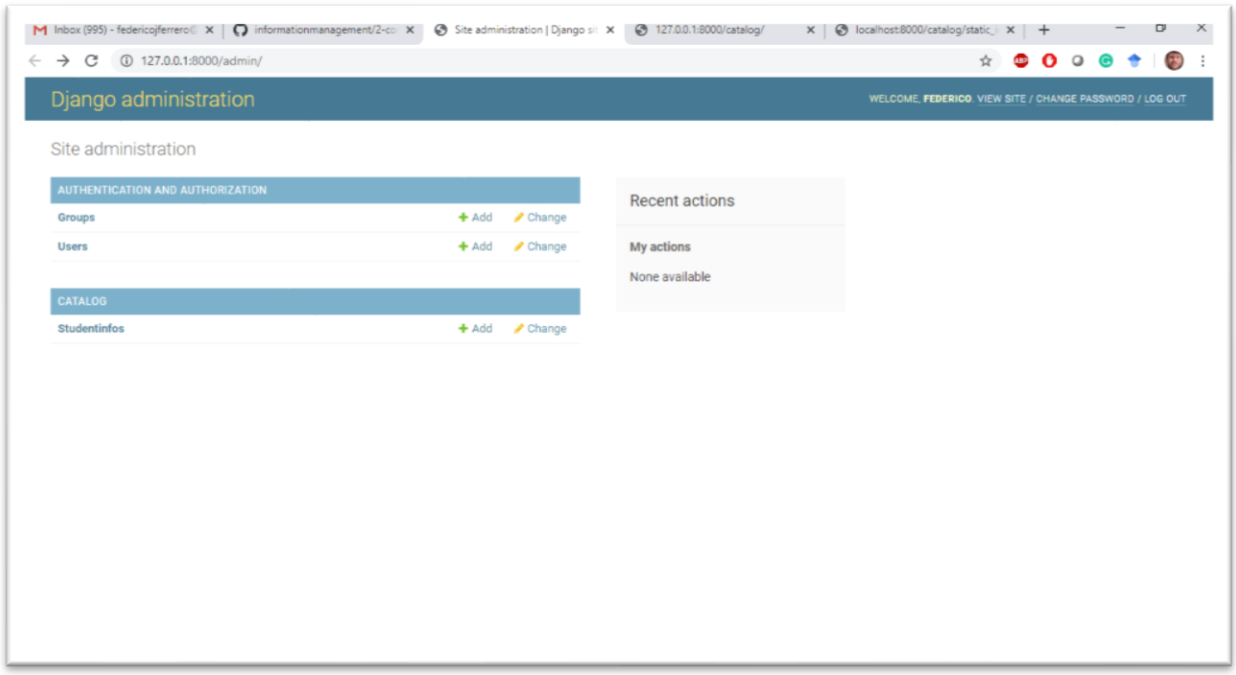
really useful. According to our particular results, it can be inferred that the opportune moment to contact Open University students is before the first 10 days of classes and before the end of the first month of classes.

Web application to report university administrators

The development of the demo web application will be described here according to the technical steps followed for its generation. Specifically, the prototype that will be shown below includes web pages that show lists of students with relevant variables and the graphics previously described. As stated before, this app is aimed at university administrators, university authorities in charge of making decisions, and teams whose function is providing support to students at risk of dropout.

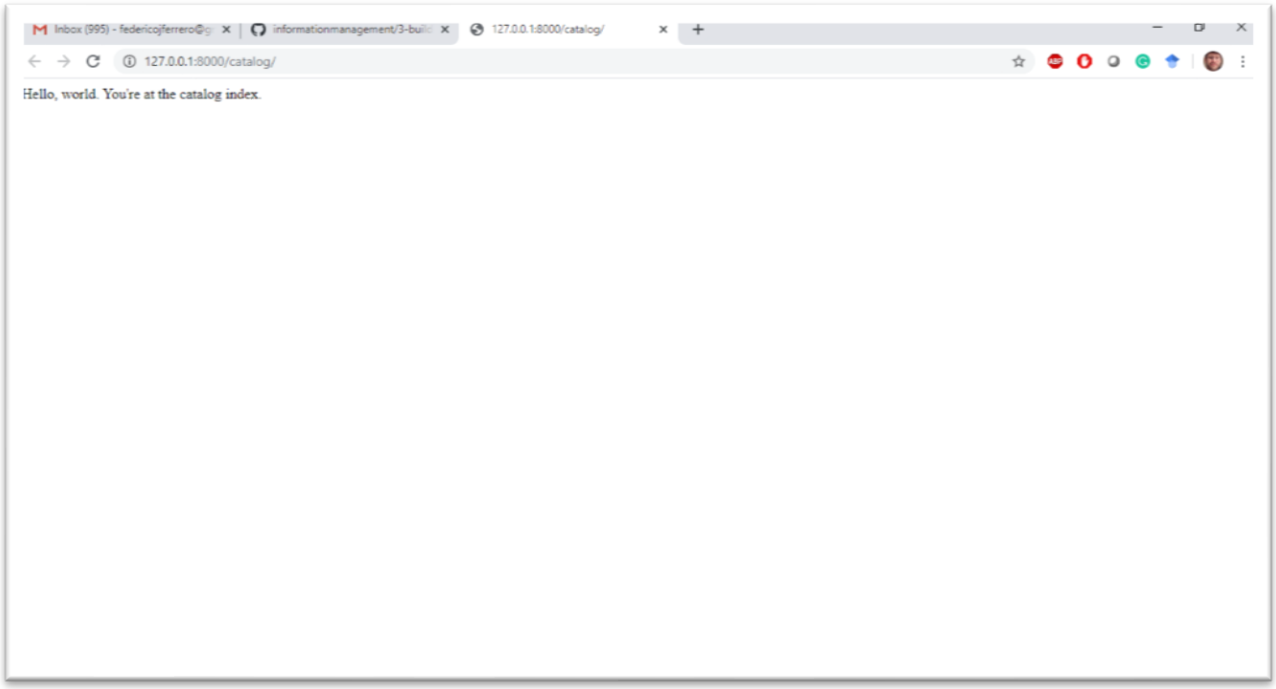
First, Django was installed and a project with "my site" name was created. After confirming that the server was running (http://127.0.0.1:8000/) we connected Django to our PostgreSQL pgAdmin Database ("OpenUniversity") and migrated the data[1].

Second, a super user account was created, and we connected to the database (http://127.0.0.1:8000/admin).

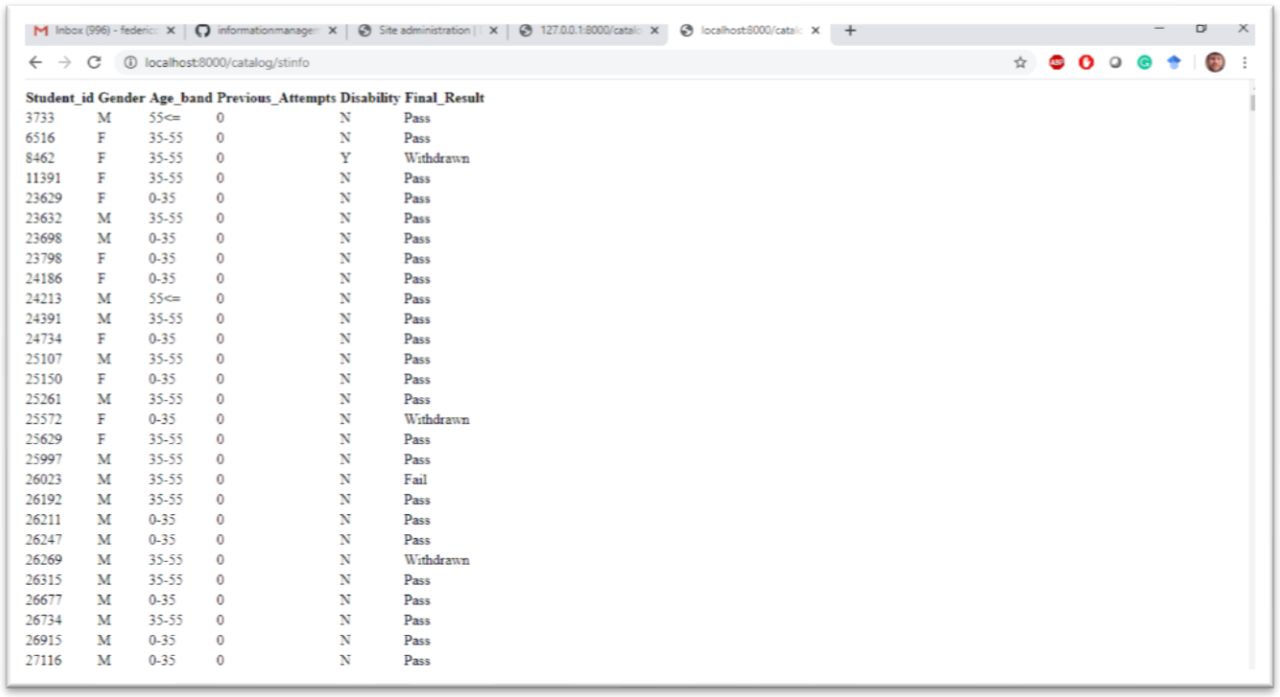


Fourth, we created a demo app named "catalog" which was checked as the screenshot shows below.

[1] The adaptation of the code according to the specific features of our database can be checked at the folder "epps6354".

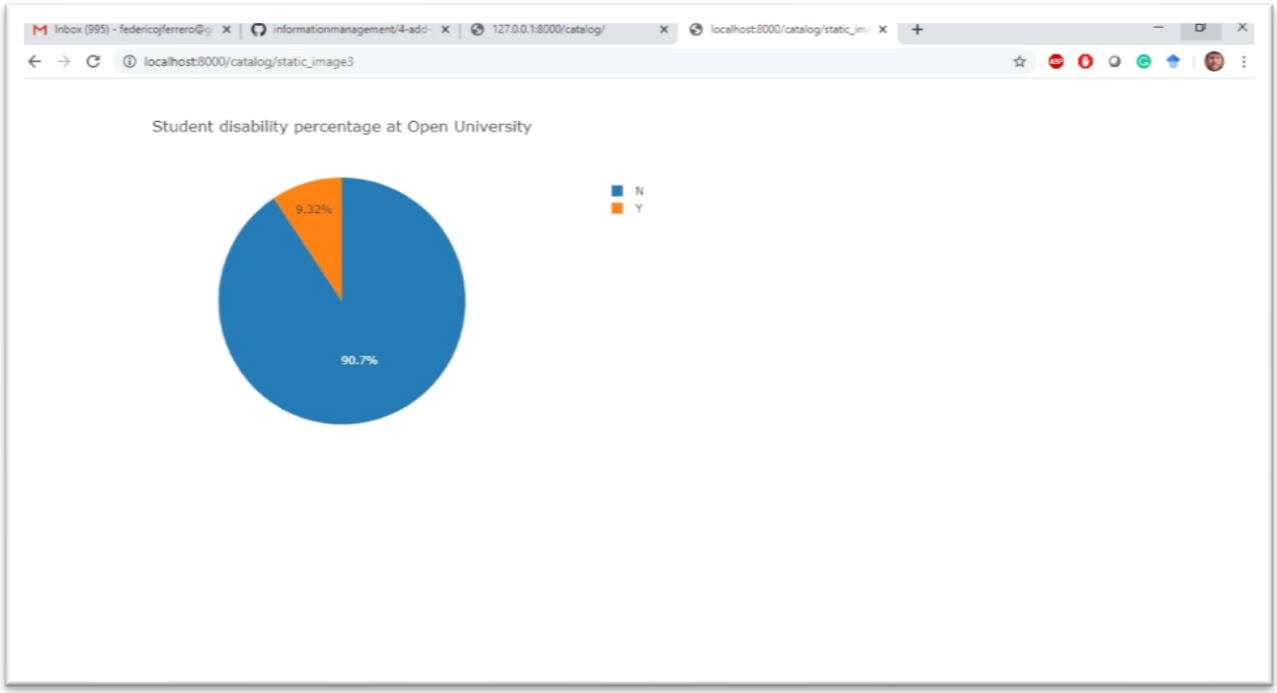


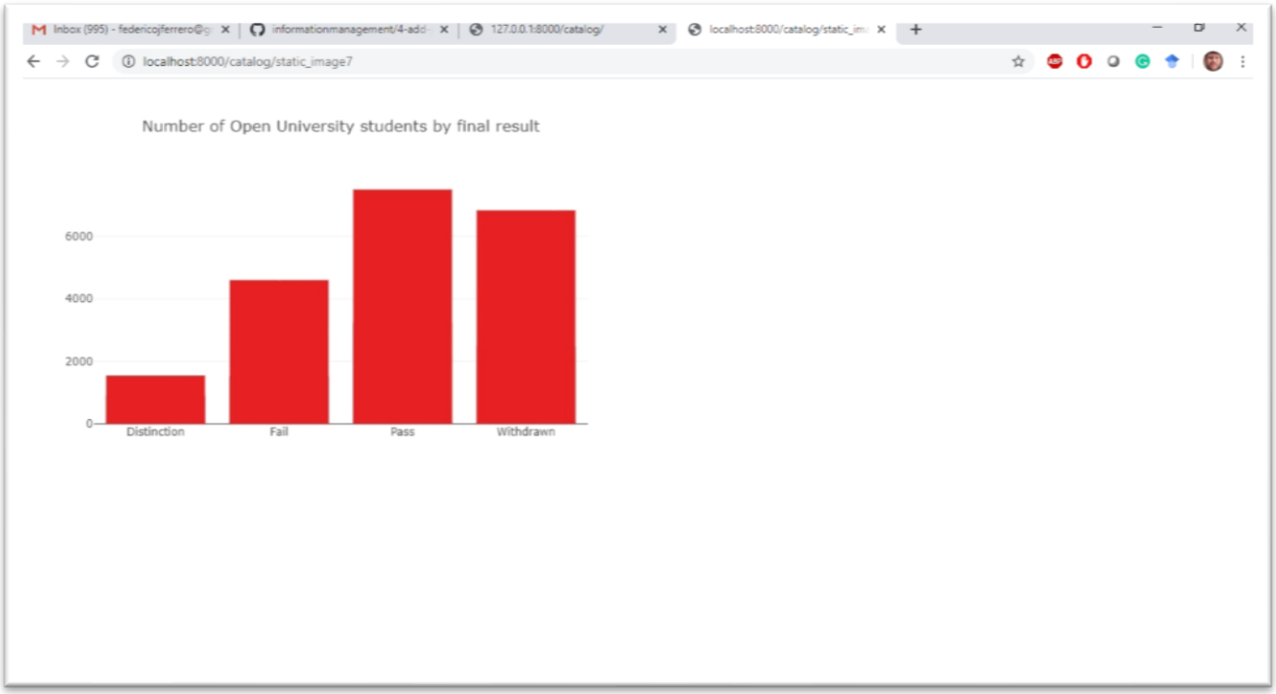
Fifth, to obtain the HTTP response, that is a table with data coming from our students; we modified several files (views.py, template.html, urls.py, models.py, admin.py, and settings.py) according to the specific information of our OpenUniversity database. When visited http://localhost:8000/catalog/stinfo the output was correct.

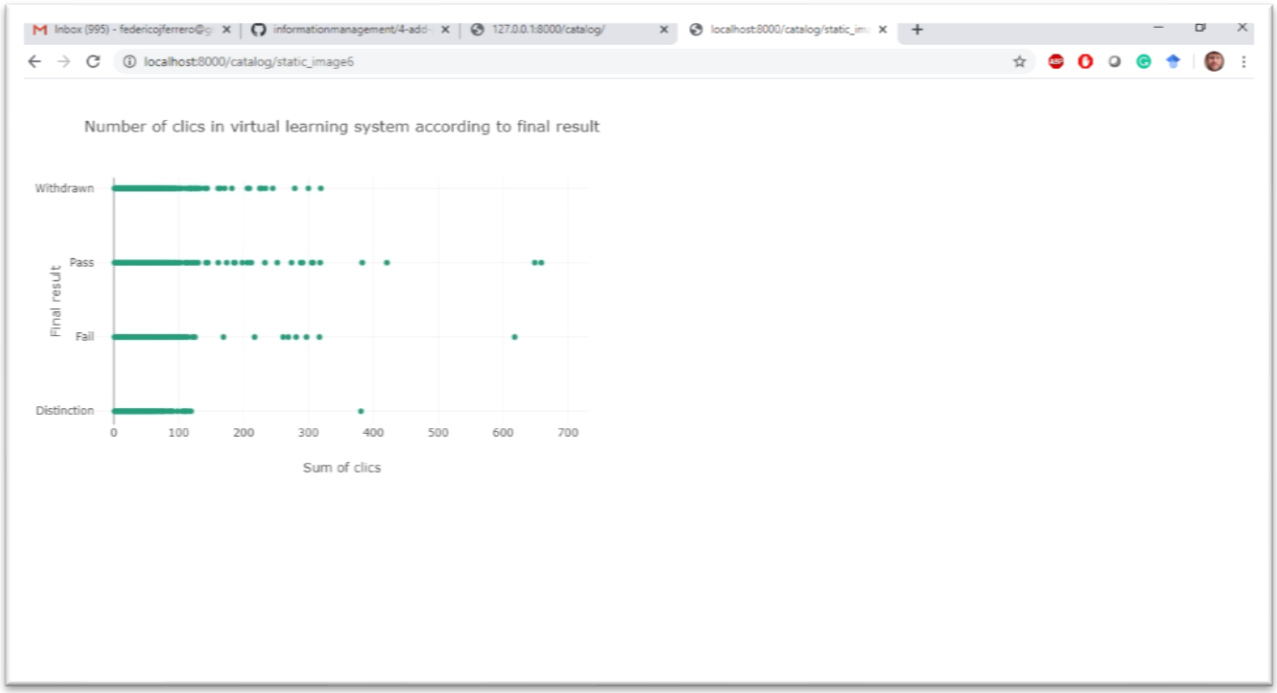


Student_id	Gender	Age_band	Previous_Attempts	Disability	Final_Result
3733	M	55<=	0	N	Pass
6516	F	35-55	0	N	Pass
8462	F	35-55	0	Y	Withdrawn
11391	F	35-55	0	N	Pass
23629	F	0-35	0	N	Pass
23632	M	35-55	0	N	Pass
23698	M	0-35	0	N	Pass
23798	F	0-35	0	N	Pass
24186	F	0-35	0	N	Pass
24213	M	55<=	0	N	Pass
24391	M	35-55	0	N	Pass
24734	F	0-35	0	N	Pass
25107	M	35-55	0	N	Pass
25150	F	0-35	0	N	Pass
25261	M	35-55	0	N	Pass
25572	F	0-35	0	N	Withdrawn
25629	F	35-55	0	N	Pass
25997	M	35-55	0	N	Pass
26023	M	35-55	0	N	Fail
26192	M	35-55	0	N	Pass
26211	M	0-35	0	N	Pass
26247	M	0-35	0	N	Pass
26269	M	35-55	0	N	Withdrawn
26315	M	35-55	0	N	Pass
26677	M	0-35	0	N	Pass
26734	M	35-55	0	N	Pass
26915	M	0-35	0	N	Pass
27116	M	0-35	0	N	Pass

Sixth, we added web pages to our demo web application with static images through the edition of urls.py and views.py as well as the creation of html pages. We verified the different results (the first web page below correspond to http://localhost:8000/catalog/static_image4)




localhost:8000/catalog/static_image7
Number of Open University students by final result
6000
4000
2000
0
Distinction
Fail
Pass
Withdrawn


localhost:8000/catalog/static_image6
Number of clics in virtual learning system according to final result
Withdrawn
Pass
Fail
Distinction
Final result
0
100
200
300
400
500
600
700
Sum of clics

Conclusions

In this project we have designed, built, and analyzed a database with data from the Open University with the aim of describing students' performances and generating predictions about how the phenomenon of dropout occurs in that institution. Then, a prototype of a web application has been

created with the purpose of being a report aimed to whom should implement pedagogical actions that mitigate the phenomenon of dropout at the university.

Specifically, we described: a) the characteristics of the data, b) the design process of the relational database, c) the preliminary analyzes around predictors of abandonment usually mentioned in educational literature, d) the results of the own prediction model of failing a course, and finally, e) the sequence of steps followed to develop the prototype of the web application intended to show university administrators the previously built report.

The results analyzed in this project can provide orientation to the university administrators about which are the practices and students' profiles that have effects on its academic possible failure. As was referred before, students between 35 and 55 years (probably workers) and students with disabilities are a risk group while those who spend more time in the VLE, as well as those who take courses a second time, are more likely to be successful.

Based on these results, groups of students with considerable risk of abandonment can be detected. After this identification phase, students may be contacted by specialists in order to provide them personalized pedagogical assistance and support (in the best case, before the first 10 days or 30 days of classes). In those instances, the possibility of carrying out a more detailed analysis of the particularities of each case would allow not to reduce the approach only to the variables valued here.

Finally, it should be said that the use of predictive strategies in education must be approached carefully since their results have an orientation value and are not considered unappealable determinants of the outcomes that students could achieve. On the contrary, they serve the purposes of guiding interventions when the magnitude of the students is, as in the case of the database used, unmanageable.

References

Arnold, K. E., and Pistilli, M. D. (2012, April). Course signals at Purdue: Using learning analytics to increase student success. In *Proceedings of the 2nd international conference on learning analytics and knowledge* (pp. 267-270). ACM.

Arnold, K. E., Tanes, Z., and King, A. S. (2010). Administrative perceptions of data-mining software Signals: Promoting student success and retention. *The Journal of Academic Administration in Higher Education*, 6(2), 29-39.

Breiter, A. (2016, July). Datafication in education: a multi-level challenge for IT in educational management. In *International Conference on Stakeholders and Information Technology in Education* (pp. 95-103). Springer, Cham.

Gandomi, A., and Haider, M. (2015). Beyond the hype: Big data concepts, methods, and analytics. *International journal of information management*, 35(2), 137-144.

Gasevic, D., Dawson, S., Rogers, T., and Gasevic, D. (2016). Learning analytics should not promote one size fits all: The effects of instructional conditions in predicting academic success. *The Internet and Higher Education*, 28, 68-84.

Iten, L., Arnold, K. and Pistilli, M. (*2008)*. Mining real-time data to improve student success in a gateway course. *Eleventh Annual TLT Conference*. Purdue University, Indiana.

Jayaprakash, S. M., Moody, E. W., Lauría, E. J., Regan, J. R., and Baron, J. D. (2014). Early alert of academically at-risk students: An open source analytics initiative. *Journal of Learning Analytics*, 1(1), 6-47.

O'Neil, C. (2016). *Weapons of math destruction: How big data increases inequality and threatens democracy*. Broadway Books.

Pistilli, M. and Arnold, K. (2010). Purdue Signals. Mining real-time academic data to enhance student success. *About Campus*, 15 (3), 22-24.

Sclater, N. Peasgood, A. and Mullan, J. (2016). *Learning Analytics in Higher Education. A review of UK and international practice*. London: Jisc.

Selwyn, N. (2015*). Data entry: Towards the critical study of digital data and education. Learning, Media and Technology.* 40 (1), 64–82.

Silberschatz, Abraham, Korth, Henry F. and Sudarshan, S., (2019). Database system concepts. 7th edition. New York: McGraw-Hill.

Simpson, O. (2006). Predicting student success in open and distance learning. *Open Learning: The Journal of Open, Distance and e-Learning*, 21(2), 125-138.

Tanes, Z., Arnold, K. E., King, A. S., and Remnet, M. A. (2011). Using Signals for appropriate feedback: Perceptions and practices. *Computers & Education*, 57(4), 2414-2422.

Van Dijck, J. (2014). Datafication, dataism and dataveillance: big data between scientific paradigm and ideology. *Surveillance & Society*, 12 (2), 197-208.

Resource used to construct the web application prototype:

Ho Karl Github account:
https://github.com/datageneration/informationmanagement/blob/master/workshop/ApplicationDevelopment/2-connect-database.md