Ecole Polytechnique Federale de Lausanne

Semester project

A free MOOC market: Predicting success of a course

Omar Boujdaria

supervised by
Ramtin Yazdanian

January 10, 2020

A free MOOC market: Predicting success of a course

R.Yazdanian, O.Boujdaria, EPFL

January 2020

Table

1 Introduction

Udemy is an online learning platform aimed at professional adults and students. The platform is host to video based, interactive courses in various fields as Business, Photography or Computer Science. This project aims to define and predict success for a course, based on metadata and textual content.

2 Dataset Description

Our dataset was parsed from Udemy in February 2019, and contains 9717 courses all in the field of Computer Science. It is presented as 3 tables :

- Course Details : listing the courses explored in this study, each row representing one distinct course.
- Course Syllabi : in this platform, courses are a collection of chapters, and chapters are collection of quizzes, lectures, and practices. This table represents the course components. (Approximately 600k course components in total)
- Course Reviews : listing the reviews available about the courses, which are grades ranging from 0 to 5, along with a text comment.

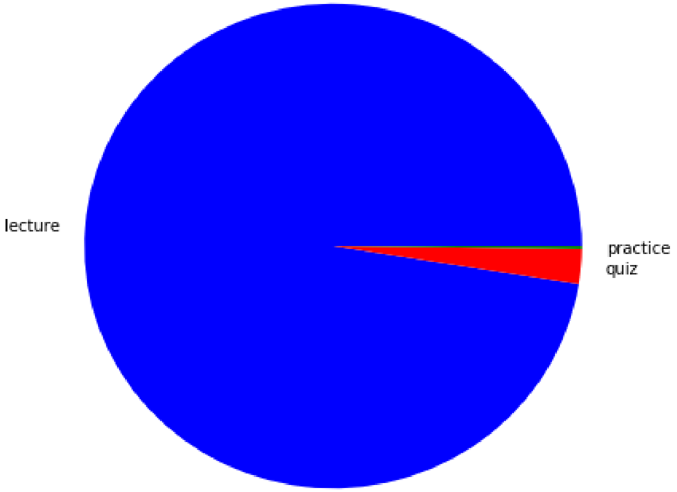


Figure 1: A plot of the course content proportions

2.1 Course Details

The CourseDetails dataframe has 21 columns for 9717 records.

avg_rating	Average rating (from 1 to 5)
avg_rating_recent	Average rating obtained on a recent set of ratings
created	Date and time of creation of the course
is_paid	Boolean, True if the course has a price, ie not for free
last_update_date	Last modification to the course
num_lectures	Number of lectures
num_practice_test	Number of published practice tests
num_quizzes	Number of published quizzes
num_reviews	Number of total reviews
num_reviews_rec	Number of recent reviews
num_subscribers	Number of overall subscribers
objectives	List of sentences describing the goal of the course
prerequisites	List of knowledge or material needed to take the course
price	String, the price in dollars, or 'Free' if the course is not paid
published_time	First date the course was published
rating_dist	Dict of each rating and its count (1 to 5)
target_audiences	List of sentences describing the audience it was made for and what they can find in this course
title	String, Title of the course
url	String, url of the course at Udemy website
content_length	Duration of the course in hours
publication_gap	The time between course creation and course publication, course publication means going online

2.2 Course Syllabi

Each row from the syllabi represent a part of a course (a video, a quizz, a practice or can be course title or chapter title) from any course, the dataframe has 15 columns for 604.683 rows.

course_id	the course this video is related to
chapter_tiltle	chapter where the content is included, if the line represent a chapter or a course, this field is NaN
chapter_desc	chapter description, mostly empty
chapter_created	time of creation of chapter, NaN if the line representsa chapter or a course
content_class	if the row is a lecture, a quizz, a pratice, a chapter, or a course
title	Title of the part of the course
description	description, courses have no description, others are mostly empty
created_content	time of creation
is_published	is published, either True, or Nan
content_summary	duration if video, size if file, or pages if document
url	url of the course part

2.3 Course Reviews

Course reviews lists 2.762.845 reviews for 9143 courses, capped at 10000 rating per course.

course_id	the course being reviewed
created_date	date and time the review was posted
rating	a grade ranging from 0 to 5 (0.5 step)
text	textual comment, not mandatory, only present for 592.607 of the ratings

3 Data Exploration

3.1 Topic Detection, assignment, modelling

The goal of this part is to better understand the data by assigning a topic to each course, for further study of topic impact of a course success. GensSim module is used in this part.

The potential sources of text that can be used for topic detection are :

- Course Title
- Course objectives
- Course prerequisites
- Course content titles
- Course content descriptions

Gensim's available preprocessing methods are not suited for this dataset, as a lot of short, yet important keywords are suppressed. Examples are "C" or "C++", "PHP", "R"...
The text data preprocessing pipeline is described bellow :

1. Removing punctuation
2. Lowercasing
3. Tokenization
4. Removing english stopwords
5. Lemmatizing and stemming of words longer than 3 characters
6. Further removal of other words that are independent of the topic but abundant in a course dataset, for example learning related (as "learn","build") or quality related (as "good", "best"), the complete list of these words can be found in the Appendix.

Our first tentative topic modelling is made through Latent Dirichlet Allocation. The basic idea behind LDA is that this algorithm defines each topic as represented by an unknown set of words. These are the topics that our documents cover, but we don't know what they are yet. LDA tries to map all the (known) documents to the (unknown) topics in a way such that the words in each document are mostly captured by those topics.

The fundamental assumption here is that documents with the same topic will use similar words. It's also assumed as well that every document is composed by a mixture of topics, and every word has a probability of belonging to a certain topic.

Unfortunately, LDA results were not as conclusive as expected, as topic's were poorly interpretable, almost meaningless.

We move to Topic Assigning after creating a list of potential topics and running a rule-based automated assigning of the topics.

A course will have a general topic, one from :

- Programming Language
- Web and Design
- DataBase systems

And will also have a more refined topic, from a larger set of topics, and depending on the general topic. The potential refined topics are mainly programming languages or software to be learned through the concerned course.

As only 5% of the courses remained with an undetermined topic, they are dropped. The cause for the undetermined topic is that they are too specific and/or too different from the dataset by their content.

3.2 Ratings Distribution and Reviews Sentiment Analysis

We can see that most reviews are positive with 53% being 5 stars, and 85% of reviews being 4 stars or higher. While mostly favorable reviews are given, bad reviews are less abundant : a user is more willing to review if he has a good opinion about the course.

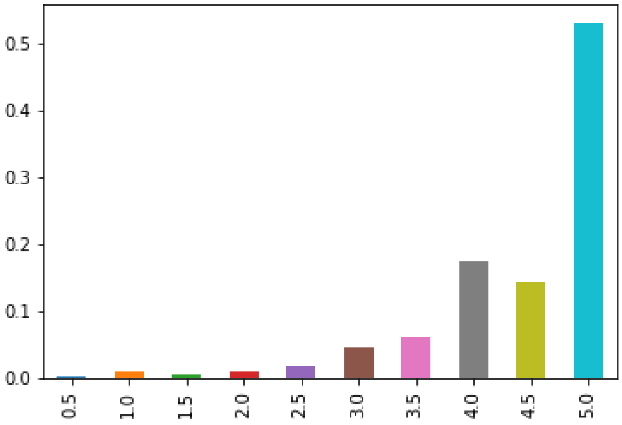


Figure 2: Ratings Distribution

Next, we will be analysing the sentiments and we will compare ratings with text against ratings without, as we investigate the following hypothesis:

Hypothesis : Do users who take time to write constructive reviews, as opposed to users who give a bare grade, tends to be less appreciative of the course ?

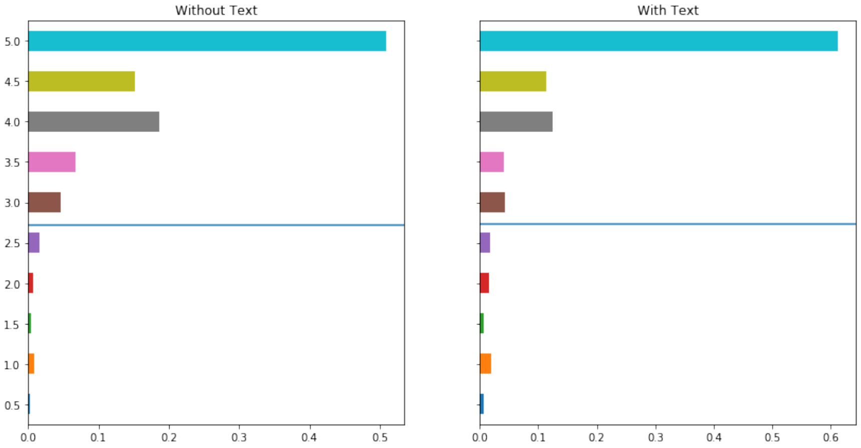


Figure 3: Ratings distribution for textual and non-textual reviewss

We study sentiments in textual reviews. To do so, we 'll be using the Vader model, which a pretrained NLP model from the nltk package.

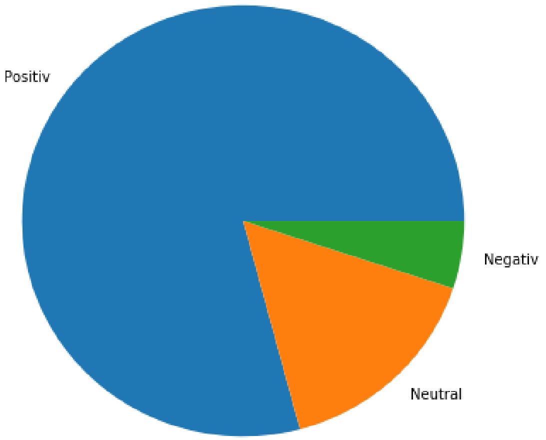


Figure 4: Sentiments proportions among textual reviews

Results : Looking at the ditribution of commented vs uncommented ratings, we see that both groups have routhly the same distribution and mean. Our hypothesis appear to be incorrect.

3.3 Difference between Paid and Free courses

In this section we will be investigating the difference in the number of subscribers between paid and free course.

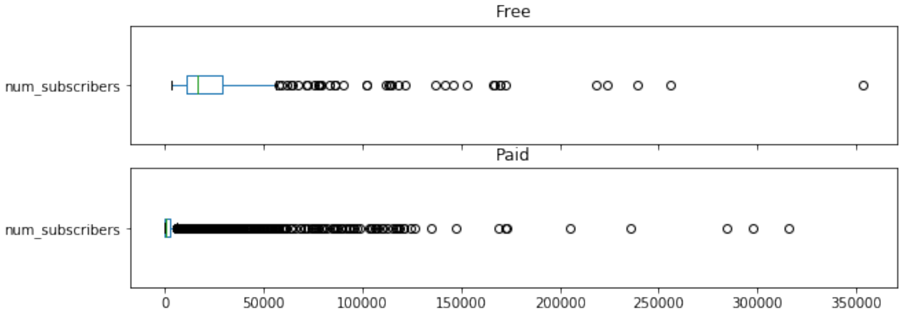
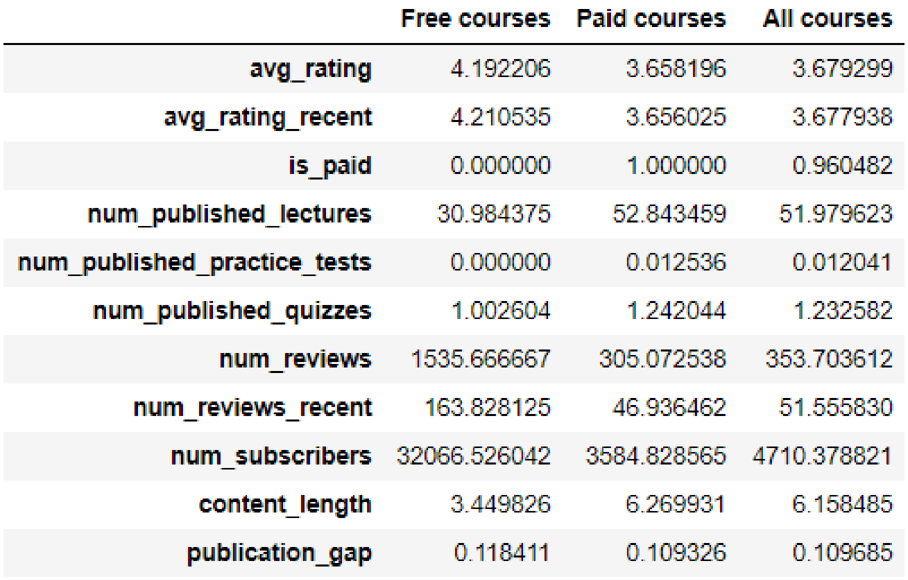


Figure 5: Distributions of NumberOfSubscribers for Free and Paid courses

We observe in Figure 5 that both distributions of number of subscribers for free and paid courses are very skewed.



	Free courses	Paid courses	All courses
avg_rating	4.192206	3.658196	3.679299
avg_rating_recent	4.210535	3.656025	3.677938
is_paid	0.000000	1.000000	0.960482
num_published_lectures	30.984375	52.843459	51.979623
num_published_practice_tests	0.000000	0.012536	0.012041
num_published_quizzes	1.002604	1.242044	1.232582
num_reviews	1535.666667	305.072538	353.703612
num_reviews_recent	163.828125	46.936462	51.555830
num_subscribers	32066.526042	3584.828565	4710.378821
content_length	3.449826	6.269931	6.158485
publication_gap	0.118411	0.109326	0.109685

Figure 6: Average statistics over Free and Paid courses

First, we see the predominance of the paid courses as they represent 96% of this dataset. With this big proportion being paid, we can see that the means of the paid group are very close to the means of the full dataset.

We observe that for success indicators, free courses are dominating: in terms of total subscribers, the free courses hit 9 times more subscriber and 5 times more reviews than the paid ones on average.

Conversely, free courses also seems to be less diligent in their content : the content length is doubled for paid courses, with 6h15 of lectures on averages for paid courses against 3h30 for free courses. The number of lectures is also higher for paid courses (+70%).

Remark : Further analysis will be done over the paid set only.

4 Defining success :

In order to define success, we will aim to find a metric that will divide the courses in 2 distinct clusters. To do so, we define several metrics that we will be analysing.

4.1 Tentative Metrics

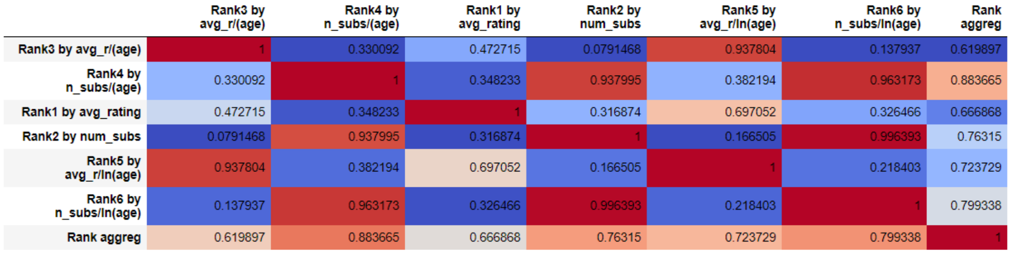
- Average rating
- Number of subscribers
- Average rating divided by the age of the course in days
- Number of subscribers divided by the age of the course in days
- Average rating divided by a logarithmic function of the age of the course in days
- Number of subscribers divided by a logarithmic function of the age of the course in days

4.2 Choosing the metric

As all these metrics are plausible, we aggregate them by ranking the courses by each of these metrics and combining these rankings.

One way to combine the 6 rankings, is to minimize the distance to all 6 ranks: ⟶ By Mean of rankings

We want our output rank to be such that each course rank is at minimal total distance of its input ranks.



	Rank3 by avg_r/(age)	Rank4 by n_subs/(age)	Rank1 by avg_rating	Rank2 by num_subs	Rank5 by avg_r/ln(age)	Rank6 by n_subs/ln(age)	Rank aggreg
Rank3 by avg_r/(age)	1	0.330092	0.472715	0.0791468	0.937804	0.137937	0.619897
Rank4 by n_subs/(age)	0.330092	1	0.348233	0.937995	0.382194	0.963173	0.883665
Rank1 by avg_rating	0.472715	0.348233	1	0.316874	0.697052	0.326466	0.666868
Rank2 by num_subs	0.0791468	0.937995	0.316874	1	0.166505	0.996393	0.76315
Rank5 by avg_r/ln(age)	0.937804	0.382194	0.697052	0.166505	1	0.218403	0.723729
Rank6 by n_subs/ln(age)	0.137937	0.963173	0.326466	0.996393	0.218403	1	0.799338
Rank aggreg	0.619897	0.883665	0.666868	0.76315	0.723729	0.799338	1

Figure 7: Rankings Correlation

Finally, we will only keep the Rank4 **(n_subs/age)**, as it is the most correlated one with the aggregated rank. We now plot the distribution of **(n_subs/age)** over the dataset.

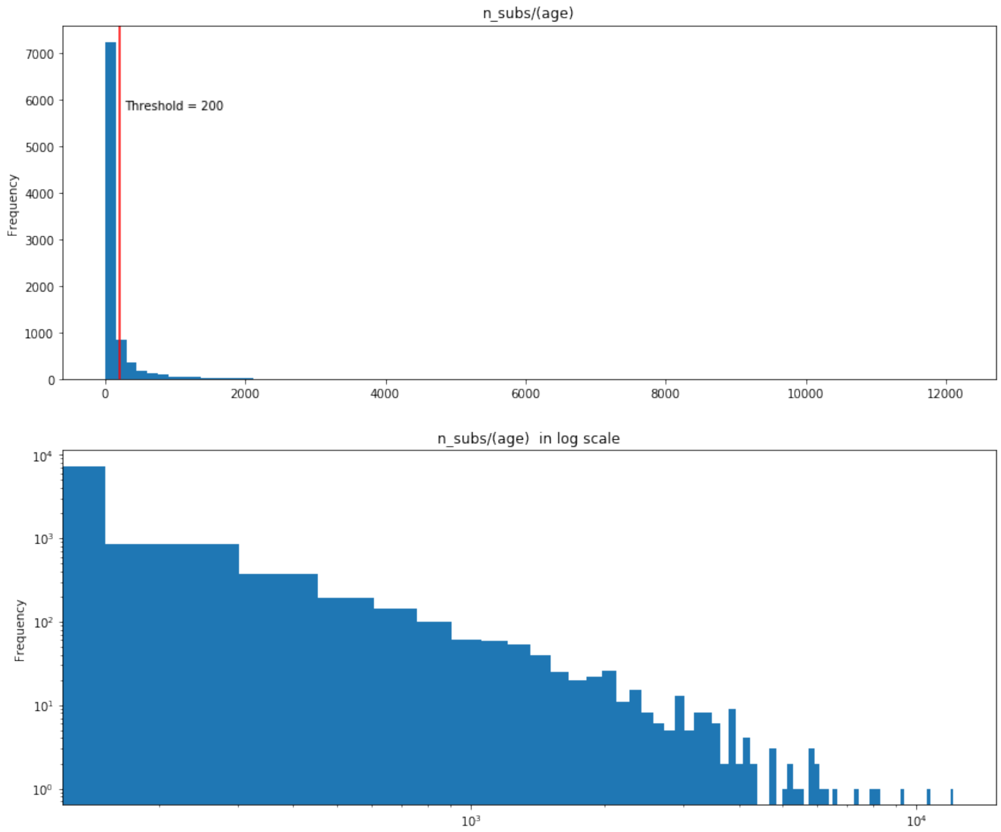


Figure 8: Selected metric, respectively in lin-lin and log-log scales

We have 18.5% of the courses that score higher than 200 at this metric (**(n_subs/age)**). We label them as successful and we now have our dependant variable

We have now defined a successful set of courses.

5 Measuring and Predicting success

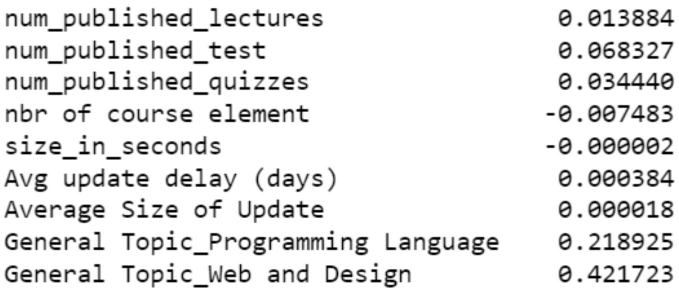
To predict success, one can only take into account features that are available at the creation and publication of the course. We will consider that a course has a scheduled of updates at the time of publication.

5.1 Features

- Number of lectures
- Number of quizzes
- Number of practice test
- Total number of elements available in the course
- Size of the course, in seconds
- Average update delay, representing the expectation of the delay at which the course is updated, in days
- Average update size

5.2 Classification

In order to get interpretability, we will use a Logistic regression classifier. Bellow are the weights of the corresponding features :



```
num_published_lectures               0.013884
num_published_test                   0.068327
num_published_quizzes                0.034440
nbr of course element               -0.007483
size_in_seconds                     -0.000002
Avg update delay (days)              0.000384
Average Size of Update               0.000018
General Topic_Programming Language   0.218925
General Topic_Web and Design         0.421723
```

Figure 9: Regression Weights

We see that the general topic WebDesign appear to be the most succesfull one, followed by Programming Languages. Database Systems topic being the least performant.

Appendices

Meaningless words : Here a list of lemmatized, stemmed words that were removed in preprocessing :

- instal; plus; cross; save; anyday; by; creat; the; post; solut; make; alongsid; nearer; complet; the; one; templat; amongst; collect; opposit; on; solut; less; aboard; introduct; absent; around; hour; circa; workshop; apud; astrid; project; definit; notif; conclus; event; below; abov; good; learn; in; upon; guid; into; unlik; underneath; from; method; nearest; and; throughout; like; build; program; but; versus; seven; through; per; languag; better; until; cours; test; with; two; here; preview; intermedi; bad; start; advanc; basic; modern; zero; beginn; introduc; nine; off; get; short; project; principl; real-world; to; across; among; minus; four; build; of; amidst; befor; project; along; get; fundament; video; without; despit; insid; step; standard; near; part; develop; basic; environ; dure; midst; welcom; upsid; with; tutori; fundament; beyond; mid; out; start; master; up; overview; direct; three; pro; apropo; intro; total; essenti; thru; setup; free; come; atop; about; learn; dive; scratch; navig; section; tag; complet; against; third; form; some; after; easi; resourc; edit; this; besid; beneath; behind; import; initi; bonus; lesson; over; ontop; except; onto; five; for; toward; under; input; lesson; till; at; down; depth; theori; second; basic; pre; code; amid; advanc; develop; learn; vice; first; step; build; mini; bonus; pretic; than; cod; between; outsid; creat; topic; besid; past; summari; compon; basic; let; as; understand; over; learn; eight; introduct; practic; lectur; ten; exampl; materi; six; everyday; start; sinc; extra; via; vs; toward; within; profession; cours

Source code of the project The source code of the project is available at Github :

`www.github.com/OmarBoujdaria/A-free-MOOC-market`

References

[1] `www.towardsdatascience.com/`

[2] `www.Udemy.com, of course...`

[3] `www.overleaf.com for the production of this document.`