

**EXPLORATORY DATA ANALYSIS**

**ENGINEERING CLINIC PROJECT REPORT**

***Submitted by***

**MURALI KRISHNAN.P (18BCS001)**

**THANUMALYEN.K (18BCS002)**

**ISHA KARTHIKA.D.A (18BCS003)**

**MANJU.S (18BCS007)**

**MRIDULA.C.S (18BCS008)**

***In partial fulfillment for the award of the degree***

*of*

**BACHELOR OF ENGINEERING**

in

**COMPUTER SCIENCE AND ENGINEERING**

**KUMARAGURU COLLEGE OF TECHNOLOGY**

**COIMBATORE-641 049**

(An Autonomous Institution Affiliated to Anna University, Chennai)

**December 2020**

### TABLE OF CONTENTS

|  |  |  |  |
| --- | --- | --- | --- |
| **CHAPTER NO.** | **TITLE** | | **PAGE NO.** |
|  | **ABSTRACT** | | 3 |
| 1 | **INTRODUCTION** | |  |
|  | 1.1 | INTRODUCTION | 3 |
|  | 1.2 | OBJECTIVES OF THE PROJECT | 4 |
|  | 1.3 | SCOPE OF THE PROJECT | 4 |
| 2 | **LITERATURE REVIEW** | |  |
|  | 2.1 | LITERATURE REVIEW OF JOURNALS | 5 |
| 3 | **PROBLEM DEFINITION** | | 10 |
| 4 | **EXPLORATORY DATA ANALYSIS** | |  |
|  | 4.1 | UNDERSTANDING THE VARIABLES | 10 |
|  | 4.2 | ANALYSIS OF VARIABLES | 11 |
|  | 4.3 | MISSING VALUES TREATMENT | 13 |
|  | 4.4 | OUTLIER TREATMENT | 14 |
|  | 4.5 | DATA VISUALIZATION | 18 |
|  | 4.6 | VARIABLE TRANSFORMATION | 28 |
|  | 4.7 | PREDICTION OF TARGET VARIABLE - 1 | 28 |
|  | 4.8 | PREDICTION OF TARGET VARIABLE - 2 | 30 |
| 5 | **CONCLUSION** | | 33 |
| 6 | **REFERENCES** | | 33 |

**ABSTRACT**

Online learning has become an increased practice among students and professionals. Online learning refers to taking up courses through the internet. The increase in number of people taking up online courses is because the biggest difference between traditional and online learning is the fact that students can learn from anywhere and can manage their time accordingly to complete all the assessments and gain a certificate. There are a number of popular online learning platforms like LinkedIn Learning, Udemy, Skillshare, Coursera, Codecademy, edx, etc. In this project, dataset of Udemy courses has been used to analyze and infer various information about the data. Data visualization has been done for more precise and easy inference of data. This project is based on analyzing the attributes using various python libraries like numpy, pandas, matplotlib, seaborn and scikit-learn. A prediction on number of subscribers has been made by using linear regression algorithm based on other attributes like number of reviews, number of lectures, content duration and whether the course is paid or unpaid. Another prediction is made for finding the content duration based on number of lectures using linear regression algorithm. Finally, conclusions were drawn according to the metrics derived.

1. **INTRODUCTION**
   1. **INTRODUCTION**

In this project, Udemy courses dataset is being used for Exploratory Data Analysis. As of January 2020, Udemy platform has more than 35 million students and 57,000 instructors teaching the courses in over 65 languages. There have been over 400 million course enrollments in the platform. The students and instructors come from more than 180 countries around the world. The dataset used in the project has about 3500 records and it presents about the course ID, title of the course, course url, number of subscribers, number of reviews, number of lectures in the course, content duration of the course, level (beginner, intermediate, expert or all), published timestamp and subject. It also defines whether the course is paid or not and also the price of the courses. In this project, Exploratory Data Analysis of the dataset is done and prediction has been done.

* 1. **OBJECTIVES OF THE PROJECT**

The objective of this project is to analyze the dataset and infer various information like average number of learners in each courses, number of reviews provided for each courses based on whether the course is paid or unpaid, to explore more information based on whether the course is paid or unpaid, to visualize the inferences taken from the dataset and to perform all the steps involved in Exploratory Data Analysis. Once the dataset is preprocessed it becomes ideal for prediction using Scikit-learn which is a free software machine learning library for Python programming language. In this project, two predictions are made and the machine learning model is trained using Linear Regression algorithm which is a type of supervised leaning in machine learning. The first prediction is to predict the number of subscribers based on number of reviews, number of lectures, content duration and whether the course is paid or unpaid. The second prediction is to predict the content duration based on number of lectures.

* 1. **SCOPE OF THE PROJECT**

Using numpy, a library for Python programming language to perform mathematical operations on arrays.

Using pandas, a software library written for the Python programming language for data manipulation and analysis.

Using Matplotlib, a plotting library for the Python programming language.

Using Seaborn, a Python data visualization library.

Using Scikit-learn, a free software machine learning library for Python programming language.

1. **LITERATURE REVIEW**
   1. **LITERATURE REVIEW OF JOURNALS**
2. **END 2021: International Conference on Education and New Developments 2021**

Abstract:

Due to the COVID-19 pandemic, we decided to shift to a full virtual event in 2021, since the health and safety of our attendees is our top priority.

We will have sessions with live presentations throughout the 3 days of the conference (June 26 to 28). The authors will be able to present their contributions virtually, either live (online platform) or by sending a pre-recorded video presentation - http://end-educationconference.org/covid-19-update/

Conference Goals:

Education, in a global sense, is a right since we are born. Every experience has a formative effect on the constitution of the human being, in the way one thinks, feels and acts. One of the most important contributions resides in what and how we learn through the improvement of educational processes, both in formal and informal settings.

Our International Conference seeks to provide some answers and explore the processes, actions, challenges and outcomes of learning, teaching and human development. Our goal is to offer a worldwide connection between teachers, students, researchers and lecturers, from a wide range of academic fields, interested in exploring and giving their contribution in educational issues.

The International Conference on Education and New Developments 2021 invites you to showcase presentations, from the most diverse countries and cultures, to promote growth in research methods intimately related to teaching, learning and applications in Education nowadays. We wish to illustrate developments and new ways of thinking in these main fields: Teachers and Students; Projects and Trends; Teaching and Learning; Organizational Issues (see below in "Themes of Interest").

The conference will cover several types of presentation formats, incorporating Invited Keynote Speakers, oral and thematic showcases. Conference materials will be sent to all registered attendees.

A book with selected papers from the conference will be published.

1. **SEEL 2021: 8th International Conference on Smart Education and E-learning**

Abstract:

Due to the ongoing COVID-19 pandemic, SEEL 2021 will be planned as a wholly virtual conference moving forward. More details will be updated on the website and LinkedIn in due course.

Aim:

SEEL is a major international forum for the presentation of innovative ideas, approaches, technologies, systems, findings and outcomes of research and design and development projects in the emerging areas of smart education, strategies and approaches for smart teaching, learning, and e-learning, applications of smart technology and smart systems in education and e-learning, smart classrooms and smart universities, and knowledge-based smart society.

SEEL will provide an excellent opportunity for scholars, Ph.D. students, faculty, administrators, and practitioners to meet well-known experts from all over the world and to discuss innovative ideas, approaches and novelties in smart teaching and learning, advances in smart technology, intelligent systems and state-of-art software and hardware systems for education, research findings and outcomes, best practices and case studies, national and international projects, institutional standards and policies, and many other topics related to innovative smart education, smart universities and smart schools.

1. **Using Codecademy Interactive Lessons as an Instructional Supplement in a Python Programming Course**

|  |  |
| --- | --- |
| |  | | --- | |  |   **Abstract:**  Codecademy is a great interactive website that gives you mini tasks to solve, each task getting more difficult as you proceed through the course. You also have to create a mini program/website (depending on what you learn) at the end of each section. The purpose of this paper is to describe the author’s experience using Codecademy Interactive Lessons as an instructional supplement in an introductory Python programming course. The paper provides a brief literature, about a description of how the author implemented the interactive lessons, a discussion of the positives and negatives, the extent to which the interactive lessons met the course skill outcomes, and conclusions about the overall experience. **NPTEL: A programme for free online and open engineering and science education** Abstract:  The National Programme on Technology Enhanced Learning known as NPTEL is a joint programme executed by several Indian Institutes of Technology and Indian Institute of Science. It was launched in 2003 and has as its objectives the creation of curriculum material in higher education in all of science and engineering. It has created the largest freely accessible; curriculum based video archive in the world in technical education and proposes to create an effective open education forum. This paper traces the programme details and also presents a summary of activities to be carried out in the immediate future.   1. **General Info on Udemy:**   Udemy, Inc. is an American massive open online course provider aimed at professional adults and students.  It was founded in May 2010 by Eren Bali, Gagan Biyani, and Oktay Caglar.  As of Jan 2020, the platform has more than 35 million students and 57,000 instructors teaching courses in over 65 languages.  Many learners have this doubt whether learning from Udemy will be effective and should one Enroll for courses there.  In this Notebook, I have tried to clear few doubt with respect to Udemy, It will give you a bigger picture on Udemy courses  It is found that,  32.6% courses on the Udemy is for WEB DEVELOPMENT  32.5% courses on the Udemy is for BUSINESS FINANCE  18.5% courses on the Udemy is for MUSICAL INSTRUMENTS  16.4% course on the Udemy is for GRAPHIC DESIGN   1. **Kaggle: Data Analysis by Python**   There is high positive relationship between num\_reviews and num\_subscribers. For business insight, it’s highly possible that we can get more subscribers of a new course if we have more customer reviews of that course. However, to know comprehensive factors that affect the number of subscribers, more variables are needed to be examined, such as review scores, course prices and course durations.   1. **Kaggle: Udemy Data: EDA**   INITIAL OBSEVATIONS:   * The price of the courses varied from 0 to 200 dollars * There are courses with zero subscribers * Very few courses have subscribers over 2600 * The max number of reviews are 10 times less than the max subscribers * There are courses with zero hours content   Max content duration is 78.5hours   1. **Kaggle: EDA of Udemy Courses Dataset**   Interpretation:   1. Number of subscribers and is\_paid has some negative correlation 2. Price , number of lectures, Number of reviews , number of lectures, Content duration and number of reviews has positive correlation 3. Number of lectures and content duration has very high positive correlation 4. In course\_id number of subscribers there are seems to be a little correlation. Course\_ids with id around 5000 have more subscribers. 5. Free courses have most number of subscribers 6. Paid courses have more number of reviews and number of lectures. 7. Paid courses have more content duration. Paid courses which cost more than 200 has comparatively more content duration. 8. Most of the courses have around 20k subscribers . 9. There are less number of reviews for most of the courses   Most of the courses have around 125-150 lectures Analyse categorical features  1. There are very few free courses available in Udemy. 2. There are 310 free courses and 3368 paid courses. 3. All levels means course packages that teaches from the beginning to expert level. 1807 out of 1929 of these courses are paid 4. There are very few free courses in Intermediate and Beginner 5. There are no free courses in expert level 6. There are 1099 Business Finance courses and 1067 Web Development courses 7. 91% of Business finance courses are paid and 88% of the web development courses are paid 8. Though very less , there are comparatively more free courses in web development compared to other subjects |
| |  | | --- | |  | |

1. **PROBLEM DEFINITION**

Udemy is an online learning platform which is used by many students and people around the world. They introduce a lot of new courses which are paid and some are unpaid in various subjects and domains. There are many new learners who enroll for the courses in udemy. There will be variations in number of learners, courses, reviews and price. The users sometime enroll for the course but may not be able to complete it. One can get a lot of information regarding how the course should be based on the attributes in the dataset. Using this dataset inferences can be made that can solve the problems in defining and creating a course that will be required by the students. The instructors will be able to create the contents for the course accordingly.

1. **EXPLORATORY DATA ANALYSIS**
   1. **UNDERSTANDING THE VARIABLES**

Data types are important in statistics and it has to be understood to correctly apply the statistical measurements in the data.

Data types in Udemy courses dataset for each column:

* course\_id: numerical data (continuous data).
* course\_title: non-numerical data (string data type).
* url: non-numerical data (string datatype).
* is\_paid: categorical data(nominal data).
* price: numerical data (continuous data).
* num\_subscribers: numerical data (continuous data).
* num\_reviews: numerical data (continuous data).
* num\_lectures: numerical data (continuous data).
* level: categorical data (nominal data).
* content\_duration: numerical data (continuous data).
* published\_timestamp: non-numerical data (date data type).
* subject: non-numerical categorical data (nominal data).
  1. **ANALYSIS OF VARIABLES**
* **Importing necessary libraries for the analysis:**

import numpy as np

import pandas as pd

import seaborn as sns

sns.set(color\_codes=True)

* **Reading the dataset:**

data=pd.read\_csv("udemy\_courses.csv")

data

* **Size of the dataset:**

data.shape

output: (3678, 12)

* **Information:**

data.info()

output:

<class 'pandas.core.frame.DataFrame'>

RangeIndex: 3678 entries, 0 to 3677

Data columns (total 12 columns):

# Column Non-Null Count Dtype

--- ------ -------------- -----

0 course\_id 3678 non-null int64

1 course\_title 3678 non-null object

2 url 3676 non-null object

3 is\_paid 3678 non-null bool

4 price 3677 non-null float64

5 num\_subscribers 3678 non-null int64

6 num\_reviews 3678 non-null int64

7 num\_lectures 3678 non-null int64

8 level 3678 non-null object

9 content\_duration 3678 non-null float64

10 published\_timestamp 3677 non-null object

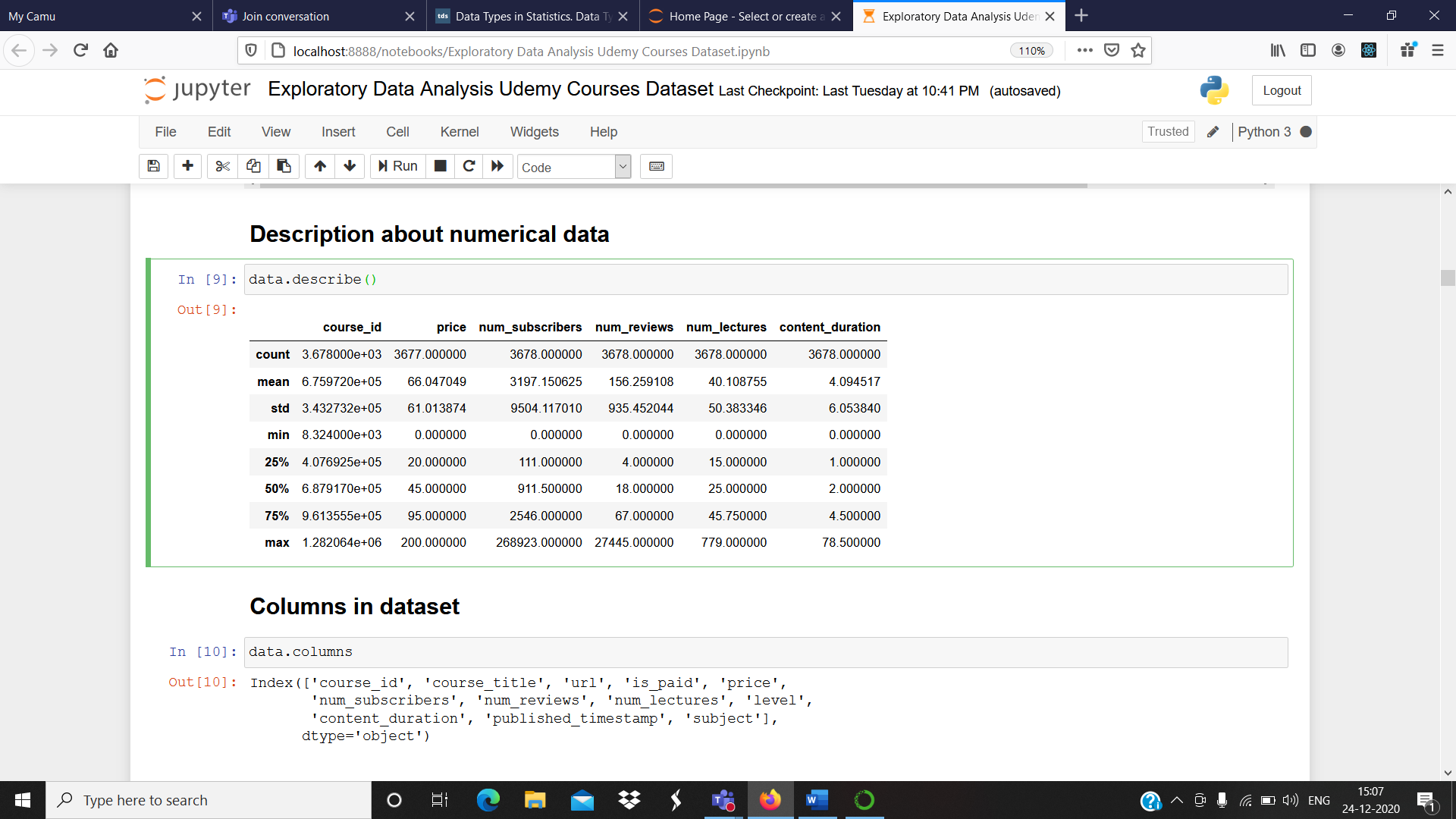
11 subject 3678 non-null object

dtypes: bool(1), float64(2), int64(4), object(5)

memory usage: 319.8+ KB

* **Description about numerical data**:

data.describe()



* **Unique values in the dataset:**

data.nunique()

output:

course\_id 3672

course\_title 3663

url 3670

is\_paid 2

price 38

num\_subscribers 2197

num\_reviews 511

num\_lectures 229

level 4

content\_duration 105

published\_timestamp 3671

subject 4

dtype: int64

* **Columns in the dataset:**

data.columns

output:

Index(['course\_id', 'course\_title', 'url', 'is\_paid', 'price',

'num\_subscribers', 'num\_reviews', 'num\_lectures', 'level',

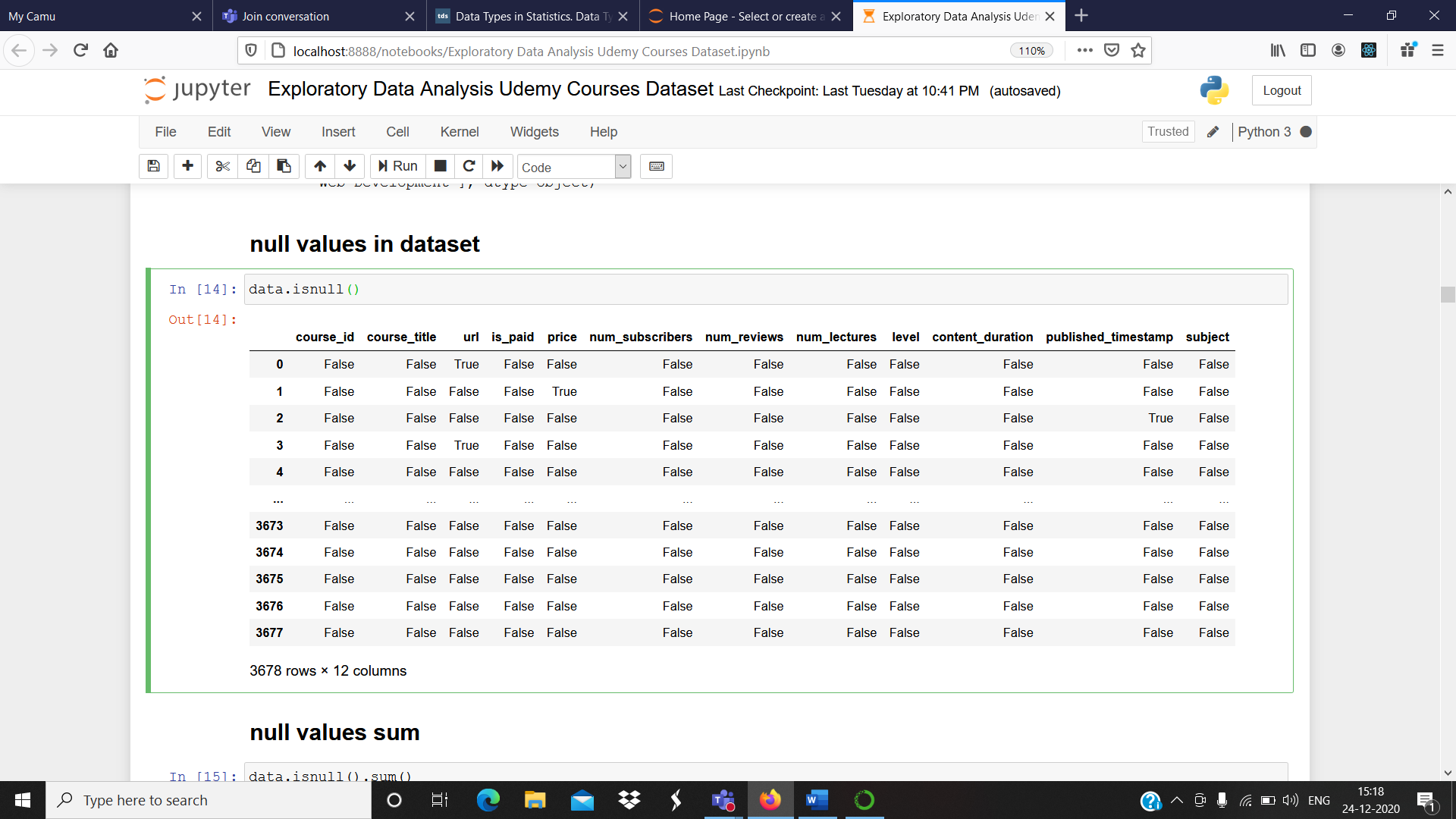
'content\_duration', 'published\_timestamp', 'subject'],

dtype='object')

* 1. **MISSING VALUES TREATMENT**

Missing values in a dataset can lead to a biased model and it can lead to wrong prediction or classification. Hence, missing values are treated. In this dataset missing values have been treated by deleting the rows that has null values in it.

data.isnull()



data.isnull().sum()

output:

course\_id 0

course\_title 0

url 2

is\_paid 0

price 1

num\_subscribers 0

num\_reviews 0

num\_lectures 0

level 0

content\_duration 0

published\_timestamp 1

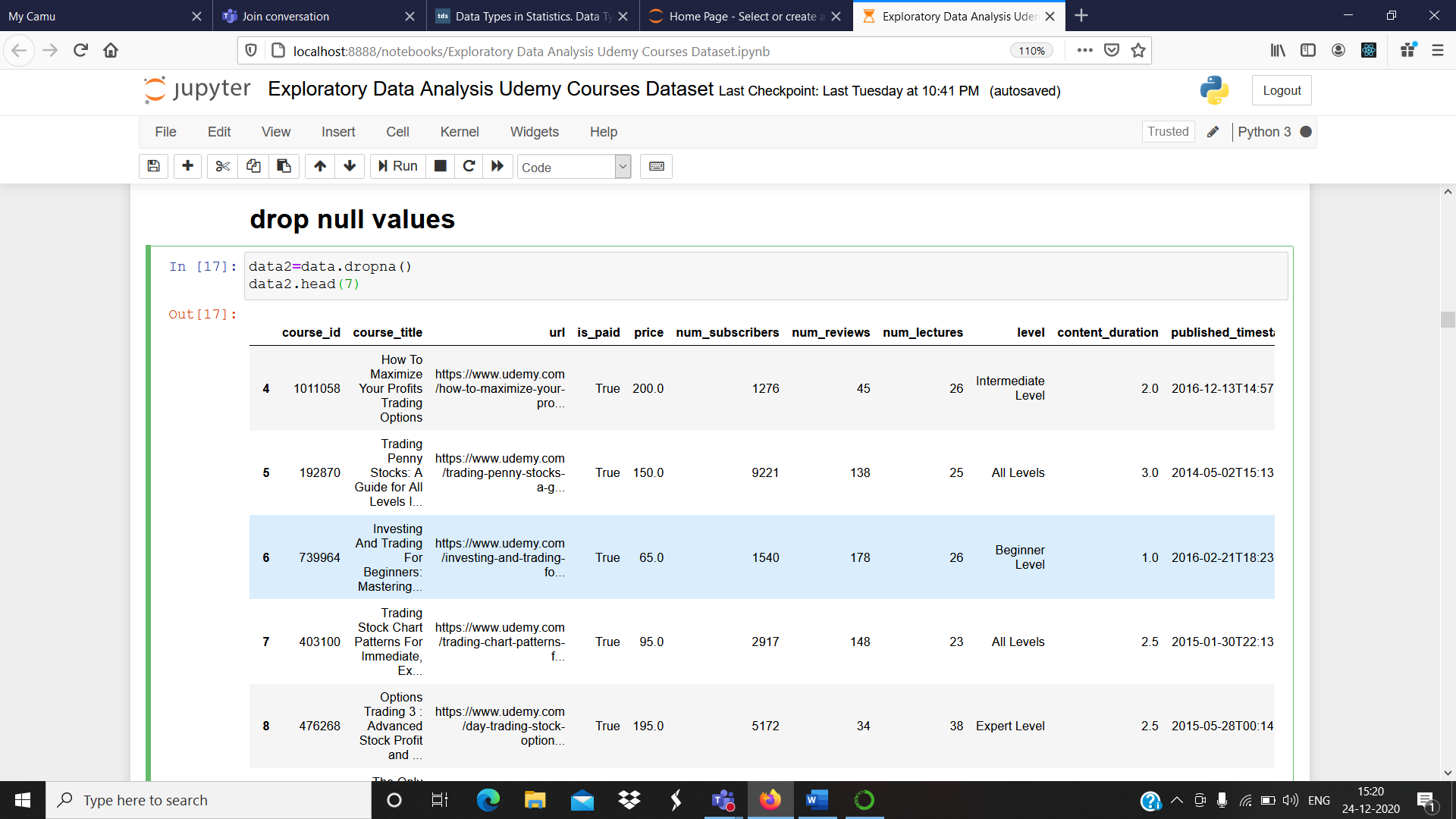
subject 0

dtype: int64

**Missing values treatment:**

data2=data.dropna()

data2.head(7)



Now, the dataset doesn’t contain any null values.

* 1. **OUTLIER TREATMENT**

**Outliers:** An outlier is a data that differs significantly from other observations. Data entry errors, measurement errors, experimental errors, data processing errors, etc. leads to outliers in a dataset.

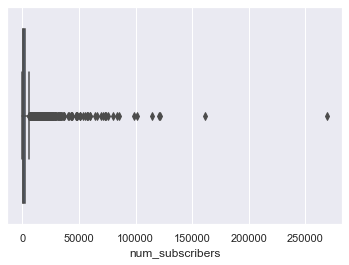
**Outlier treatment:** Outliers can change the results of the data analysis. Visualization methods like box plot, histogram and scatterplot are used to detect outliers.

In this dataset outliers are detected for number of lectures column and have been removed.

**Boxplot:**

* **Number of subscribers:**

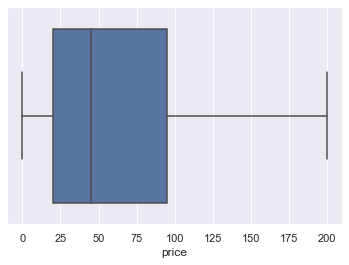
sns.boxplot(x=data2['num\_subscribers'])



From this graph, it can be inferred that number of subscribers are high for certain courses.

* **Price:**

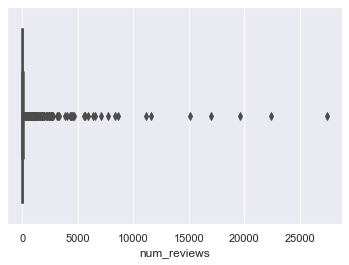
sns.boxplot(x=data2['price'])



From this graph it can be inferred that the maximum amount is 200 and the minimum amount is 0 for the price column and the median is at 45.

* **Number of reviews:**

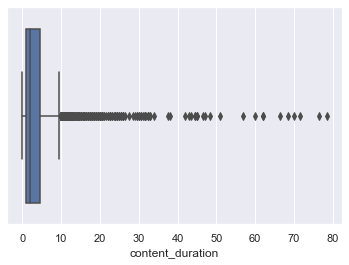
sns.boxplot(x=data2['num\_reviews'])



From this graph, it can be inferred that number reviews for certain courses reaches more than 20000.

* **Content duration:**

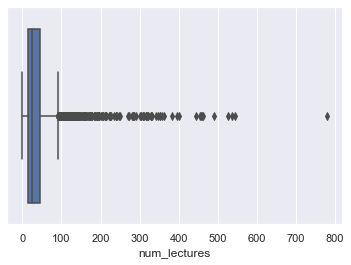
sns.boxplot(x=data2['content\_duration'])



From this graph, it can be inferred that the maximum content duration is about 78.5 hrs.

* **Number of lectures:**

sns.boxplot(x=data2['num\_lectures'])



From this graph, it has been inferred that for some courses number of lectures are more than 100.

data2['num\_lectures'].describe()

output:

count 3674.000000

mean 40.040283

std 50.262096

min 0.000000

25% 15.000000

50% 25.000000

75% 45.000000

max 779.000000

Name: num\_lectures, dtype: float64

* **Calculating interquartile range**

Q1=data2.num\_lectures.quantile(0.25)

Q3=data2.num\_lectures.quantile(0.75)

Q1,Q3

Output: (15.0, 45.0)

IQR=Q3-Q1

IQR

Output: 30.0

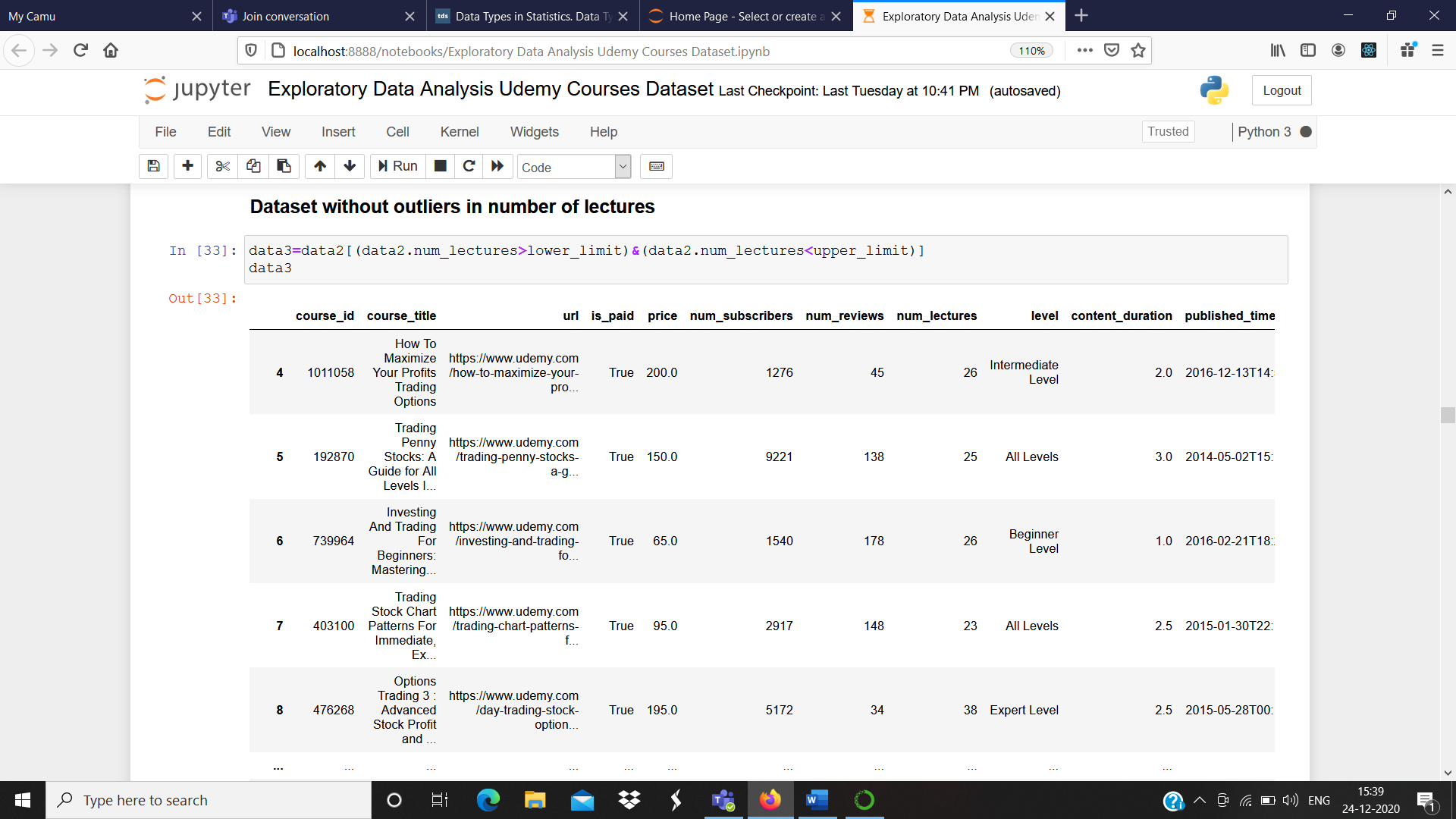
lower\_limit=Q1-1.5\*IQR

upper\_limit=Q3+1.5\*IQR

lower\_limit,upper\_limit

Output: (-30.0, 90.0)

* The outliers are deleted from the column number of lectures.

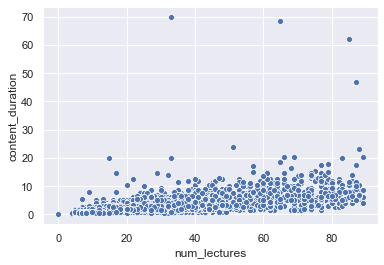


* 1. **DATA VISUALIZATION**

**Scatter plot:**

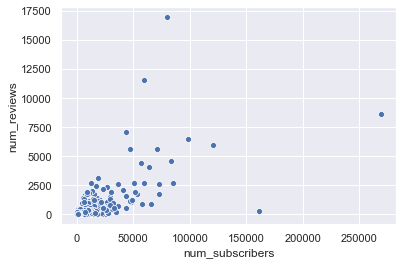
* **Number of lectures and content duration:**

sns.scatterplot(x = data3["num\_lectures"],y=data3["content\_duration"])



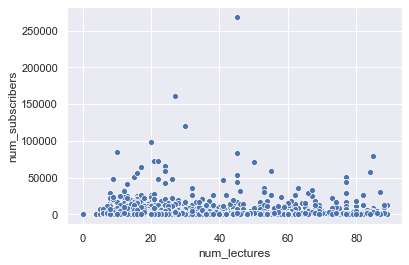
* **Number of subscribers and number of reviews**

sns.scatterplot(x = data3["num\_subscribers"],y=data3["num\_reviews"])



* **Number of reviews and number of subscribers**

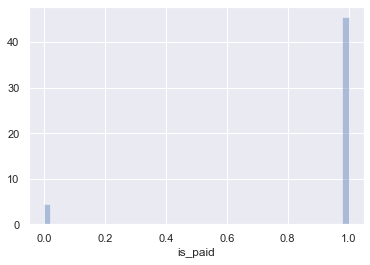
sns.scatterplot(x = data3["num\_lectures"],y=data3["num\_subscribers"])



**Histogram:**

* **Paid and unpaid courses**

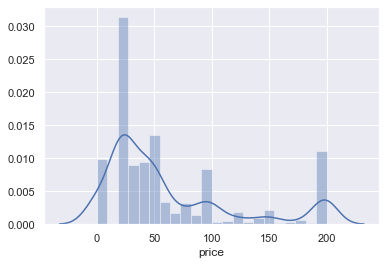
sns.distplot(data3['is\_paid'])



From this graph, it can be inferred that most of the courses are paid courses.

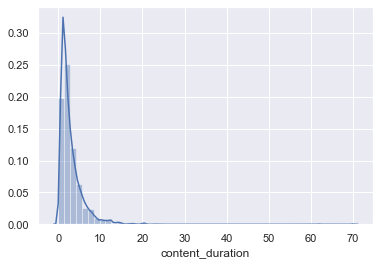
* **Price of courses**

sns.distplot(data3['price'])



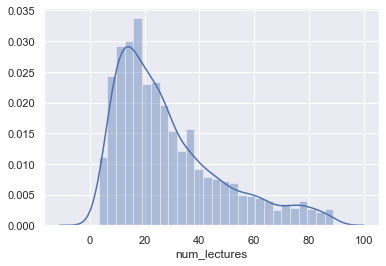
* **Content duration**

sns.distplot(data3['content\_duration'])



* **Number of lectures**

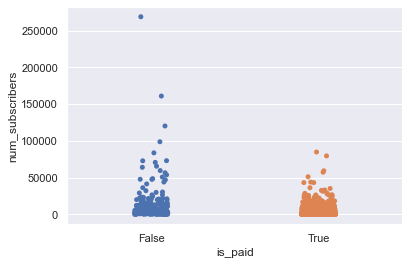
sns.distplot(data3['num\_lectures'])



**Stipplot:**

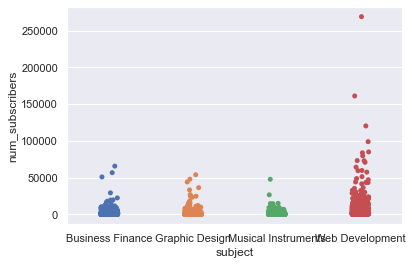
* **Number of subscribers based on whether the course is paid or unpaid**

sns.stripplot(data3['is\_paid'],data3['num\_subscribers'])



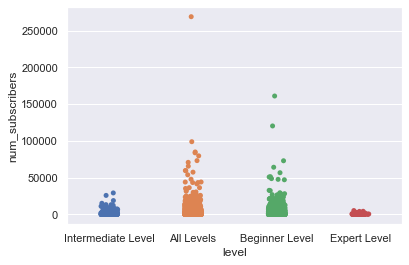
* **Number of subscribers based on subject**

sns.stripplot(data3['subject'],data3['num\_subscribers'])



* **Number of subscribers based on level**

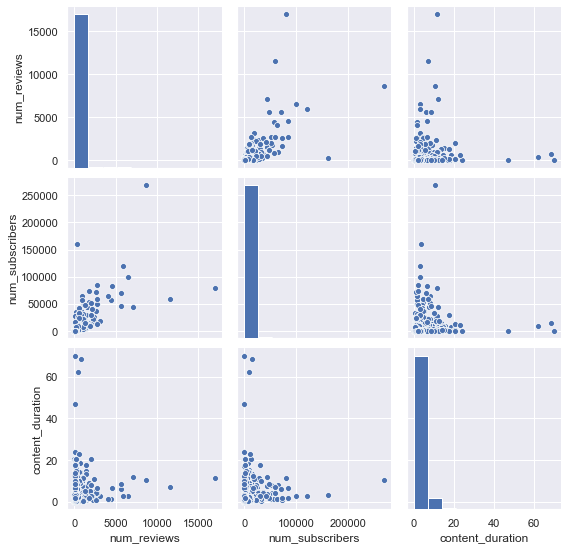
sns.stripplot(data3['level'],data3['num\_subscribers'])



**Pairplot:**

* **Number of reviews, number of subscribers and content duration**

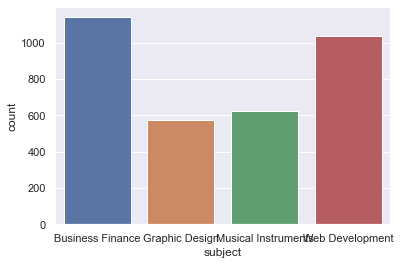
sns.pairplot(data3[['num\_reviews','num\_subscribers','content\_duration']])



**Countplot:**

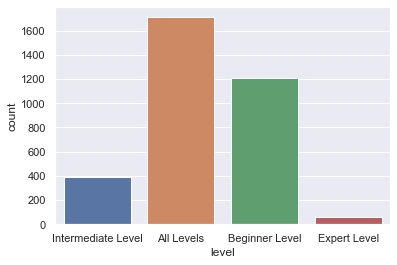
* **Number of courses based on subject**

sns.countplot(data3['subject'])

****

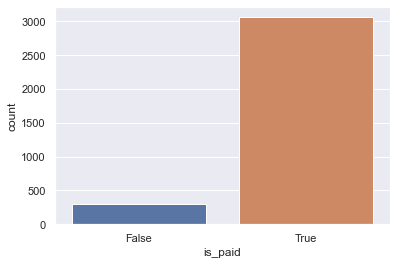
* **Number of courses based on level**

sns.countplot(data3['level'])



* **Number of courses based on whether it is paid or unpaid**

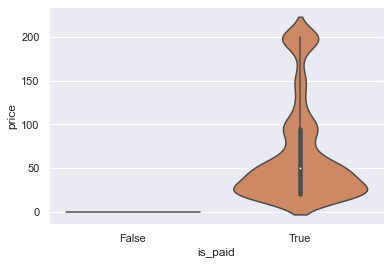
sns.countplot(data3['is\_paid'])



**Violinplot:**

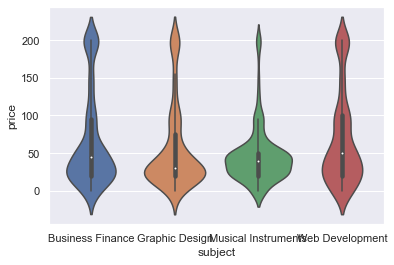
* **Based on whether course is paid or unpaid and price**

sns.violinplot(data3['is\_paid'],data3['price'])



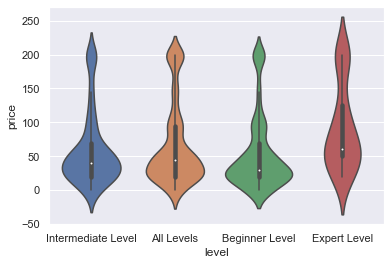
* **Based on subject and price**

sns.violinplot(data3['subject'],data3['price'])



* **Based on level and price**

sns.violinplot(data3['level'],data3['price'])



* 1. **VARIABLE TRANSFORMATION**

**Label encoding:** Label encoding is converting each value in a column to a number. It is an approach to encode categorical variables.

**Label encoding of is\_paid column:**

from sklearn.preprocessing import LabelEncoder

lE=LabelEncoder()

label=lE.fit\_transform(data3["is\_paid"])

data3=data3.drop("is\_paid",axis="columns")

data3["is\_paid"]=label

After label encoding of is\_paid column true value will be replaced with 1 and false value will be replaced with 0.

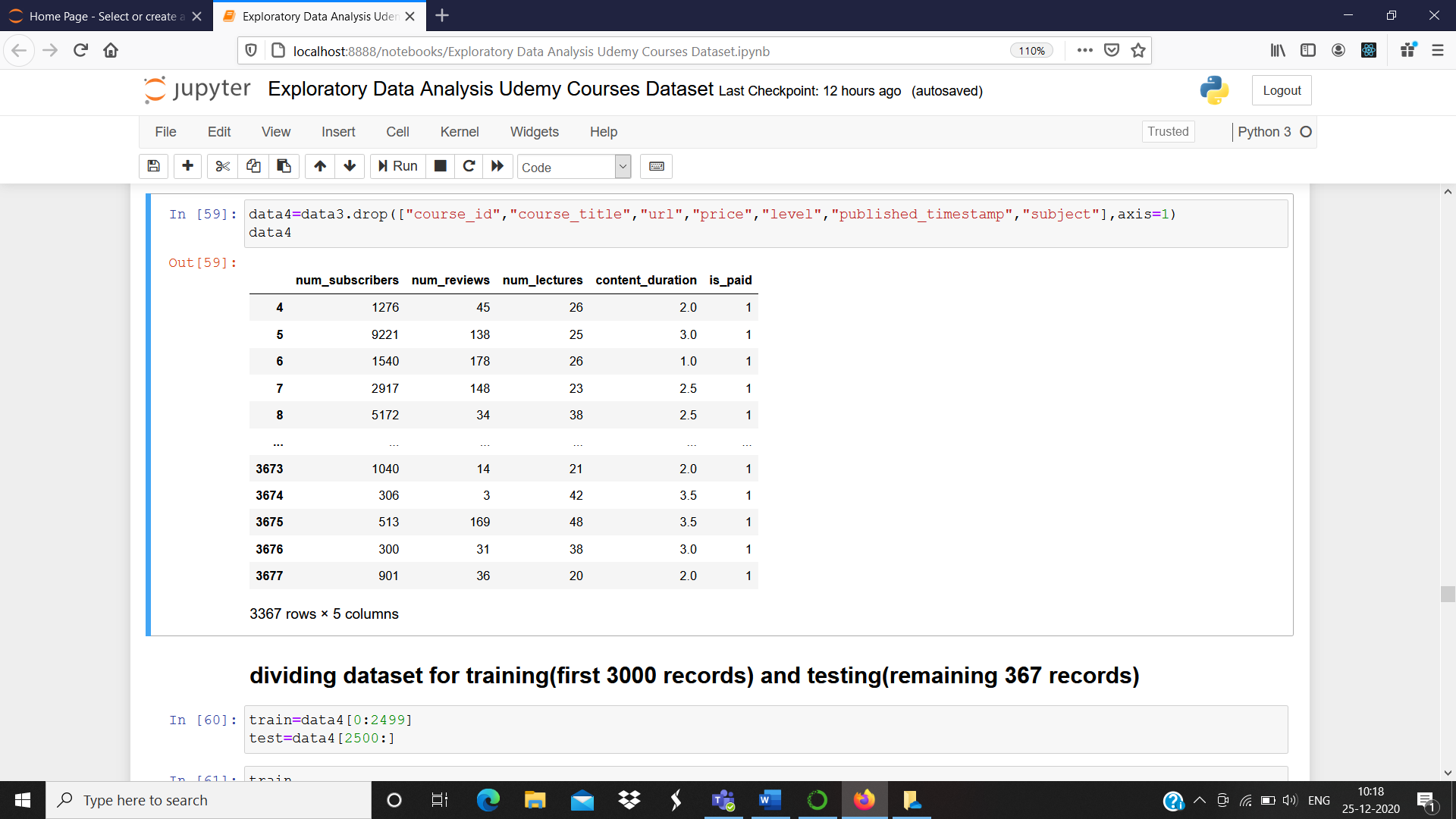
* 1. **PREDICTION OF TARGET VARIABLE – 1**

**Prediction of number of subscribers based on number of reviews, number of lectures, content duration and whether the course is aid or unpaid.**

* **Taking only the required columns from the dataset.**

data4=data3.drop(["course\_id","course\_title","url","price","level","published\_timestamp","subject"],axis=1)

data4



* **Dataset is now divided into training and testing data. The machine learning model is trained using Linear Regression and prediction is done on test data.**

train=data4[0:2499]

test=data4[2500:]

x\_train=train.drop("num\_subscribers",axis=1)

y\_train=train["num\_subscribers"]

x\_test=test.drop("num\_subscribers",axis=1)

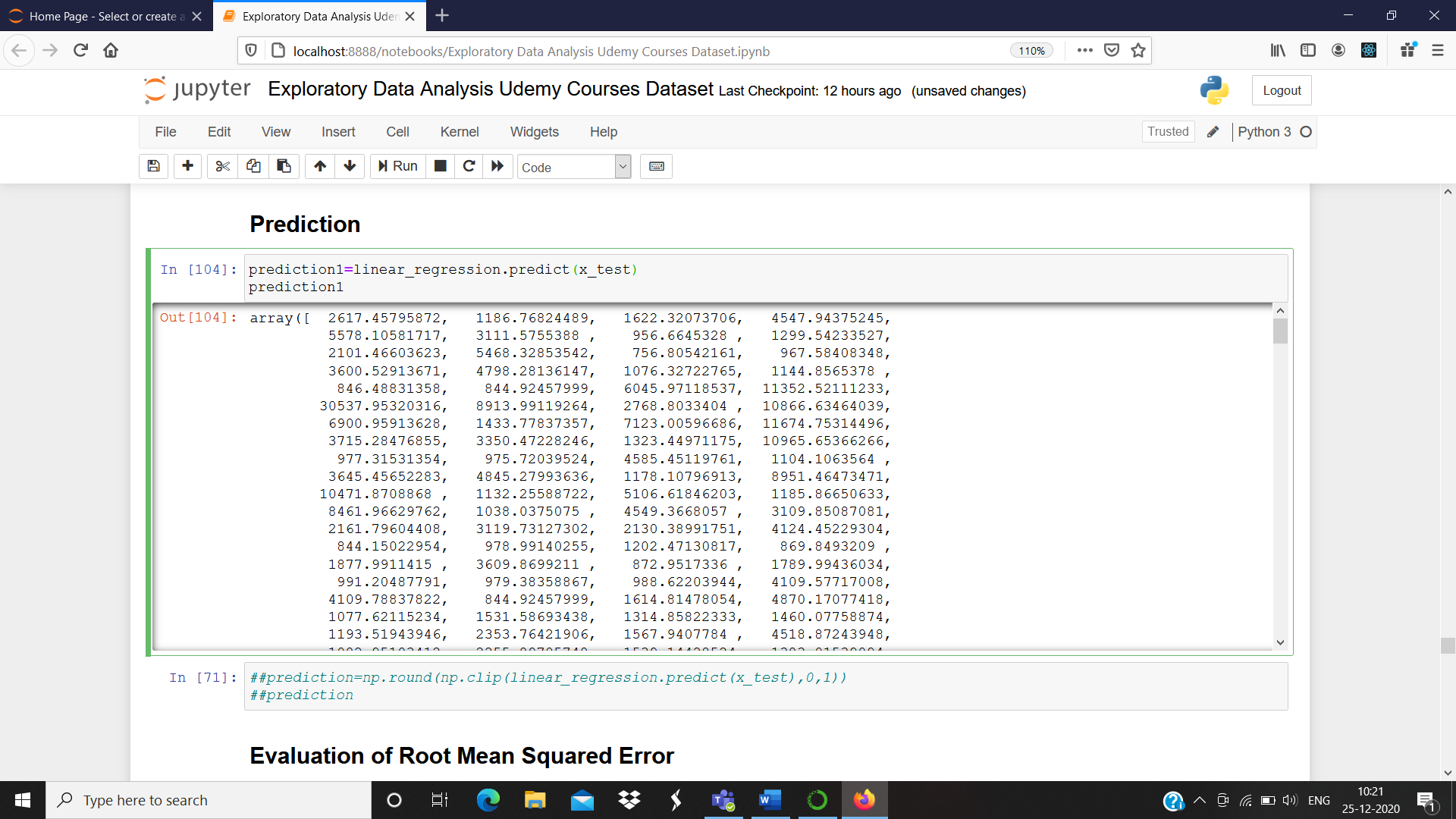
true\_data=test["num\_subscribers"]

from sklearn.linear\_model import LinearRegression

linear\_regression=LinearRegression()

linear\_regression.fit(x\_train,y\_train)

prediction1=linear\_regression.predict(x\_test)



**Evaluation of score, MSE, RMSE and R2 value**

linear\_regression.score(x\_test,true\_data)

**output:** 0.27789568455157976

linear\_regression.score(x\_train,y\_train)

**output:** 0.6119777558125968

from sklearn.metrics import mean\_squared\_error

mse\_test = mean\_squared\_error(true\_data, prediction1)

rmse\_test = np.sqrt(mse\_test)

rmse\_test

**output:** 11947.502597790983

mse\_train = mean\_squared\_error(y\_train, linear\_regression.predict(x\_train))

rmse\_train = np.sqrt(mse\_train)

rmse\_train

**output:** 3448.6335174651185

**R-Squared value:** R-Squared can take any value between 0 and 1. It shows how well the data fit the regression model.

from sklearn.metrics import r2\_score

r2=r2\_score(true\_data,prediction1)

r2

**output:** 0.27789568455157976

From this value, a conclusion can be drawn that only 28% of the data fit the regression model.

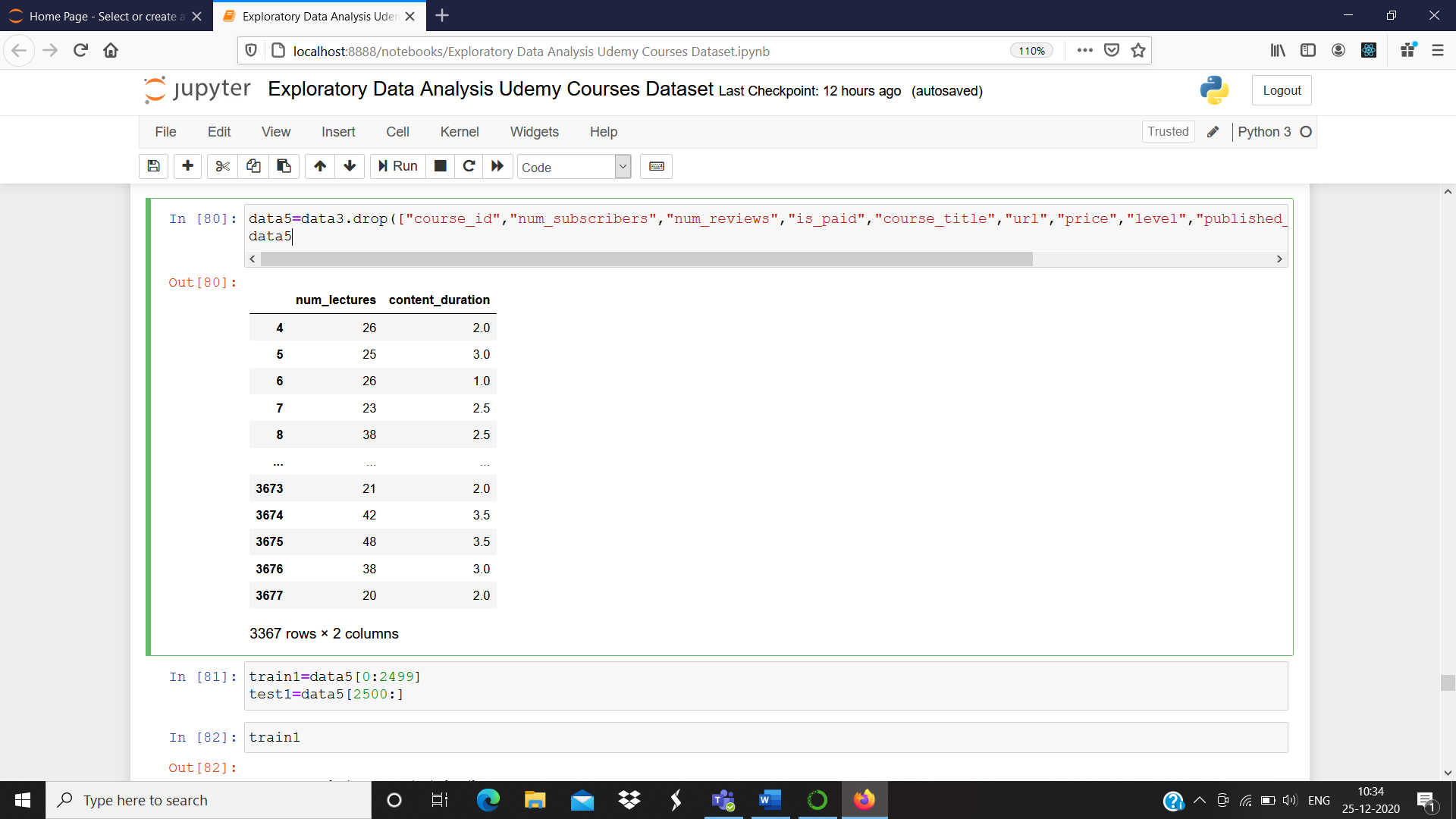
* 1. **PREDICTION OF TARGET VARIABLE – 2**

**Prediction of content duration based on number of lectures.**

* **Taking only the required column from the dataset.**

data5=data3.drop(["course\_id","num\_subscribers","num\_reviews","is\_paid","course\_title","url","price","level","published\_timestamp","subject"],axis=1)

data5



* **Dataset is now divided into training and testing data. The machine learning model is trained using Linear Regression and prediction is done on test data.**

train1=data5[0:2499]

test1=data5[2500:]

x\_train1=train1.drop("content\_duration",axis=1)

y\_train1=train1["content\_duration"]

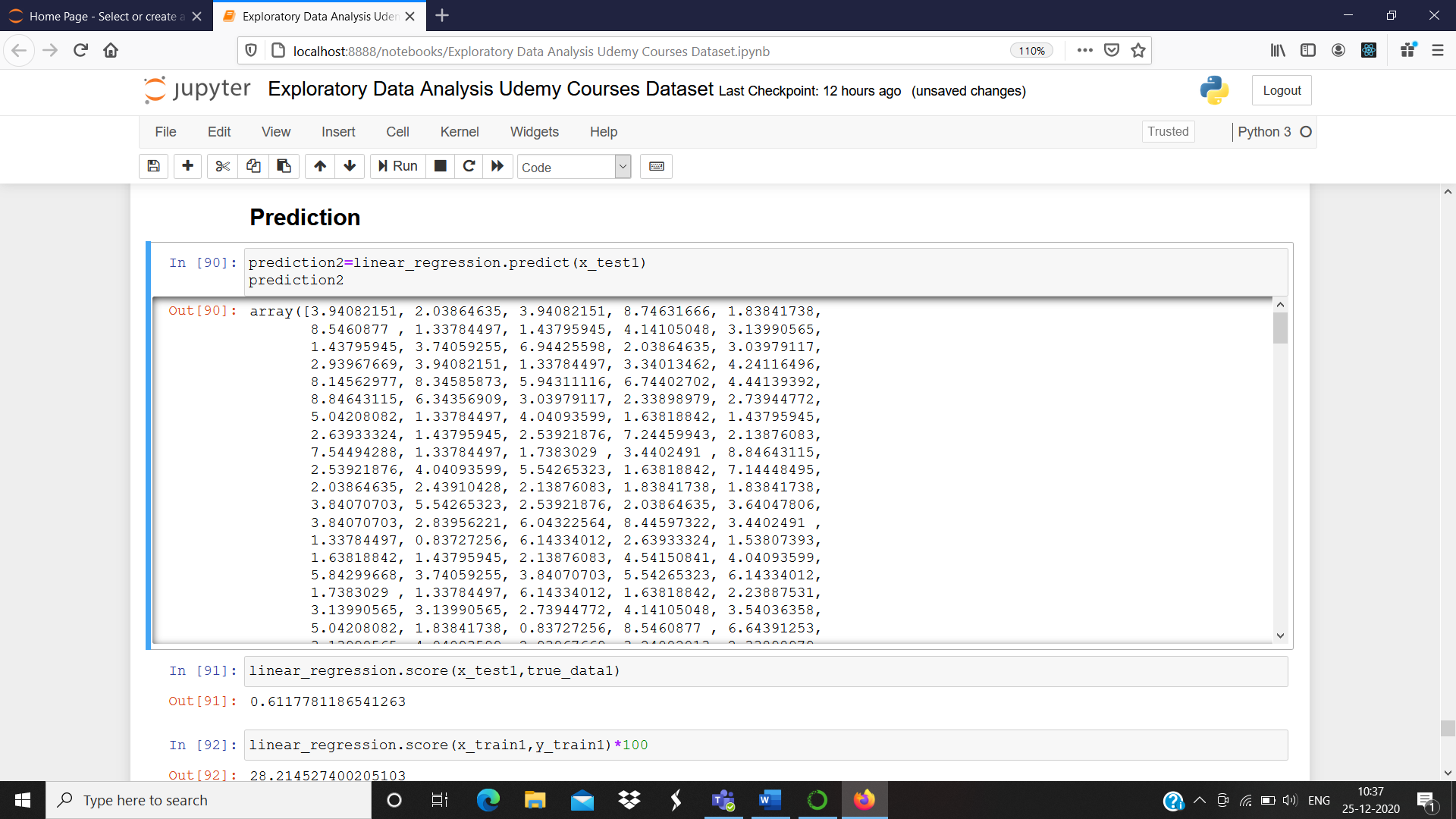
x\_test1=test1.drop("content\_duration",axis=1)

true\_data1=test1["content\_duration"]

a=linear\_regression.fit(x\_train1,y\_train1)

prediction2=linear\_regression.predict(x\_test1)

prediction2



**Evaluation of score, MSE, RMSE and R2 value:**

linear\_regression.score(x\_test1,true\_data1)

**output:** 0.6117781186541263

linear\_regression.score(x\_train1,y\_train1)

**output:** 0.282145274002051

mse\_test1 = mean\_squared\_error(true\_data1, prediction2)

rmse\_test1 = np.sqrt(mse\_test1)

rmse\_test1

**output:** 1.7720077776352403

mse\_train1 = mean\_squared\_error(y\_train1, linear\_regression.predict(x\_train1))

rmse\_train1 = np.sqrt(mse\_train1)

rmse\_train1

**output:** 2.9984482392725633

r2=r2\_score(true\_data1,prediction2)

r2

**output:** 0.6117781186541263

From this value, a conclusion can be drawn that about 61% of the data fit the regression model.

**Prediction by giving input:**

num\_lectures=int(input("Enter number of lectures: "))

predictions=a.predict([[num\_lectures]])

print("Predicted number of lectures: ",predictions[0])

**output:** Enter number of lectures: 23

Predicted number of lectures: 2.4391042758986865

1. **CONCLUSION**

This project thus focuses on inferring information that helps in providing valuable information to the instructors about the number of students taking up different courses based on various criteria, so that they will be able to prepare the content for the courses in such a way that it benefits the students and helps them to learn and gain knowledge on various fields. From the above made predictions, some valuable data has been inferred like predicting the content duration based on number of lectures and number of subscribers are predicted based on various other criteria.

1. **REFERENCES**
   1. <http://www.wikicfp.com/cfp/call?conference=e-learning>
   2. <http://www.wikicfp.com/cfp/servlet/event.showcfp?eventid=112893&copyownerid=158787>
   3. <https://www.researchgate.net/publication/328842461_Using_Codecademy_interactive_lessons_as_an_instructional_supplement_in_a_Python_programming_course>
   4. <https://ieeexplore.ieee.org/abstract/document/5314112>
   5. Kaggle