The dataset contains information about Udemy courses including the subject being taught, the price of the course , is the course paid or not ,the level of the course(beginner,medium,expert)etc

//importing libraries

import numpy as np

import pandas as pd

import matplotlib.pyplot as plt

import seaborn as sns

//reading dataset which i downloaded from kaggle

df1=pd.read\_csv('./Udemy courses/1.csv')

print(df1.head())

course\_id ... subject

0 1070968 ... Business Finance

1 1113822 ... Business Finance

2 1006314 ... Business Finance

3 1210588 ... Business Finance

4 1011058 ... Business Finance

//removing course\_id column as it not of any use to us

df1.drop(['course\_id'],axis=1,inplace=True)

print(df1.head())

course\_title url is\_paid ... content\_duration published\_timestamp subject

0 Ultimate Investment Banking Course https://www.udemy.com/ultimate-investment-bank... True ... 1.5 2017-01-18T20:58:58Z Business Finance

1 Complete GST Course & Certification - Grow You... https://www.udemy.com/goods-and-services-tax/ True ... 39.0 2017-03-09T16:34:20Z Business Finance

2 Financial Modeling for Business Analysts and C... https://www.udemy.com/financial-modeling-for-b... True ... 2.5 2016-12-19T19:26:30Z Business Finance

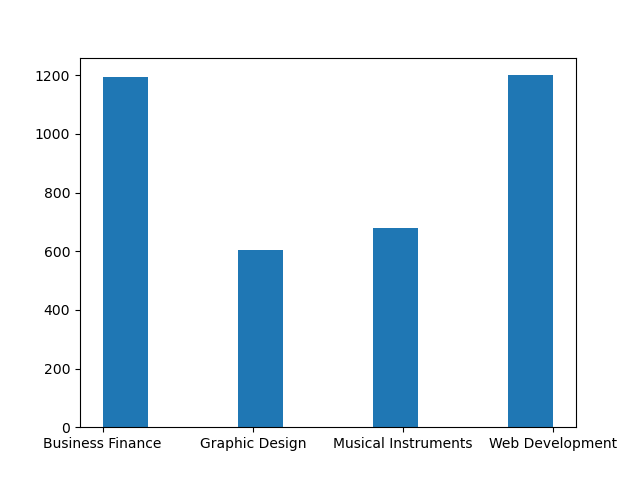
3 Beginner to Pro - Financial Analysis in Excel ... https://www.udemy.com/complete-excel-finance-c... True ... 3.0 2017-05-30T20:07:24Z Business Finance

4 How To Maximize Your Profits Trading Options https://www.udemy.com/how-to-maximize-your-pro... True ... 2.0 2016-12-13T14:57:18Z Business Finance

//count the numbers of course according to subject

plt.hist(df1['subject'])

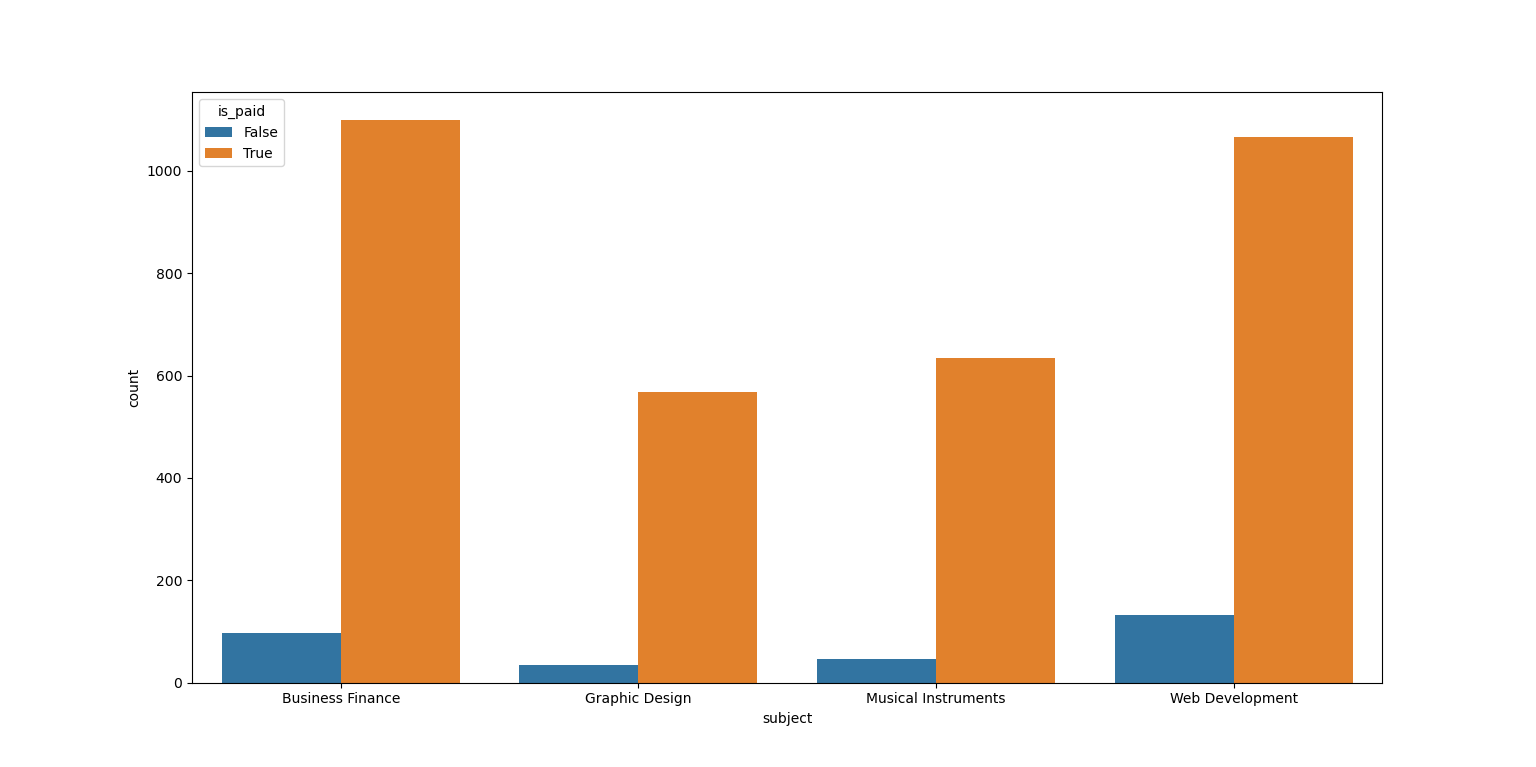
plt.show()



//counting numbers of course according to sunject and discriminating whether the course is paid or free

sns.countplot(x='subject',data=df1,hue='is\_paid')

plt.show()



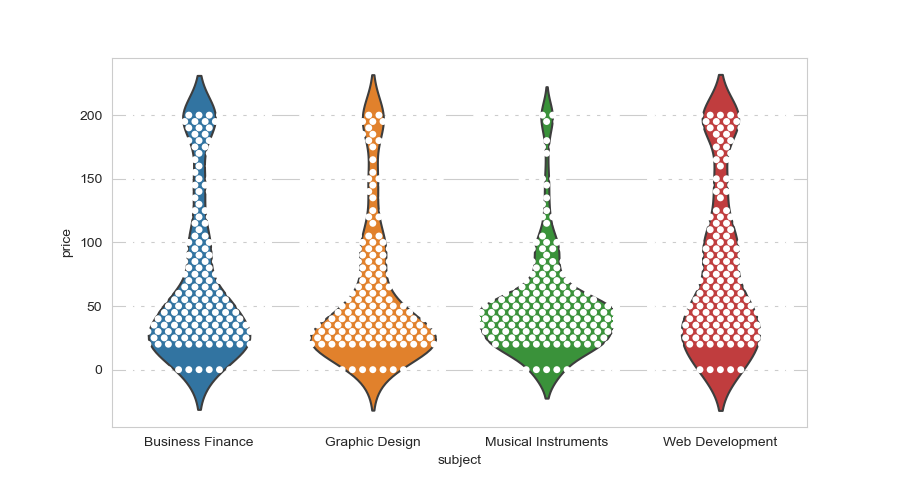
//setting style for grid and plottinf mix of violentplot and swarmplot describing subject vs price

sns.set\_style('whitegrid')

sns.violinplot(x='subject',y='price',data=df1,inner=None)

sns.swarmplot(x='subject',y='price',data=df1 ,color="white", edgecolor="gray")

plt.plot()

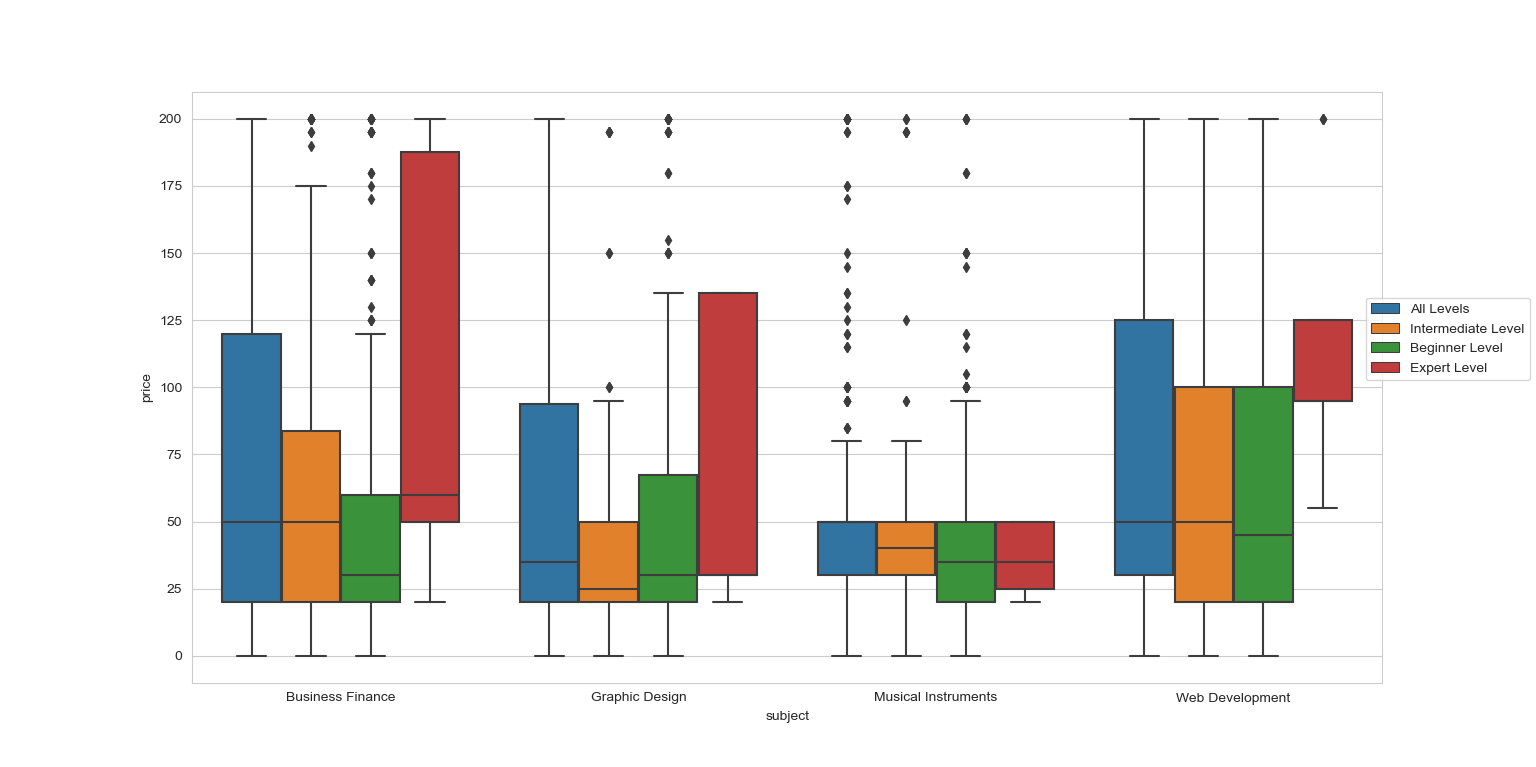


//making boxplot for better understanding of subject vs price and differentating on the basis of level

sns.boxplot(x='subject',y='price',data=df1,hue='level')

plt.legend(bbox\_to\_anchor=(0.98, 0.5, .35, .102), loc=3)

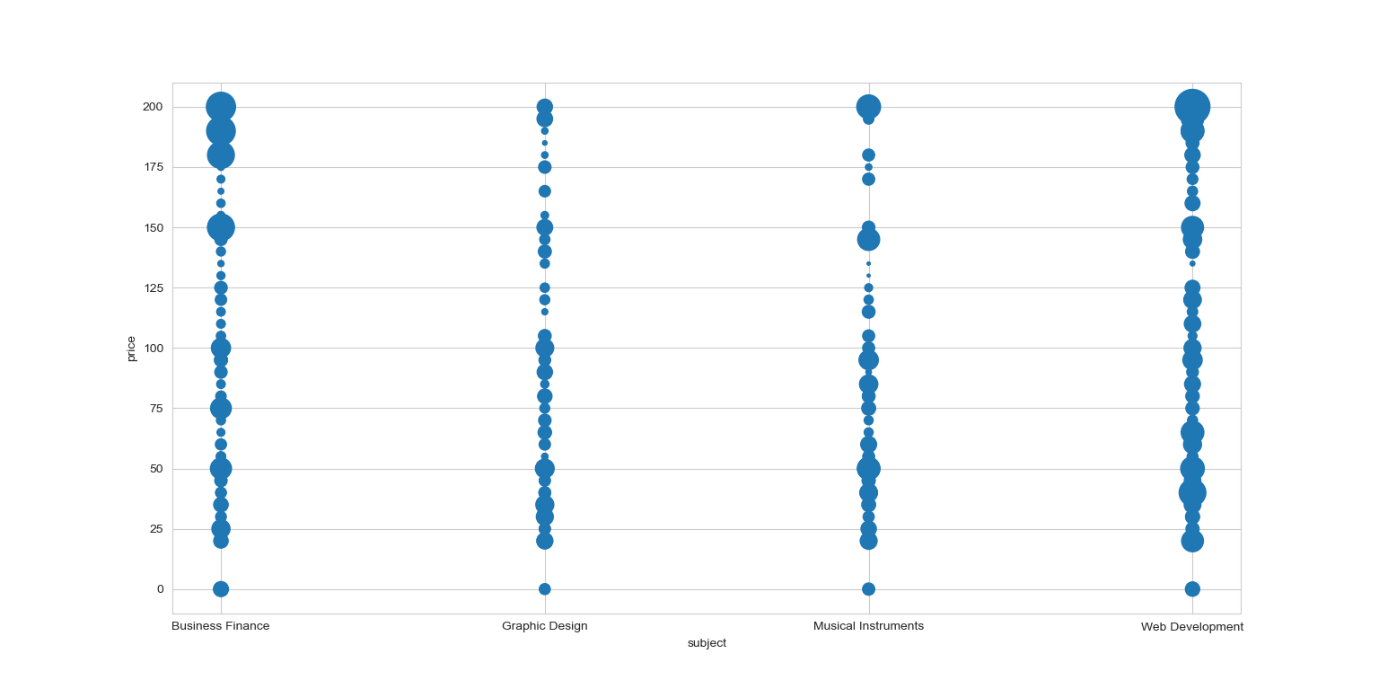
plt.show()



//scatter plot to have a better understanding of subject vs price where the dot size describe the numbers of lecture in that course

df1.plot.scatter(x='subject',y='price',s=df1['num\_lectures'])

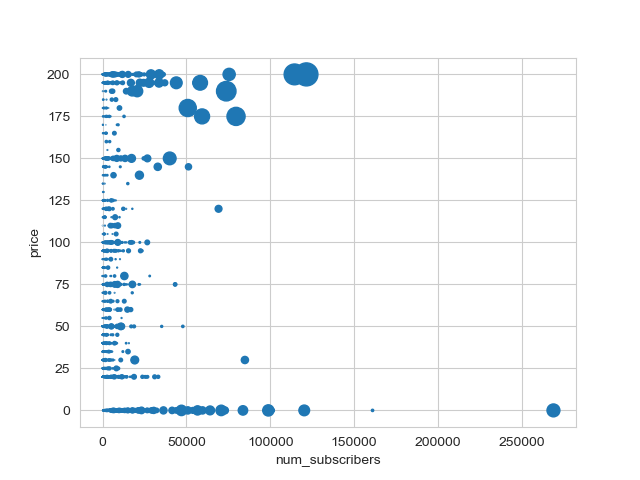
plt.show()



//scatter plot of num\_subscribers vs price where dot size denotes the nums of revies on that course

df1.plot.scatter(x='num\_subscribers',y='price',s=df1['num\_reviews']\*(1/100))

plt.show()



//To find the maximum price for each subject

print(df1.groupby(['subject']).max()['price'])

subject

Business Finance 200

Graphic Design 200

Musical Instruments 200

Web Development 200

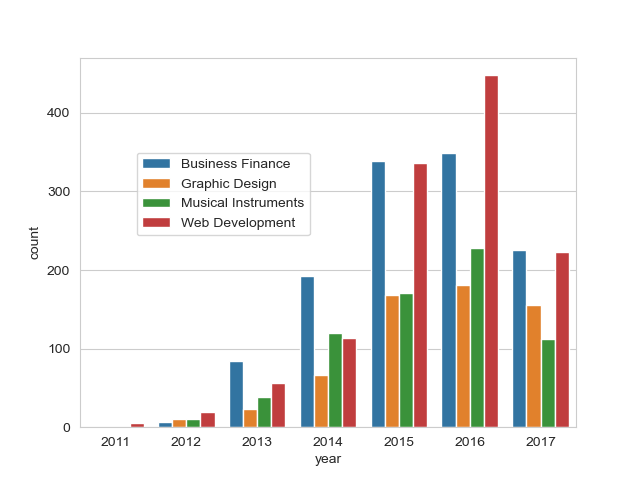
//To find the popularity for each subject courses in each years

df1['year']=pd.DatetimeIndex(df1['published\_timestamp']).year

sns.countplot(x='year',data=df1,hue='subject')

plt.legend(bbox\_to\_anchor=(0.10, 0.5, .35, .102), loc=3)

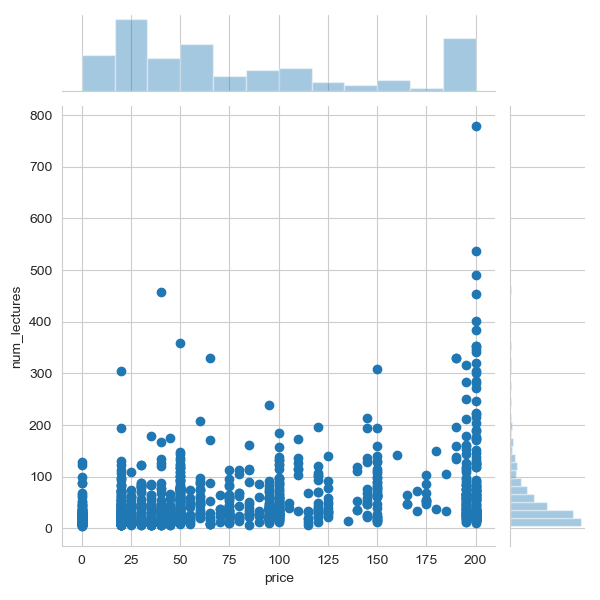
plt.show()



//to find the price vs numbers of lecture for particular subject like web development

df2=df1[df1['subject']=='Web Development']

sns.jointplot(x='price',y='num\_lectures',data=df2)

plt.show()