





SUBJECT- PYTHON WITH AI

**SUBMITTED BY- TUSHER MANDAL**

**SUBMITTED TO- ANKIT PRAMANIK**

**SUBMITTED ON- 22th JULY, 2019**

STUDENT PROFILE



**NAME- TUSHER MANDAL**

**COLLEGE- HOOGHLY ENGINEERING AND**

**TECHNOLOGY COLLEGE (HETC)**

**COURSE - B.TECH(COMPUTER SCIENCE**

**AND ENGINEERING)**

**SEM- 5TH SEM, 3RD YEAR**

**YEAR OF PASSING- 2021**

**PHONE NO- 9163950742**

**EMAIL-** [**tuhinmandal2003@gmail.com**](mailto:tuhinmandal2003@gmail.com)

CONTENTS

* ***Acknowledgement***
* ***Introduction***
* ***Objectives***
* ***Hardware and Software Requirements***
* ***Future Scope***
* ***Advantages***
* ***Snapshots***
* ***Coding***
* ***Conclusion***
* ***Bibliography***

ACKNOWLEDGEMENT

*I have taken efforts in this project. However, it would not have been possible without the kind support and help of many individuals and organizations. I would like to extend my sincere thanks to all of them.*

*I am highly indebted to National Institute for Industrial Training for their guidance and constant supervision as well as for providing necessary information regarding the project & also for their support in completing the project. I would like to express my gratitude towards Mr. ANKIT PRAMANIK and Mr. SAYANTAN CHAKRABORTY encouragement which helped me in completion of this project.*

INTRODUCTION

*Python comes with a huge amount of inbuilt libraries. Many of the libraries are for Artificial Intelligence and Machine Learning. Some of the libraries are Tensor flow (which is high-level neural network library), scikit-learn (for data mining, data analysis and machine learning), pylearn2 (more flexible than scikit-learn), etc.*

*Python has an easy implementation for Open CV. What makes Python favourite for everyone is its easy implementation. For other languages, students and researchers need to get to know the language before getting into ML or AI with that language. This is not the case with python. Even a programmer with very basic knowledge can easily handle python. Apart from that, the time someone spends on writing and debugging code in python is way less when compared to C, C++ or Java. This is exactly what the students of AI want. They don't want to spend time on debugging the code for syntax errors, they want to spend more time on their algorithms and heuristics related to AI. Not just the libraries but their tutorials, handling of interfaces are easily available online. People build their own libraries and upload them on Git Hub or elsewhere to be used by others.*

OBJECTIVES

*Python is a powerful open source programming language, which means that it’s free to use while having all the properties that a programming language should have.*

* *It is a versatile programming language that supports Object-Oriented Programming, Structured Programming, and functional programming patterns.*
* *Python sports an easy to understand and readable syntax that ensures that the development time is cut into half when compared with other programming languages.*
* *Python enables you to perform data analysis, data manipulation, and data visualization, which are very important in data science.*

Hardware and Software Requirements

***Software Requirements :-***

***Operating System : Windows/Linux***

***Front End : Python 3.7***

***Platform : Anaconda***

***Hardware Requirements*:-**

***Machine : HP Probook X360***

***Speed : 233MHz and above***

***Hard disk : 10GB***

***RAM : 8 GB***

Future Scope

*➢ Python programming language is undoubtedly dominating the other languages when future technologies like Artificial Intelligence(AI) comes into the play.*

*➢ There are plenty of python frameworks, libraries, and tools that are specifically developed to direct Artificial Intelligence to reduce human efforts with increased accuracy and efficiency for various development purposes.*

*➢ It is only the Artificial Intelligence that has made it possible to develop speech recognition system, autonomous cars, interpreting data like images, videos etc.*

ADVANTAGES

*1) Python can be used in the development of prototypes, and it can help speed up the concept to creation process because it is so easy to use and read.*

*2) Python is ideal for general purpose tasks such as data mining, and big data facilitation.*

THEORY

***Python****is an [interpreted](https://en.wikipedia.org/wiki/Interpreted_language" \o "Interpreted language),*[*high-level*](https://en.wikipedia.org/wiki/High-level_programming_language)*,*[*general-purpose*](https://en.wikipedia.org/wiki/General-purpose_programming_language)[*programming language*](https://en.wikipedia.org/wiki/Programming_language)*. Created by*[*Guido van Rossum*](https://en.wikipedia.org/wiki/Guido_van_Rossum)*and first released in 1991, Python's design philosophy emphasizes code readability with its notable use of*[*significant whitespace*](https://en.wikipedia.org/wiki/Off-side_rule)*. Its language constructs and*[*object-oriented*](https://en.wikipedia.org/wiki/Object-oriented_programming)*approach aim to help programmers write clear, logical code for small and large-scale projects.*

*Python is*[*dynamically typed*](https://en.wikipedia.org/wiki/Dynamic_programming_language)*and*[*garbage-collected*](https://en.wikipedia.org/wiki/Garbage_collection_(computer_science))*. It supports multiple*[*programming paradigms*](https://en.wikipedia.org/wiki/Programming_paradigm)*, including*[*procedural*](https://en.wikipedia.org/wiki/Procedural_programming)*, object-oriented, and*[*functional programming*](https://en.wikipedia.org/wiki/Functional_programming)*. Python is often described as a "batteries included" language due to its comprehensive*[*standard library*](https://en.wikipedia.org/wiki/Standard_library)*.*

***Data science****is a*[*multi-disciplinary*](https://en.wikipedia.org/wiki/Multi-disciplinary)*field that uses scientific methods, processes, algorithms and systems to extract*[*knowledge*](https://en.wikipedia.org/wiki/Knowledge)*and insights from structured and unstructured*[*data*](https://en.wikipedia.org/wiki/Data)*. Data science is the same concept as*[*data mining*](https://en.wikipedia.org/wiki/Data_mining)*and*[*big data*](https://en.wikipedia.org/wiki/Big_data)*:"use the most powerful hardware, the most powerful programming systems, and the most efficient algorithms to solve problems".*

*Data science is a "concept to unify statistics, data analysis, machine learning and their related methods" in order to "understand and analyze actual phenomena" with data.It employs techniques and theories drawn from many fields within the context of [mathematics](https://en.wikipedia.org/wiki/Mathematics" \o "Mathematics),*[*statistics*](https://en.wikipedia.org/wiki/Statistics)*,*[*computer science*](https://en.wikipedia.org/wiki/Computer_science)*, and*[*information science*](https://en.wikipedia.org/wiki/Information_science)*.* [*Turing award*](https://en.wikipedia.org/wiki/Turing_award)*winner*[*Jim Gray*](https://en.wikipedia.org/wiki/Jim_Gray_(computer_scientist))*imagined data science as a "fourth paradigm" of science ([empirical](https://en.wikipedia.org/wiki/Empirical_research" \o "Empirical research),*[*theoretical*](https://en.wikipedia.org/wiki/Basic_research)*,*[*computational*](https://en.wikipedia.org/wiki/Computational_science)*and now data-driven) and asserted that"everything about science is changing because of the impact of information technology"and the*[*data deluge*](https://en.wikipedia.org/wiki/Information_explosion)*.In 2015, the*[*American Statistical Association*](https://en.wikipedia.org/wiki/American_Statistical_Association)*identified*[*database management*](https://en.wikipedia.org/wiki/Database)*,statistics and*[*machine learning*](https://en.wikipedia.org/wiki/Machine_learning)*,and*[*distributed and parallel systems*](https://en.wikipedia.org/wiki/Distributed_computing)*as the three emerging foundational professional communities.*

*In*[*computer science*](https://en.wikipedia.org/wiki/Computer_science)*, A****rtificial Intelligence****(****AI****), sometimes called****machine intelligence****,is*[*intelligence*](https://en.wikipedia.org/wiki/Intelligence)*demonstrated by [machines](https://en.wikipedia.org/wiki/Machine" \o "Machine),in contrast to the****natural intelligence*** *displayed by humans. Colloquially,the term"artificial intelligence"is often used to describe machines (or computers)that mimic "cognitive" functions that humans associate with the*[*human mind*](https://en.wikipedia.org/wiki/Human_mind)*,such as "learning" and "problem solving".*

***Some Important Features of Python:***

***What is NumPy?*** *NumPy is a general-purpose array-processing package. It provides a high-performance multidimensional array object, and tools for working with these arrays.*

*It is the fundamental package for scientific computing with Python. It contains various features including these important ones:*

* *A powerful N-dimensional array object*
* *Sophisticated (broadcasting) functions*
* *Tools for integrating C/C++ and Fortran code*
* *Useful linear algebra, Fourier transform, and random number capabilities*

***Matplotlib:***

***Matplotlib****is a*[*plotting*](https://en.wikipedia.org/wiki/Plotter)[*library*](https://en.wikipedia.org/wiki/Library_(computer_science))*for the*[*Python*](https://en.wikipedia.org/wiki/Python_(programming_language))*programming language and its numerical mathematics extension [NumPy](https://en.wikipedia.org/wiki/NumPy" \o "NumPy). It provides an*[*object-oriented*](https://en.wikipedia.org/wiki/Object-oriented_programming)[*API*](https://en.wikipedia.org/wiki/API)*for embedding*

*plots into applications using general-purpose*[*GUI toolkits*](https://en.wikipedia.org/wiki/GUI_toolkit)*like [Tkinter](https://en.wikipedia.org/wiki/Tkinter" \o "Tkinter), [wxPython](https://en.wikipedia.org/wiki/WxPython" \o "WxPython),*[*Qt*](https://en.wikipedia.org/wiki/Qt_(software))*, or*[*GTK+*](https://en.wikipedia.org/wiki/GTK%2B)*. There is also a*[*procedural*](https://en.wikipedia.org/wiki/Procedural_programming)*"pylab" interface based on a*[*state machine*](https://en.wikipedia.org/wiki/State_machine)*(like*[*OpenGL*](https://en.wikipedia.org/wiki/OpenGL)*), designed to closely resemble that of*[*MATLAB*](https://en.wikipedia.org/wiki/MATLAB)*, though its use is discouraged.* [*SciPy*](https://en.wikipedia.org/wiki/SciPy)*makes use of Matplotlib.*

***Seaborn: Python's Statistical Data Visualization Library***

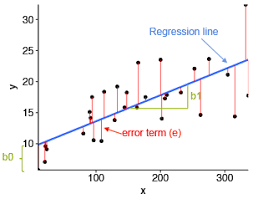
*One of the best but also more challenging ways to get your insights across is to visualize them: that way, you can more easily identify patterns, grasp difficult concepts or draw the attention to key elements. When you’re using Python for data science, you’ll most probably will have already used [Matplotlib](https://matplotlib.org/), a 2D plotting library that allows you to create publication. *

***Logistic Regression:***

***Logistic regression****is a statistical****model****that in its basic form uses a****logistic****function to****model****a binary dependent variable, although many more complex extensions exist. In****regression****analysis,****logistic regression****(or****logit regression****) is estimating the parameters of a****logistic model****(a form of binary* ***regression****).*

***Linear Regression:***

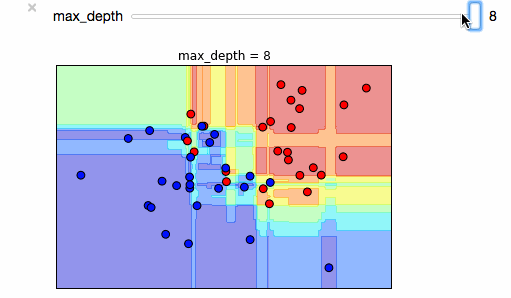
***Linear regression****is a****linear****approach to modeling the relationship between a scalar response (or dependent variable) and one or more explanatory variables.*

******

***Random Forest Classifier:***

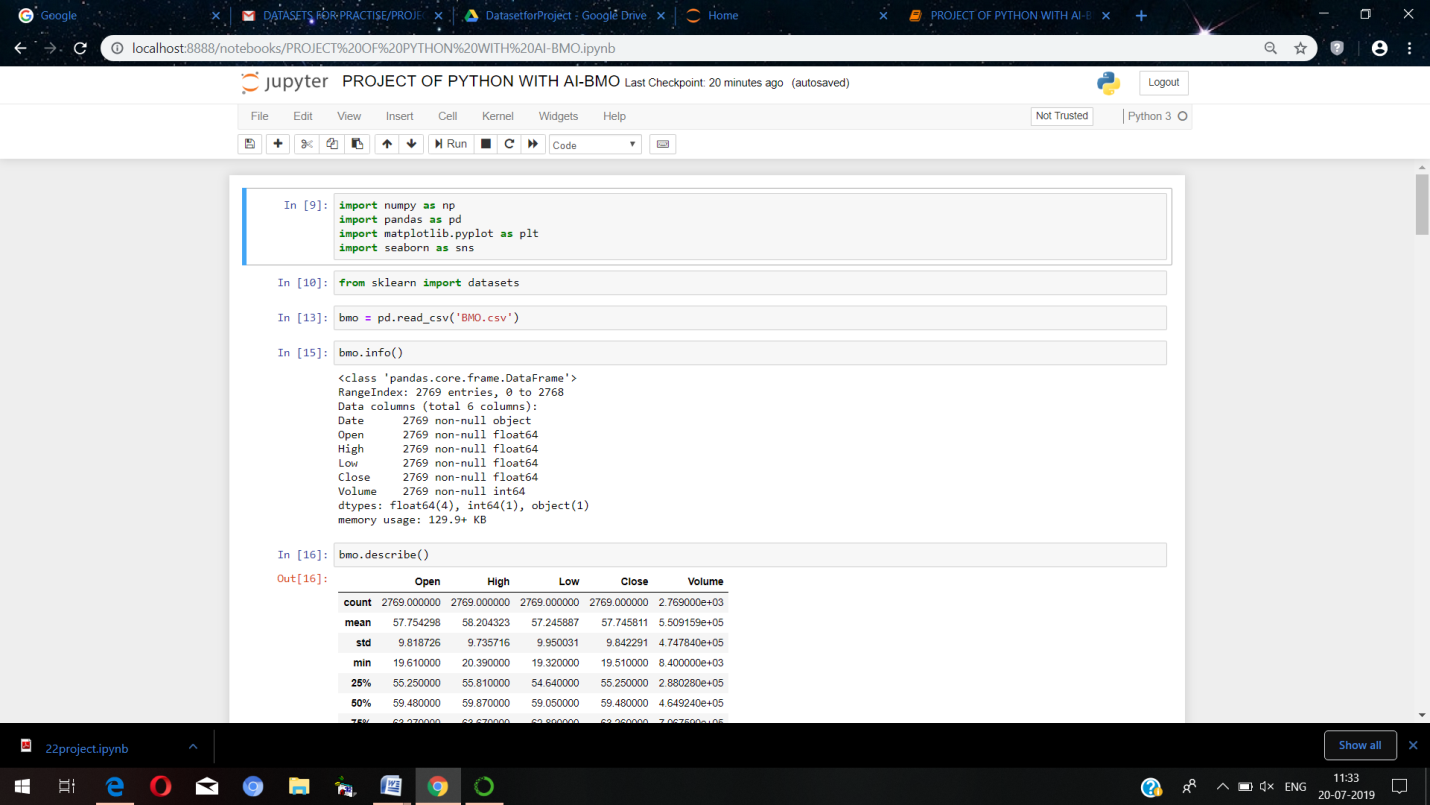
*Random forests is a supervised learning algorithm. It can be used both for classification*

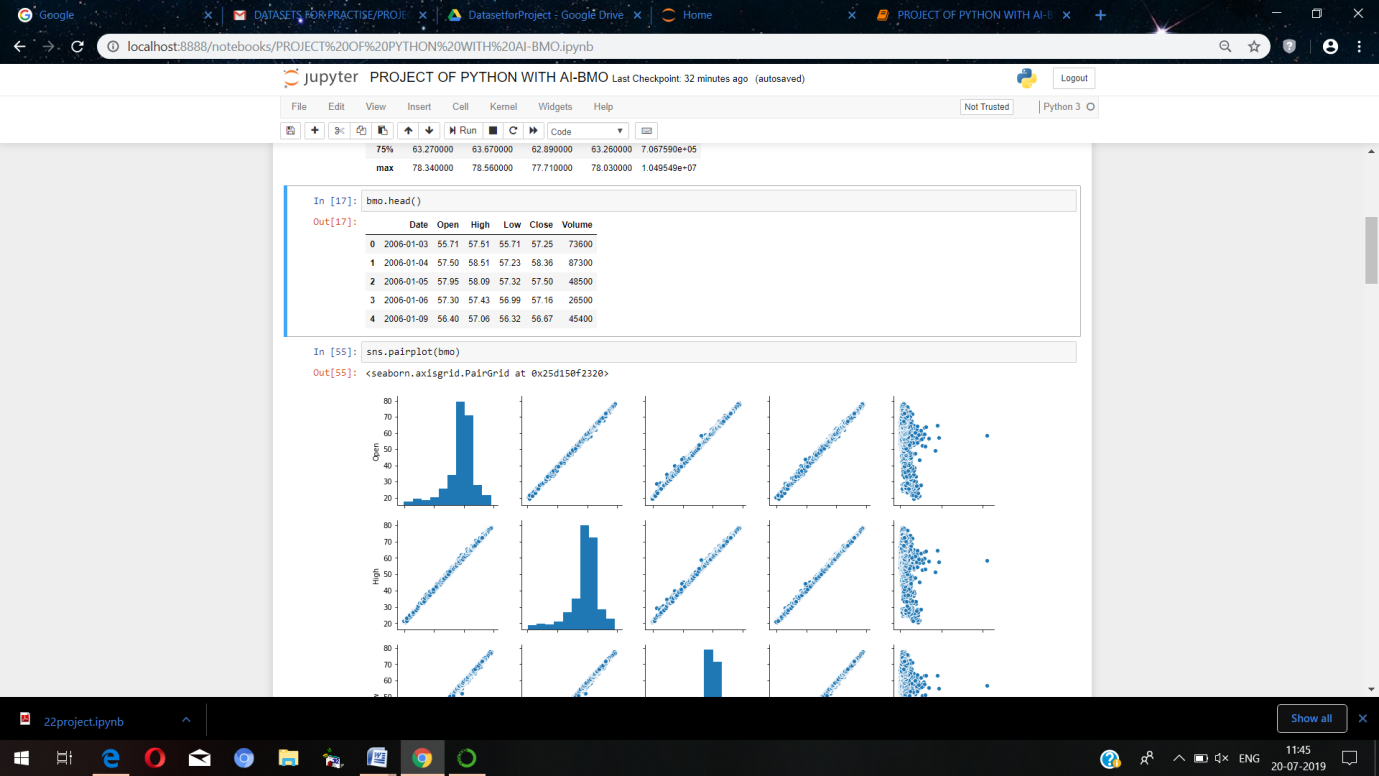
*and regression. It is also the most flexible and easy to use algorithm. A forest is comprised of trees. It is said that the more trees it has, the more robust a forest is. Random forests creates decision trees on randomly selected data samples, gets prediction from each tree and selects the best solution by means of voting. It also provides a pretty good indicator of the feature importance.*

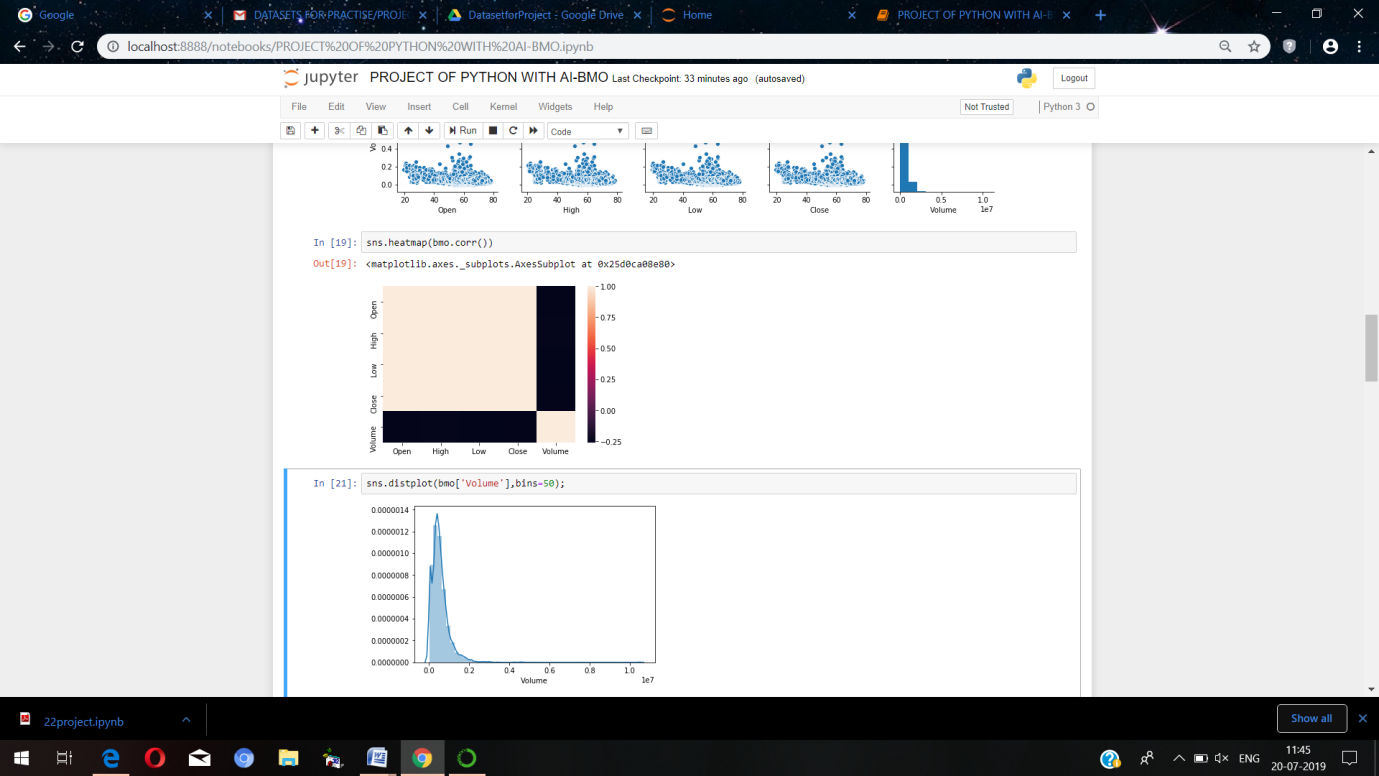
******

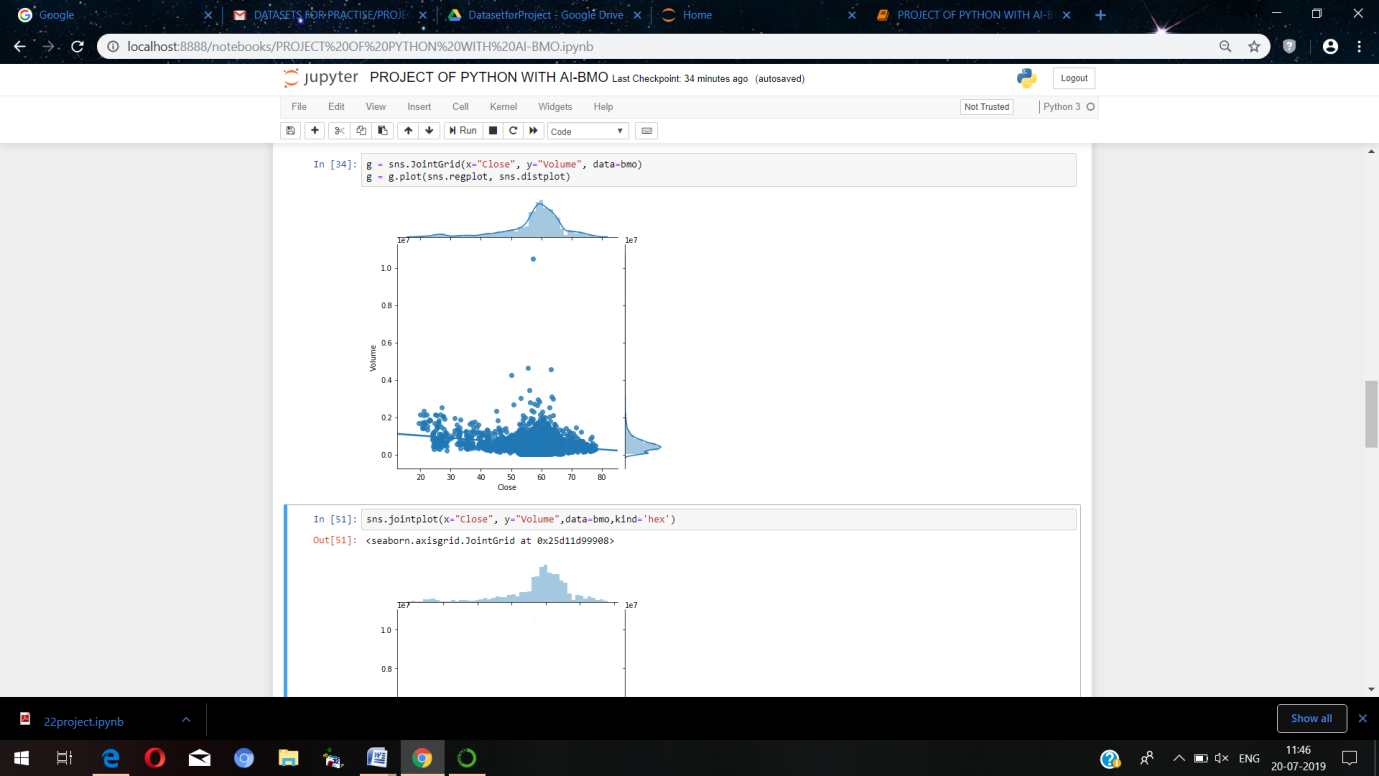
*Fig:Visualising Random Forest Classifier*

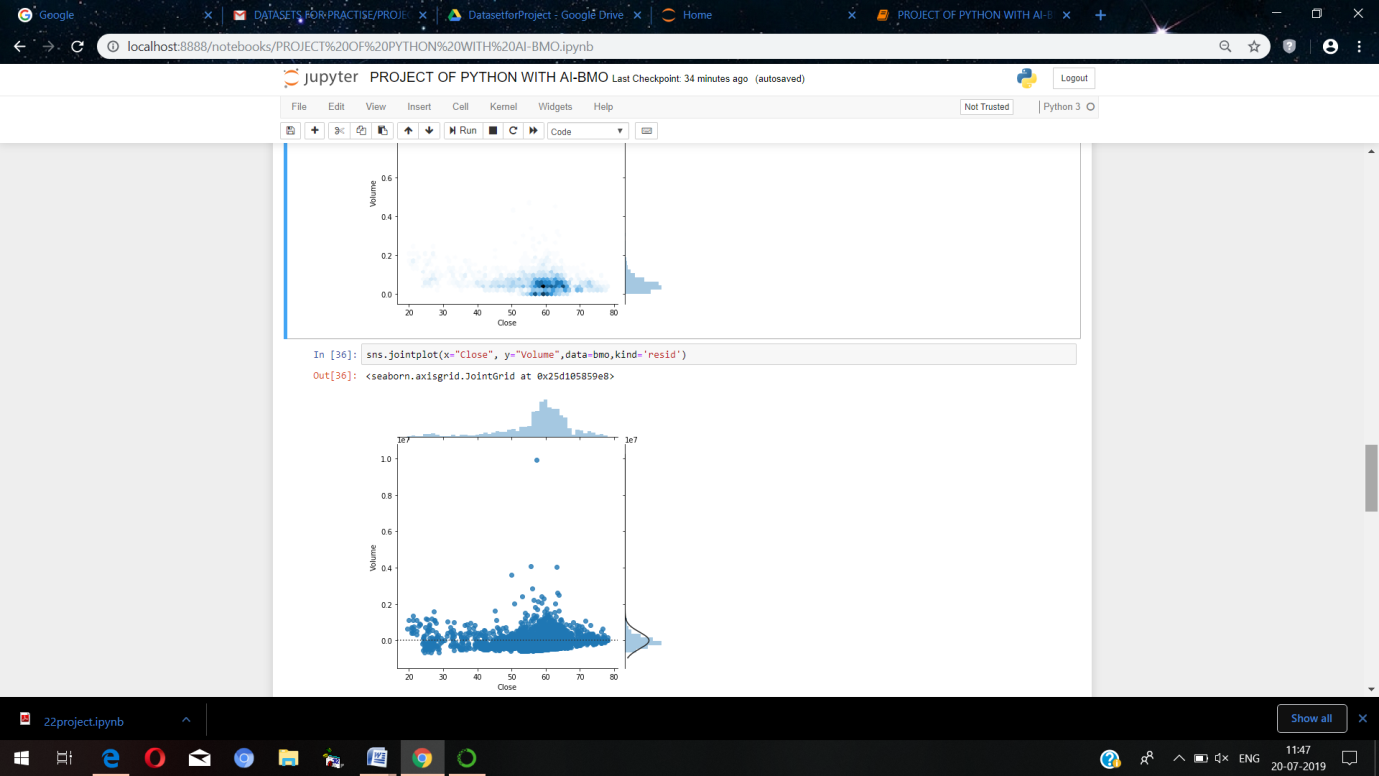
SNAPSHOTS

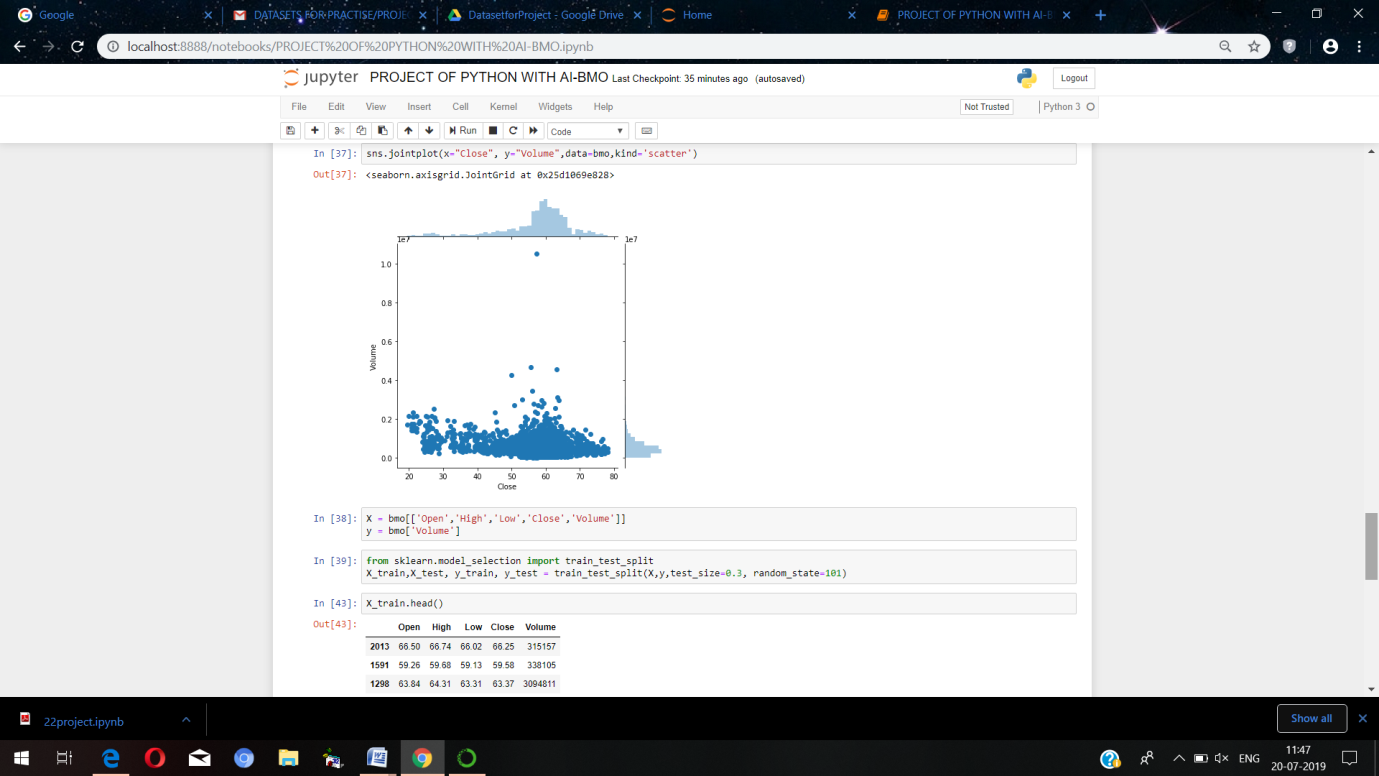


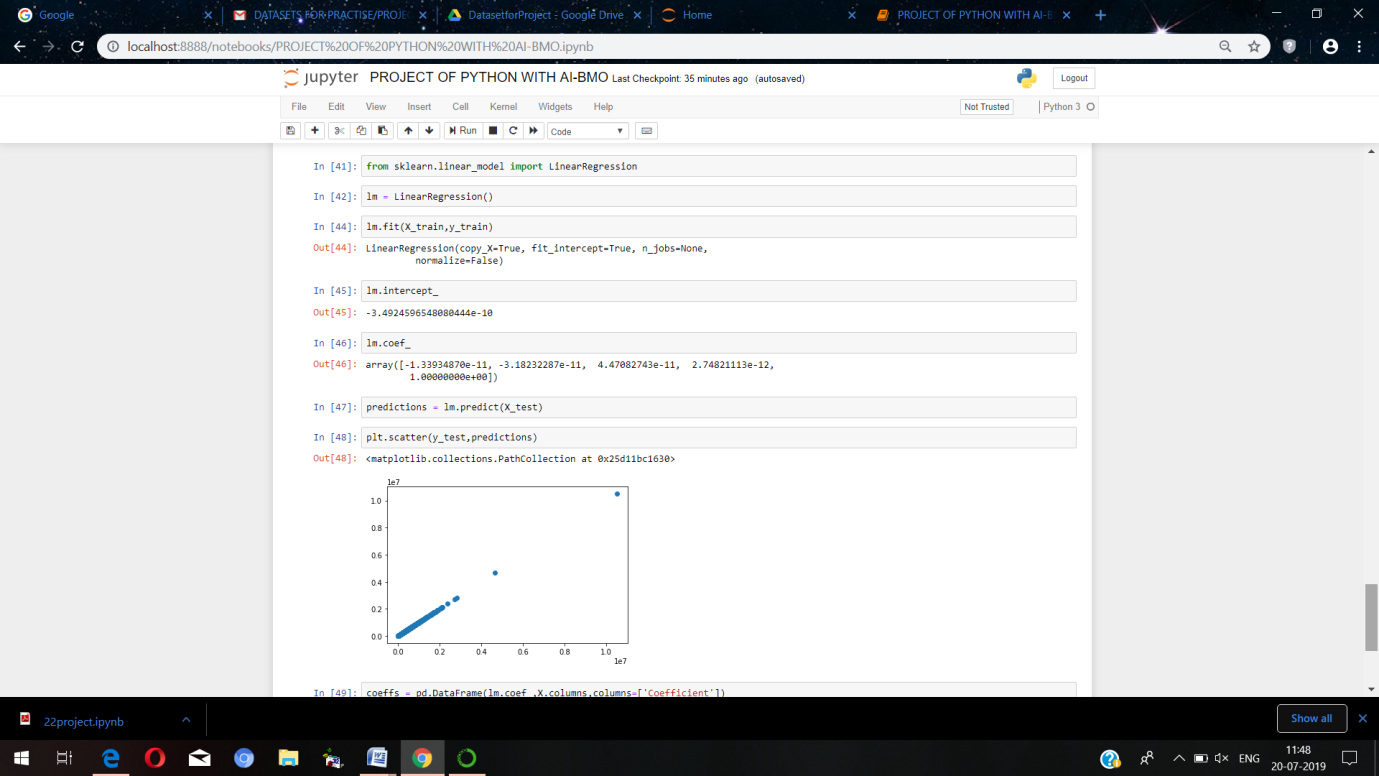


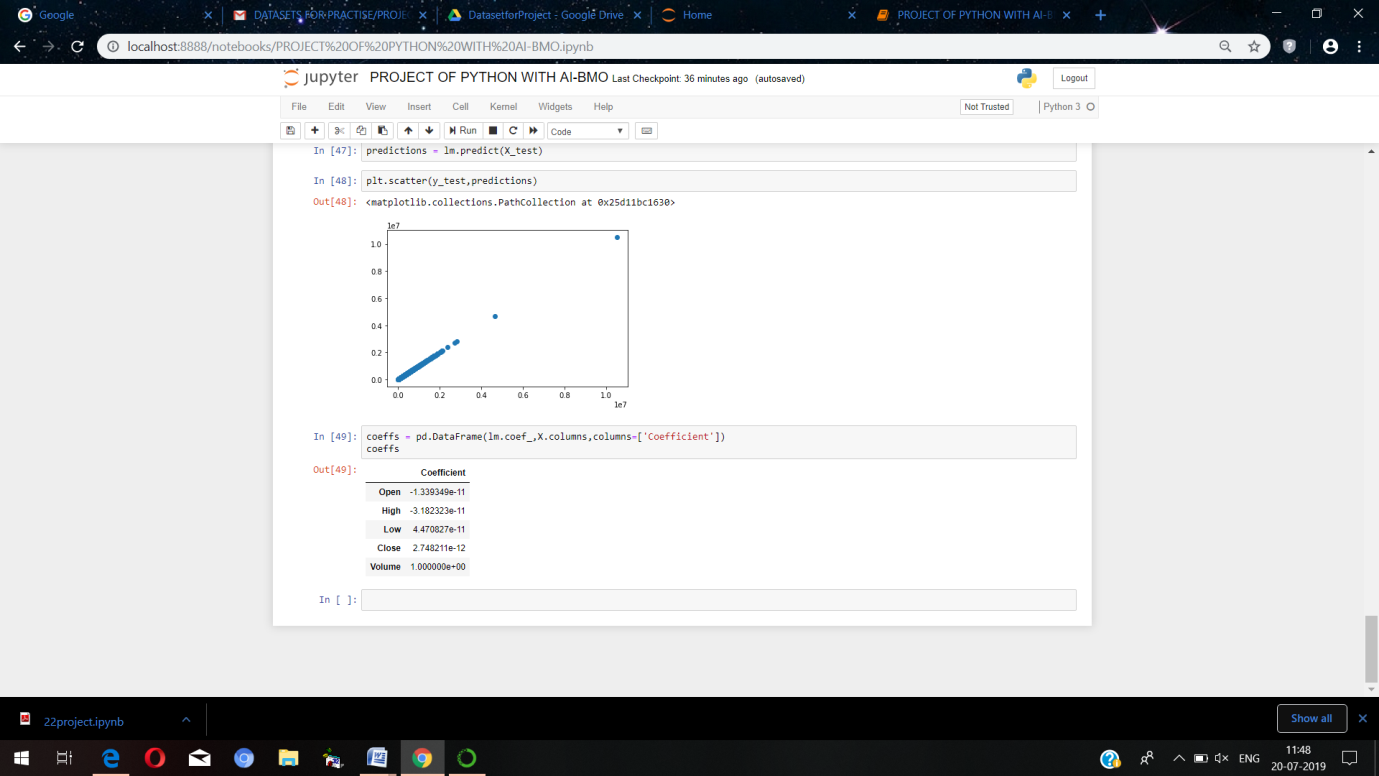












CODING

In [9]: **importnumpyasnp**

**importpandasaspd**

**importmatplotlib.pyplotasplt**

**importseabornassns**

In [10]: **fromsklearnimport** datasets

In [13]: bmo = pd.read\_csv('BMO.csv')

In [15]: bmo.info()

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

RangeIndex: 2769 entries, 0 to 2768

Data columns (total 6 columns):

Date 2769 non-null object

Open 2769 non-null float64

High 2769 non-null float64

Low 2769 non-null float64

Close 2769 non-null float64

Volume 2769 non-null int64

dtypes: float64(4), int64(1), object(1)

memory usage: 129.9+ KB

In [16]: bmo.describe()

Out[16]:

|  | **Open** | **High** | **Low** | **Close** | **Volume** |
| --- | --- | --- | --- | --- | --- |
| **count** | 2769.000000 | 2769.000000 | 2769.000000 | 2769.000000 | 2.769000e+03 |
| **mean** | 57.754298 | 58.204323 | 57.245887 | 57.745811 | 5.509159e+05 |
| **std** | 9.818726 | 9.735716 | 9.950031 | 9.842291 | 4.747840e+05 |
| **min** | 19.610000 | 20.390000 | 19.320000 | 19.510000 | 8.400000e+03 |
| **25%** | 55.250000 | 55.810000 | 54.640000 | 55.250000 | 2.880280e+05 |
| **50%** | 59.480000 | 59.870000 | 59.050000 | 59.480000 | 4.649240e+05 |
| **75%** | 63.270000 | 63.670000 | 62.890000 | 63.260000 | 7.067590e+05 |
| **max** | 78.340000 | 78.560000 | 77.710000 | 78.030000 | 1.049549e+07 |

In [17]: bmo.head()

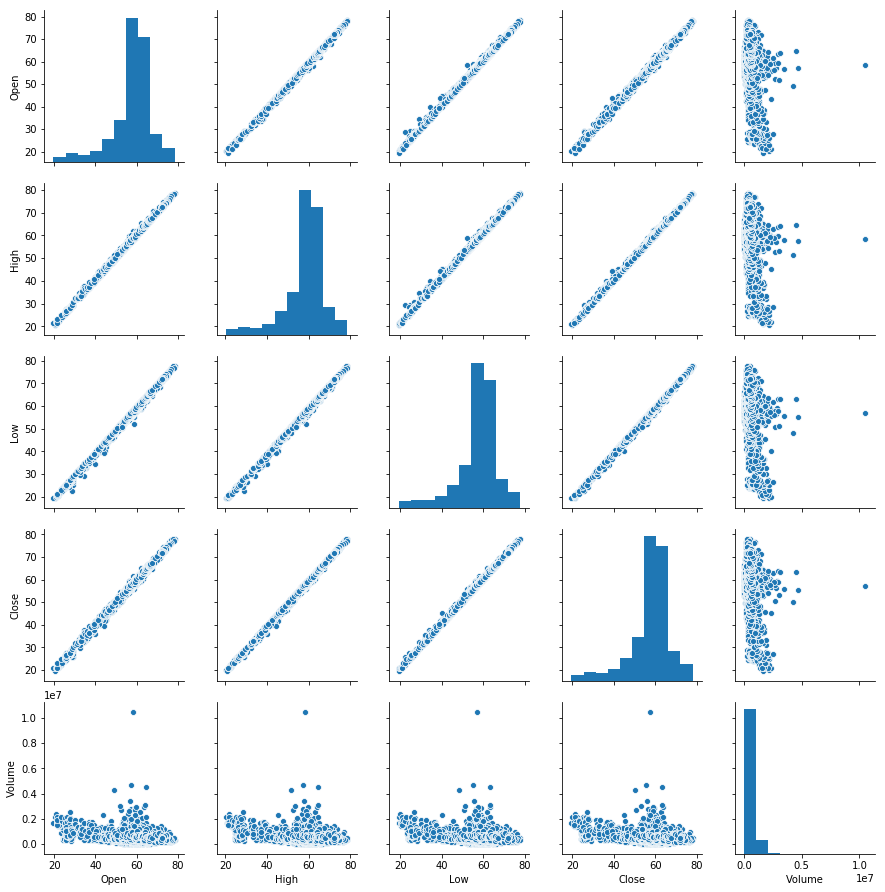
Out[17]:

|  | **Date** | **Open** | **High** | **Low** | **Close** | **Volume** |
| --- | --- | --- | --- | --- | --- | --- |
| **0** | 2006-01-03 | 55.71 | 57.51 | 55.71 | 57.25 | 73600 |
| **1** | 2006-01-04 | 57.50 | 58.51 | 57.23 | 58.36 | 87300 |
| **2** | 2006-01-05 | 57.95 | 58.09 | 57.32 | 57.50 | 48500 |
| **3** | 2006-01-06 | 57.30 | 57.43 | 56.99 | 57.16 | 26500 |
| **4** | 2006-01-09 | 56.40 | 57.06 | 56.32 | 56.67 | 45400 |

In [55]: sns.pairplot(bmo)

Out[55]:

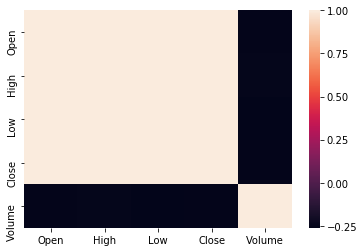
<seaborn.axisgrid.PairGrid at 0x25d150f2320>



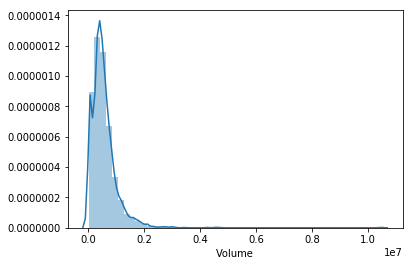
In [19]: sns.heatmap(bmo.corr())

Out[19]:

<matplotlib.axes.\_subplots.AxesSubplot at 0x25d0ca08e80>

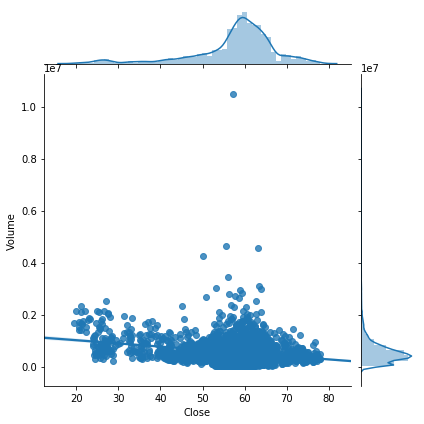


In [21]: sns.distplot(bmo['Volume'],bins=50);



In [34]: g = sns.JointGrid(x="Close", y="Volume", data=bmo)

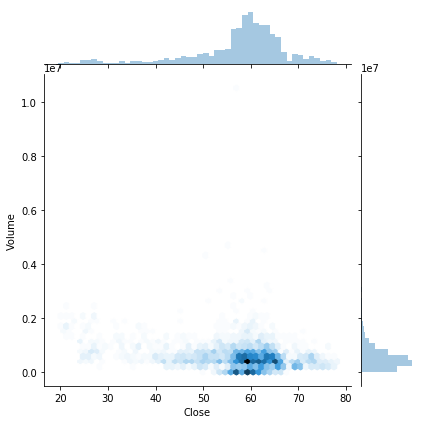
g = g.plot(sns.regplot, sns.distplot)



In [51]: sns.jointplot(x="Close", y="Volume",data=bmo,kind='hex')

Out[51]:

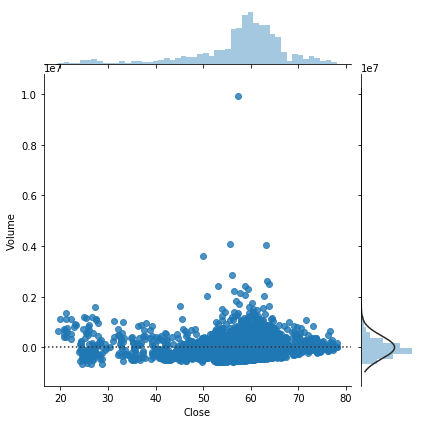
<seaborn.axisgrid.JointGrid at 0x25d11d99908>



In [36]: sns.jointplot(x="Close", y="Volume",data=bmo,kind='resid')

Out[36]:

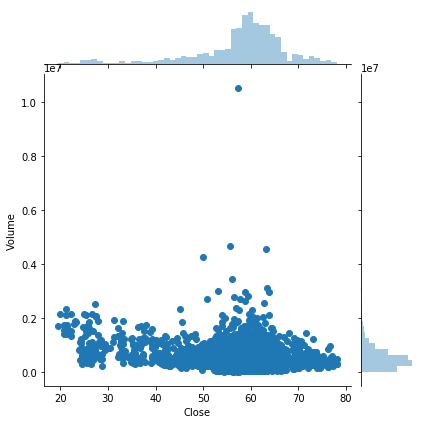
<seaborn.axisgrid.JointGrid at 0x25d105859e8>



In [37]: sns.jointplot(x="Close", y="Volume",data=bmo,kind='scatter')

Out[37]:

<seaborn.axisgrid.JointGrid at 0x25d1069e828>



In [38]: X = bmo[['Open','High','Low','Close','Volume']]

y = bmo['Volume']

In [39]: **fromsklearn.model\_selectionimport** train\_test\_split

X\_train,X\_test, y\_train, y\_test = train\_test\_split(X,y,test\_size=0.3, random\_state=101)

In [43]: X\_train.head()

Out[43]:

|  | **Open** | **High** | **Low** | **Close** | **Volume** |
| --- | --- | --- | --- | --- | --- |
| **2013** | 66.50 | 66.74 | 66.02 | 66.25 | 315157 |
| **1591** | 59.26 | 59.68 | 59.13 | 59.58 | 338105 |
| **1298** | 63.84 | 64.31 | 63.31 | 63.37 | 3094811 |
| **1240** | 59.63 | 59.63 | 58.75 | 59.35 | 447667 |
| **2215** | 72.03 | 72.23 | 71.27 | 72.09 | 577904 |

In [41]: **fromsklearn.linear\_modelimport** LinearRegression

In [42]: lm = LinearRegression()

In [44]: lm.fit(X\_train,y\_train)

Out[44]:

LinearRegression(copy\_X=True, fit\_intercept=True, n\_jobs=None,

normalize=False)

In [45]: lm.intercept\_

Out[45]:

-3.4924596548080444e-10

In [46]: lm.coef\_

Out[46]:

array([-1.33934870e-11, -3.18232287e-11, 4.47082743e-11, 2.74821113e-12,

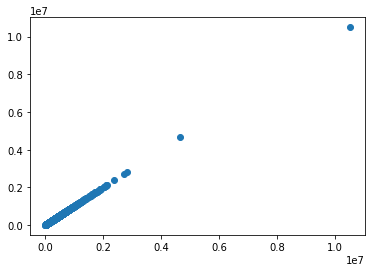
1.00000000e+00])

In [47]: predictions = lm.predict(X\_test)

In [48]: plt.scatter(y\_test,predictions)

Out[48]:

<matplotlib.collections.PathCollection at 0x25d11bc1630>



In [49]: coeffs = pd.DataFrame(lm.coef\_,X.columns,columns=['Coefficient'])

coeffs

Out[49]:

|  | **Coefficient** |
| --- | --- |
| **Open** | -1.339349e-11 |
| **High** | -3.182323e-11 |
| **Low** | 4.470827e-11 |
| **Close** | 2.748211e-12 |
| **Volume** | 1.000000e+00 |

CONCLUSION

*There are no doubts that AI technologies are the future. Considering the increasing popularity of the trend and the number of people ready to invest in it, the global AI market is going to reach $89.8 billion by 2025.*

*The PL is what we should think about at first. The complexity of coding as well as the availability of the experienced and qualified developers are crucial moments to take into account as well.*

*The marketing can make use of AI by means of the tech stack of the processes that are made manually by employees can be automated, it can bring more efficiency and quickly analyze large data sets, for example. Gartner says that by 2020 AI technologies will be used in at least one of the sales processes by 30% of companies over the world. Besides that, according to Accenture reports, the profitability will rise by 38% by 2035 and AI will create $14 trillion of additional revenue.*

*The e-commerce sales are expected to be about $4.5 trillion by 2021. And that’s not without AI technologies used. Thanks to the AI the sites provides the customers with 24/7 service and assistance by means of the chatbots, improve consumers experience by analyzing the CRM data in moments with AI tech, IOT, and other examples of using AI in e-commerce. High diversity of built-in libraries, simple syntax, readability, compatibility, rapid testing of sophisticated algorithms, accessibility to nonprogrammers, and other features make Python worthy of your attention. All that ease the process, save your budget and increase the popularity of Python. Taking to account all the advantages we get using the PL, the conclusion is obvious — Python is what we need to consider to your AI-based project.*

Bibliography

*The contents have been gathered from the following:*

*✓ Information: Google*

*✓ Images: Google Images*

*✓ Coding and snapshots: Self-performed.*

\_\_\_\_\_\_\_\_\_\_\_\_\_\_