Master's Thesis

IUBH University of Applied Sciences

M.Sc. Computer Science

Python Written Assessment

Author name: Ketansingh Thakur

Matriculation no: 92123502

Tutor Name: Lino Antoni Giefer

Submission Date: April 14, 2022

TABLE OF CONTENTS

List of Diagrams

Table of Abbreviations

IFDT: -	Ideal Function Data Table
TDT: -	Training Data Table
TDDT: -	Test Data’s Data Table
UT: -	Unit Test
TD: -	Test Data
SQL: -	Structured Query Language
CSV: -	Comma Separated Values
HTML: -	Hypertext Markup Language
LBL: -	Line by Line

Introduction

For getting the essential knowledge the coursebook give because the brief idea for the completion of the assessment. Through this assessment we'll acquire the fundamentals of python and undergo different technologies.

Through recent studies for the completion of the we are able to find and learn lot of things like testing, data integration, data comparison with different files and prepared to learn new packages for visualization. during this assessment we have got 3 dataset and every one of them are csc files. this is often a relevant topic because this may help as find things from an outsized dataset. Will help possesses to minimize the trouble doing it find the answer and reducing the time therefore the subject has relevancy.

Our main objective is to hunt out the acceptable result from the info set that has been provided to us. As a result, able to we will find the simplest fix outcome and able to visualize it. We are able to cover topics like pandas, SQLAlchemy, NumPy, matplotlib, bokeh etc.

The scope of this study is restricted to finding that uses training data to choose the four ideal functions which are the simplest fit out of the fifty provided. There are basic criteria for selecting the simplest functions from the fifty provided we'd like to finish the standards to accumulate the outcomes. Though this assessment able to "> we will create a python program that helps to unravel the standards and able to achieve the result. Also, we'll use the program for other data set that following the standards of the program.

In addition to all or any of those we will work with version system. And successfully added the project to version system.

Python Research on Dataset

This is research done on python language with using data set. during this research we have three csv files. With the three csv files we must find the simplest fit functions. All the csv file consists of X and Y values which having negative and positive floating-point numbers. The three csv files That we've is.

1. Ideal.csv
2. Test.csv
3. Train.csv

The ideal file consists of fifty ideal function and other two files contain test data and train data. Packages used for completing the programs are Pandas.

1. NumPy
2. Matplotlib
3. SQLAlchemy
4. Bokeh
5. Unit test

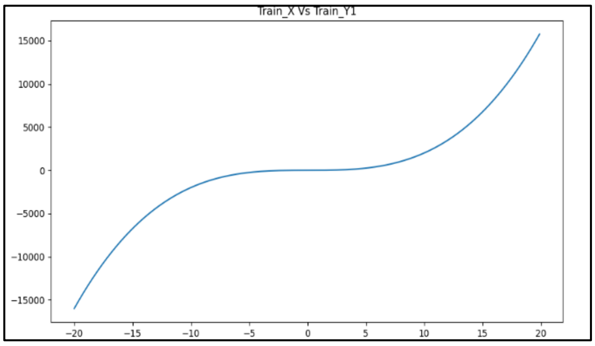
These are the important packages utilized in our python assessment each of these packages having their own importance within the program. of those packages are installed using pip. Pandas could also be a python package providing fast, flexible, and expressive data structures designed to make working with “relational” or “labelled” data both easy and intuitive. NumPy could also be a python package want to do mathematical Functions in python it's important package of python. Matplotlib could also be a library for creating all types of static, animated, and interactive visualizations in Python. it'll open window for displaying the figure. SQLAlchemy is that the Python SQL package toolkit and Object Relational Mapper that gives application developers the complete power and adaptability of SQL.Bokeh could also be a Python library for creating interactive visualizations for contemporary web browsers. it's almost like matplotlib it'll display the figure as an html enter the browser.

Unit test may be a Python library which is used for first level of software testing where the smallest testable parts of a software are tested. this is often used to validate that each unit of the software performs as designed. we will identify the capability of our program. These are all the important packages that's used in our program.

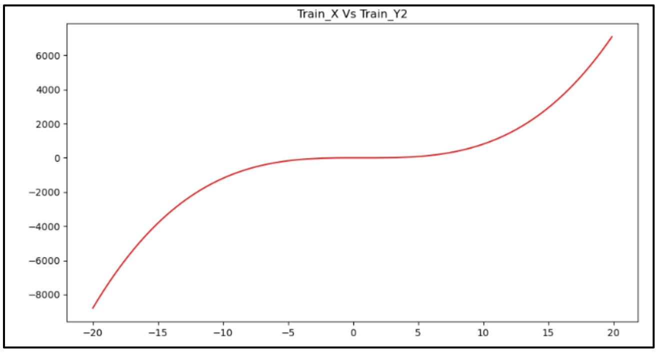
For using the csv enter our project, we use the subsequent lines of code in the program.

```
Train_data_set = pd.read_csv('train.csv')
Ideal_data_set = pd.read_csv('ideal.csv')
Test_data_set = pd.read_csv('test.csv')
```

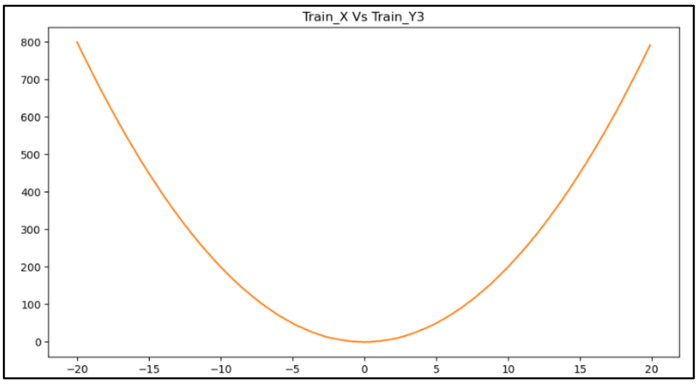
For plotting the training data, we use matplotlib package within the program which has 'axs[value,value]' allows to put graphs in the desire position we will be plotting four graphs for training data. The plotted graphs are shown below.



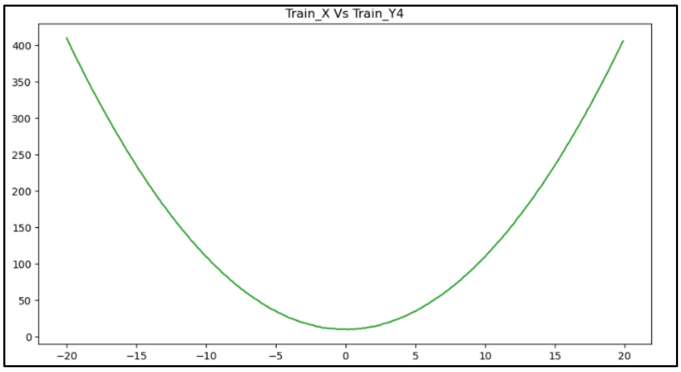
Plotting the deviations of Train_X, Train_Y1.



Plotting the deviations of Train_X, Train_Y2.

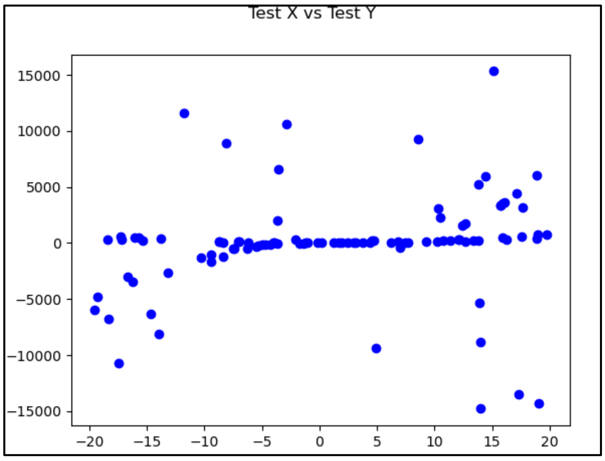


Plotting the deviations of Train_X, Train_Y3.



Plotting the deviations of Train_X, Train_Y4.

Plotting the deviations of Test_X, Test_Y.



With the help of the matplotlib package we'll visualize the test data and thus the training data as graph of these graphs are easy to know. we'll easily identify the variation of each graph.

For finding the proper function for the training function we'd wish to attenuate the sum of all y-deviations squared (Least-Square). For that we'd wish to feature a function within the program to hunt out the deviation. I even have created a function for locating ideal function by following criterion 1.

```
def ideal_function (train data, ideal data):
if not isinstance(train_data, pd.Series):
                    raise MyException(train_data, "Exception raised! {} Must be a Pandas
series".format(train_data))
          squared_sum = []
          for j in range(1, len(ideal_data.columns)):
squared_sum.append((j, sum(abs(train_data - ideal_data['y'+str(j)].values))))
squared_sum.sort(key = lambda x: x[1])
   return squared_sum[0]


Ideal_Y1  = ideal_function(Train_Y1, Ideal_data)
Ideal_Y2  = ideal_function(Train_Y2, Ideal_data)
Ideal_Y3  = ideal_function(Train_Y3, Ideal_data)
Ideal_Y4  = ideal_function(Train_Y4, Ideal_data)

print("Ideal Function (Y1) is:", "y" + str(Ideal_Y1[0]),"," ' Ideal Function (Y2) is:',"y" +
          str(Ideal_Y2[0]))
print("Ideal Function (Y3) is:", "y" + str(Ideal_Y3[0]), "," ' Ideal Function (Y4) is:', "y" +
          str(Ideal_Y4[0]))
```

As a results of the function, we will find the utmost deviation. Following below are the utmost deviation.

```
Maximum_dev1: 0.4999210000000005
Maximum_dev2: 0.49989428999999996
```

```
Maximum_dev3: 0.49958680000000033
Maximum_dev4: 0.49947372999999995
```

After finding the utmost deviation we can map the individual test cases to the four ideal functions. As a result, we will create the following code form mapping the functions.

“Mapping for first ideal function”

```
if abs(Test_data['y'][x] - Ideal_data['y' + str(Ideal_Y1[0])][i]) <= maximum_dev1 * np.sqrt(2):
    Test_data['No of Ideal Y'][x] = 'y' + str(Ideal_Y1[0])
    Test_data['Delta Y'][x] = abs(Test_data['y'][x] - Ideal_data['y' + str(Ideal_Y1[0])][i])
```

“Mapping for second ideal function”

```
if abs(Test_data['y'][x] - Ideal_data['y' + str(Ideal_Y2[0])][i]) <= maximum_dev2 * np.sqrt(2):
    Test_data['No of Ideal Y'][x] = 'y' + str(Ideal_Y2[0])
    Test_data['Delta Y'][x] = abs(Test_data['y'][x] - Ideal_data['y' + str(Ideal_Y2[0])][i])
```

“Mapping for third ideal function”

```
if abs(Test_data['y'][x] - Ideal_data['y' + str(Ideal_Y3[0])][i]) <= maximum_dev3 * np.sqrt(2):
    Test_data['No of Ideal Y'][x] = 'y' + str(Ideal_Y3[0])
    Test_data['Delta Y'][x] = abs(Test_data['y'][x] - Ideal_data['y' + str(Ideal_Y3[0])][i])
```

“Mapping for fourth ideal function”

```
if abs(Test_data['y'][x] - Ideal_data['y' + str(Ideal_Y4[0])][i]) <= maximum_dev4 * np.sqrt(2):
    Test_data['No of Ideal Y'][x] = 'y' + str(Ideal_Y4[0])
    Test_data['Delta Y'][x] = abs(Test_data['y'][x] - Ideal_data['y' + str(Ideal_Y4[0])][i]
```

“To calculate the number of test data points assigned to the ideal function”

```
number_assigned = len(Test_data) - Test_data.count()
print("The number of x-y-pair values that can be assigned to the four chosen ideal functions:",
            len(Test_data) - number_assigned.values[3])
```

The number of x-y-pair values that may be assigned to the four chosen ideal functions: 100We got the worth as 100.

For the assessment I have created a python_assignment.db using SQLAlchemy which consists of three tablesideal_functions

1. test_data_table
2. training_data

The ideal functions table may be a table loaded with fifty ideal functions which are provided in the csv. The training data table is loaded with the training data set which is provided within the csv.

ideal functions
X
Y1
Y2
.
.
Y50

training data
X
Y1
Y2
Y3
y4

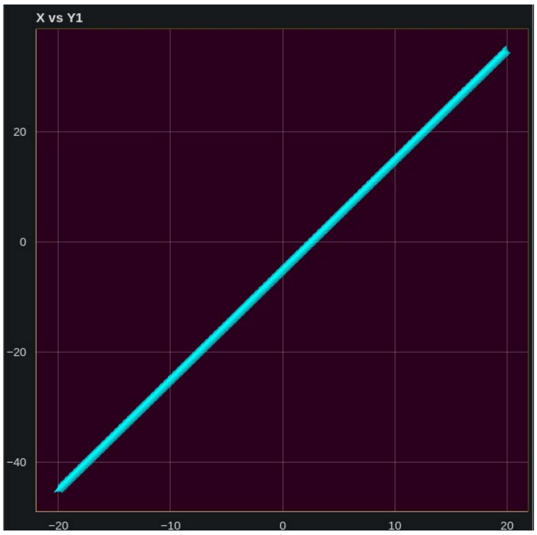
schema of ideal function table schema of training data table

After the creation of both tables, we'd like to load the test data line by line from the csv file if it matches the criterion of the for ideal functions, we'd like to store the results in another table which consist of four columns.

test_data_table
X
Y
Delta y
No of ideal y

schema of training data table.

For plotting the four ideal functions data, we use bokeh package within the program which will display the output in the browser.



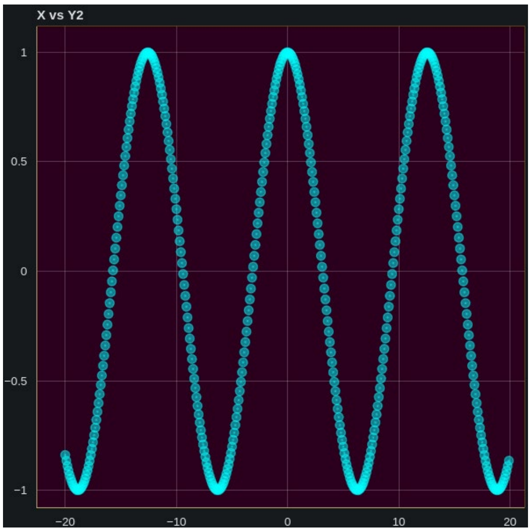
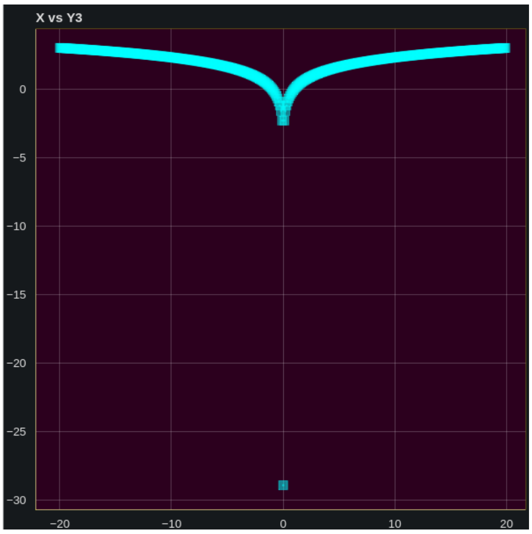


Figure: - Plotting the deviations of X and Y1

Figure: - Plotting the deviations of X and Y2.



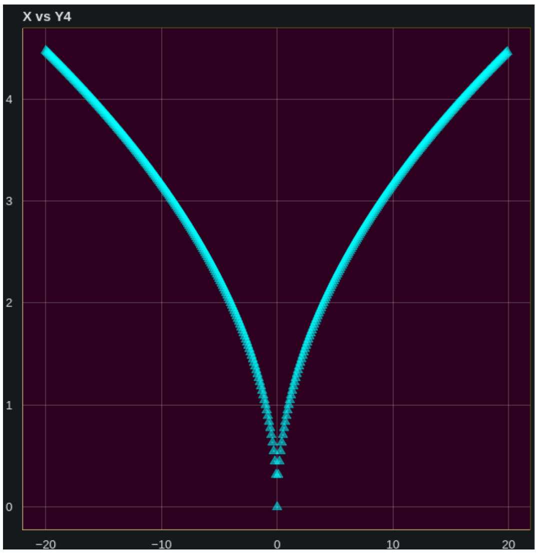


Figure: - Plotting the deviations of X and Y3.

Figure: - Plotting the deviations of X and Y4.

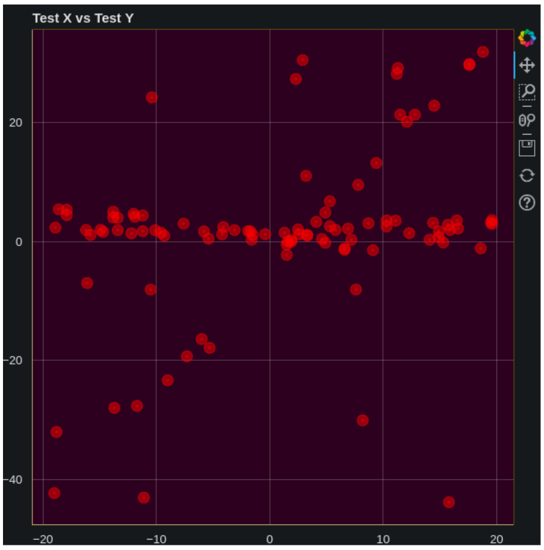


Figure: - Plotting the deviations of Test X and Test Y.

Additional Task

In the task we have a GitHub repository which is using by different people and in the repository, we've a branch develop. On this branch all the operations of the developer team have combined us a developer of this project we should always also need no work in same way, so we need to pull the branch to the local environment. Any changes wiped out the branch it only effects in the develop branch it won’t affect in the main branch.

The command required to pull the remote branch is.

```
git fetch origin
```

git fetch is employed for fetching from remote repository to make we have all the latest changes downloaded.

```
git branch
```

git branch is employed to display all the branches of the current repository. within the repository we can see the develop branch and we need to move to the develop branch, so we’d like to type the command.

```
git checkouts develop
```

Now we are within the branch develop. Then we are going to edit our program codes in our file and made some necessary changes into the file so we need to push all changes to the remote repository that our teammate can review and merge the change with their codes, so we’d like to push the code. very first thing   before pushing we need to add the changes.

```
git add.
```

Then we to commit these changes.

```
git commit -m “changes in develop branch”
```

So, we will see that all the changes have been staged. and that we need to push all these changes to the remote repository.

```
git push -u origin develop
```

then we will see that the branch is pushed to remote repository and our team members can view the changes of the branch that we have modified and the and add those changes to their code after merging the branch.

Conclusion

I have had all the topics of the course book helps on complete the assessment the course book gives me a basic idea of the way to work with python and its packages course book was use full till the top. At the highest of this assessment, we'll achieve the ultimate goal on visualising the perfect data. Initially we had three csv file which we have visualised all the specified data and summarises with each of the diagram.

Main advantage in working with this assessment is that able to we will gather much knowledge about data structures packages visual library and able to find how of working with all of them the main objective behind this assessment is that we'd like to use. we've covered through topics like SQLAlchemy, Pandas, matplotlib, Bokeh, NumPy etc.

Through recent studies for the completion of the we are able to find and learn lot of things like testing, data integration, data comparison with different files and prepared to learn new packages for visualization. during this assessment we have got 3 dataset and every one of them are csv files. this is often often a relevant topic because this may help as find things from an outsized dataset. Will help possesses to minimize the trouble doing it find the answer and reducing the time therefore the subject has relevancy.

Finally, we are fulfilled all the necessity the task has given to us. While doing the additional task for git we got an entire idea about the way to work with during a team environment while working within the same project it gives a thought close to manage all the changes wiped out the project shouldn't affect the changes of the teammates done.

Bibliography

1. Books read:

    1.1. Python Course Book.

    1.2. Python: The complete reference.

    1.3. Python for Data Analysis

    1.4. Pandas for Everyone

2. Sites Visited:

    2.1. Pandas official site pandas

    2.2. Bokeh official site Bokeh

    2.3. NumPy official site NumPy

    2.4. W3school W3Schools

    2.5. SQLAlchemy SQLAlchemy

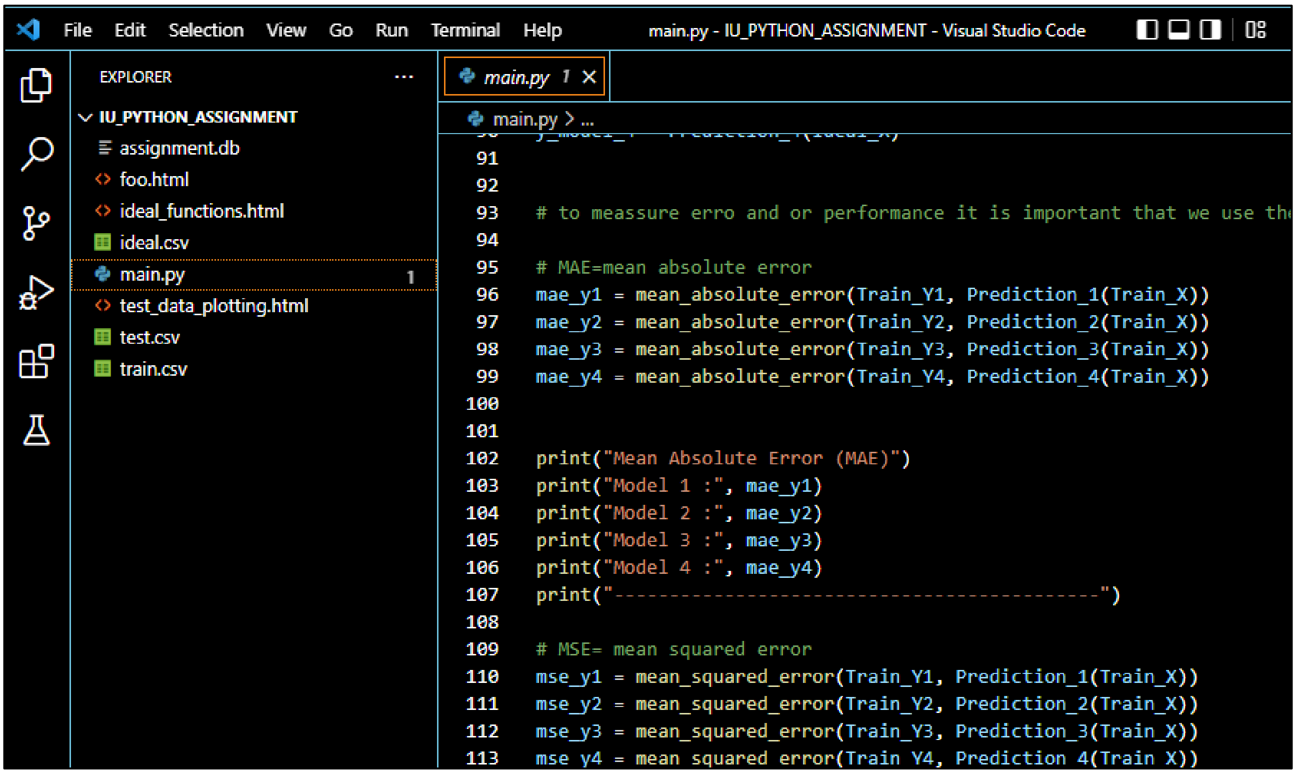
    2.6. Stack overflow Stack Overflow

List of Appendices

Appendices and materials

1. VScode

VScode is the tool in which I created my python code.



```python


# to meassure erro and or performance it is important that we use th

# MAE=mean absolute error
mae_y1 = mean_absolute_error(Train_Y1, Prediction_1(Train_X))
mae_y2 = mean_absolute_error(Train_Y2, Prediction_2(Train_X))
mae_y3 = mean_absolute_error(Train_Y3, Prediction_3(Train_X))
mae_y4 = mean_absolute_error(Train_Y4, Prediction_4(Train_X))


print("Mean Absolute Error (MAE)")
print("Model 1 :", mae_y1)
print("Model 2 :", mae_y2)
print("Model 3 :", mae_y3)
print("Model 4 :", mae_y4)
print("-------------------------------------------")

# MSE= mean squared error
mse_y1 = mean_squared_error(Train_Y1, Prediction_1(Train_X))
mse_y2 = mean_squared_error(Train_Y2, Prediction_2(Train_X))
mse_y3 = mean_squared_error(Train_Y3, Prediction_3(Train_X))
mse_y4 = mean_squared_error(Train_Y4, Prediction_4(Train_X))
```