**Ajay Kumar Garg Engineering College**

27th KM STONE DELHI-HAPUR BYPASS ROAD, P.O. ADHYATMIK NAGAR

GHAZIABAD-201009

An Internship Training Report

on

**MACHINE LEARNING WITH PYTHON**



At

**ANTRIX ACADEMY, NOIDA**

Submitted by: **Ayush Jain**

Year: **2nd**

Semester: **3rd**

Section: **CSE 1**

Branch: **Computer Science Engineering**

Roll No.: **2000270100058**

Submitted to: **Mr. Vishal Choudhary**

**(Asst. Professor)**

**CERTIFICATE OF TRAINING**



**Ajay Kumar Garg Engineering College**

27th KM STONE DELHI-HAPUR BYPASS ROAD, P.O. ADHYATMIK NAGAR

GHAZIABAD-201009

**INTERNSHIP CERTIFICATE**

This is to certify that **AYUSH JAIN** student of **AJAY KUMAR GARG ENGINEERING COLLEGE** B. Tech Second Year **Computer Science Engineering** branch, has undergone Internship Training in Machine Learning with Python from 1 September 2021 to 21 October 2021

**Mr. Vishal Choudhary Dr. Sunita Yadav**

Faculty Internship Assessment Lab HoD, CSE

**ABSTRACT**

This project based on Machine Learning gives us a clear idea of how accurate predictions can be done by the computer when the data is fed to the computer and the computer is trained using the provided dataset. Through this project, we can clearly see at small level how computers can achieve intelligence and predict outcomes based on the inputs fed to the computer by the user or the controller. This can help in making predictions automated and free of user interruption. This project uses Regression of Supervised learning and the models used are Mean Squared Error and Linear Regression Models. These are the basic Regression Models which are pre-built by the SKLEARN module which can be directly used as functions in PYTHON.

List of Figures

Project Report……………………………………………………………….…1

Certificate…………………………………………………………….......……2

Abstract………………….……………………………………………….……4

List of figures………………….……………………………….……….....….5

Introduction to organization…………………………………….………….…6

Objective of project………………………………………………………………….………………….7

Scope of project……………………………………………………………………….……….…………8

Python…………………………………………………………..……………………...……………….9-14

Machine Learning………………………………………..……………………………………….15-20

Regression………………………………..……………………………………………..……………21-24

Project………….…………………………..…………………………………………..……………………25

Libraries used……………………………………..……………………..………………………….25-26

Source Code……………………………………………………………………………..……………27-33

Advantages and Limitations of Project……………..……………………….………………..33

Conclusion and Reference………………………………………………………………………34-35

**INTRODUCTION TO ORGANIZATION**

**Antrix Academy of Data Science (AADS)** is a renowned Data Analytics training organization located in Noida, well known for providing quality education in advanced fields of Data Science/Data Analytics, such as Advanced Data Mining, Data Analytics, Data Visualization, Data Predictive Analytics, and Data-Based Crime Investigation, Computer Training Institutes, SAP Training Institutes, Computer Training Institutes For Python, Computer Training Institutes For Java, Computer Training Institutes For Web Designing, Computer Training Institutes For Data Science, Software Testing Institutes, Computer Training Institutes For Amazon Web Services and much more. Antrix Academy Of Data Science offers short-term courses and certificate courses. The Data Science Program Modules has been created to meet the increasing demand for Data Scientist in all major Sectors. This will help organizations create future leaders in the Data Science sector which will make India the next Data-Driven superpower. The course encompasses an understanding of the general management concepts along with an in-depth understanding of the core subjects in Data Science, Data Analysis, Regression.

We are the fastest growing training company in Data Science, have trained more than 5000+ candidates in the data science domain in collaboration with our institutional partnership. We are the delivery partner for Microsoft, Adobe, CompTIA, Hortonworks, SAS, Tableau, and IBM CE Partner- HeadStart Education. At AADS, many professionals from the corporate world have joined the current batch of Data Science Using Excel, R, Python, SPSS, SAS, Tableau, and Machine Learning from AADS. Our Data Science Program Modules has been created to meet the increasing demand for Data Scientist in all major Sectors. This will help organizations create future leaders in the Data Science sector which will make India the next Data-Driven superpower. The course encompasses an understanding of the general management concepts along with an in-depth understanding of the core subjects in Data Science, Data Analysis, Regression. A Great opportunity for Working professionals to learn the Most Demanding career of Data Science through AADS from prominent faculties of AADS like **Mr. Rajinder Chitoria** (15+ years of diversified experience in the field of data mining & data visualization.

**OBJECTIVE OF PROJECT**

The project aims at teaching the computer to predict outcomes based on training given to the system based on dataset given. This project teaches the computer to act according to the values of different prerequisites of the dataset. The program takes the dataset, divides it into training and testing half, and builds the model based on training dataset. The accuracy of model is calculated based on the outcomes of testing dataset and the actual output in the data.

As an example, in this project, we have a Auto MPG Dataset, so to get the MPG of a vehicle, user has to enter various specifications of the vehicle and the model predicts the MPG based on the training.

**SCOPE OF THE PROJECT**

As the modern world is overfilled with data, data management and usage are major task. A lot of data is present but only a little of it is used correctly and efficiently. So, in today’s world, this method of using dataset to predict various outcomes makes this project very useful in coming future. Using this method of model creation and usage, we are training our computer to do some tasks without user intervention. This proves the scope of the project, as it makes easier for computer to take decisions and requires lesser user intervention.

**PYTHON**

Python is an interpreted high-level general-purpose programming language. Its design philosophy emphasizes code readability with its use of significant indentation. Its language constructs as well as its object-oriented approach aim to help programmers write clear, logical code for small and large-scale projects.

Python is dynamically-typed and garbage-collected. It supports multiple programming paradigms, including structured (particularly, procedural), object-oriented and functional programming. It is often described as a "batteries included" language due to its comprehensive standard library.

Guido van Rossum began working on Python in the late 1980s, as a successor to the ABC programming language, and first released it in 1991 as Python 0.9.0. Python 2.0 was released in 2000 and introduced new features, such as list comprehensions and a cycle-detecting garbage collection system (in addition to reference counting). Python 3.0 was released in 2008 and was a major revision of the language that is not completely backward-compatible. Python 2 was discontinued with version 2.7.18 in 2020.

Python consistently ranks as one of the most popular programming languages.

Python uses dynamic typing and a combination of reference counting and a cycle-detecting garbage collector for memory management.[63] It also features dynamic name resolution (late binding), which binds method and variable names during program execution.

Python's design offers some support for functional programming in the Lisp tradition. It has filter, map and reduce functions; list comprehensions, dictionaries, sets, and generator expressions. The standard library has two modules (itertools and functools) that implement functional tools borrowed from Haskell and Standard ML.

The language's core philosophy is summarized in the document The Zen of Python (PEP 20), which includes aphorisms such as:[66]

* Beautiful is better than ugly.
* Explicit is better than implicit.
* Simple is better than complex.
* Complex is better than complicated.
* Readability counts.

Syntax and semantics

Python is meant to be an easily readable language. Its formatting is visually uncluttered, and it often uses English keywords where other languages use punctuation. Unlike many other languages, it does not use curly brackets to delimit blocks, and semicolons after statements are allowed but are rarely, if ever, used. It has fewer syntactic exceptions and special cases than C or Pascal.

### Indentation

Python uses whitespace indentation, rather than curly brackets or keywords, to delimit blocks. An increase in indentation comes after certain statements; a decrease in indentation signifies the end of the current block. Thus, the program's visual structure accurately represents the program's semantic structure. This feature is sometimes termed the off-side rule, which some other languages share, but in most languages, indentation does not have any semantic meaning. The recommended indent size is four spaces.

### Statements and control flow

Python's statements include (among others):

* The assignment statement, using a single equals sign =.
* The if statement, which conditionally executes a block of code, along with else and elif (a contraction of else-if).
* The for statement, which iterates over an iterable object, capturing each element to a local variable for use by the attached block.
* The while statement, which executes a block of code as long as its condition is true.
* The try statement, which allows exceptions raised in its attached code block to be caught and handled by except clauses; it also ensures that clean-up code in a finally block will always be run regardless of how the block exits.
* The raise statement, used to raise a specified exception or re-raise a caught exception.
* The class statement, which executes a block of code and attaches its local namespace to a class, for use in object-oriented programming.
* The def statement, which defines a function or method.
* The with statement, which encloses a code block within a context manager (for example, acquiring a lock before the block of code is run and releasing the lock afterwards, or opening a file and then closing it), allowing resource-acquisition-is-initialization (RAII)-like behaviour and replaces a common try/finally idiom.[81]
* The break statement, exits from a loop.
* The continue statement, skips this iteration and continues with the next item.
* The del statement, removes a variable, which means the reference from the name to the value is deleted and trying to use that variable will cause an error. A deleted variable can be reassigned.
* The pass statement, which serves as a NOP. It is syntactically needed to create an empty code block.
* The assert statement, used during debugging to check for conditions that should apply.
* The yield statement, which returns a value from a generator function and yield is also an operator. This form is used to implement coroutines.
* The return statement, used to return a value from a function.
* The import statement, which is used to import modules whose functions or variables can be used in the current program.

### Methods

Methods on objects are functions attached to the object's class; the syntax instance.method(argument) is, for normal methods and functions, syntactic sugar for Class.method(instance, argument). Python methods have an explicit self-parameter to access instance data, in contrast to the implicit self (or this) in some other object-oriented programming languages (e.g., C++, Java, Objective-C, or Ruby).[96] Apart from this, Python also provides methods, often called dunder methods (due to their names beginning and ending with double-underscores), to allow user-defined classes to modify how they are handled by native operations such as length, comparison, in arithmetic operations, type conversion, and many more.

### Typing

Python uses duck typing and has typed objects but untyped variable names. Type constraints are not checked at compile time; rather, operations on an object may fail, signifying that the given object is not of a suitable type. Despite being dynamically-typed, Python is strongly-typed, forbidding operations that are not well-defined (for example, adding a number to a string) rather than silently attempting to make sense of them.

## Libraries in Python

Python's large standard library, commonly cited as one of its greatest strengths, provides tools suited to many tasks. For Internet-facing applications, many standard formats and protocols such as MIME and HTTP are supported. It includes modules for creating graphical user interfaces, connecting to relational databases, generating pseudorandom numbers, arithmetic with arbitrary-precision decimals, manipulating regular expressions, and unit testing.

As of September 2021, the Python Package Index (PyPI), the official repository for third-party Python software, contains over 329,000[118] packages with a wide range of functionality, including:

* Automation
* Data analytics
* Databases
* Documentation
* Graphical user interfaces
* Image processing
* Machine learning
* Mobile apps
* Multimedia
* Computer networking
* Scientific computing
* System administration
* Test frameworks
* Text processing
* Web frameworks
* Web scraping

## Development environments

Most Python implementations (including CPython) include a read–eval–print loop (REPL), permitting them to function as a command line interpreter for which the user enters statements sequentially and receives results immediately.

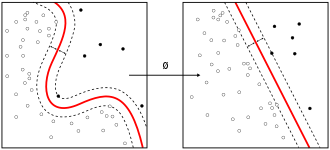
Python comes with an integrated development environment (IDE) called IDLE, which is more beginner based.

Other shells, including IDLE and IPython, add further abilities such as improved auto-completion, session state retention and syntax highlighting.

As well as standard desktop integrated development environments, there are Web browser-based IDEs; SageMath (intended for developing science and math-related Python programs); PythonAnywhere, a browser-based IDE and hosting environment; and Canopy IDE, a commercial Python IDE emphasizing scientific computing.

**Machine Learning**

Machine learning (ML) is the study of computer algorithms that can improve automatically through experience and by the use of data. It is seen as a part of artificial intelligence. Machine learning algorithms build a model based on sample data, known as training data, in order to make predictions or decisions without being explicitly programmed to do so. Machine learning algorithms are used in a wide variety of applications, such as in medicine, email filtering, speech recognition, and computer vision, where it is difficult or unfeasible to develop conventional algorithms to perform the needed tasks.



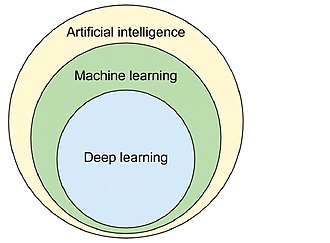
A subset of machine learning is closely related to computational statistics, which focuses on making predictions using computers; but not all machine learning is statistical learning. The study of mathematical optimization delivers methods, theory and application domains to the field of machine learning. Data mining is a related field of study, focusing on exploratory data analysis through unsupervised learning. Some implementations of machine learning use data and neural networks in a way that mimics the working of a biological brain. In its application across business problems, machine learning is also referred to as predictive analytics.

## History and relationships to other fields

The term machine learning was coined in 1959 by Arthur Samuel, an American IBMer and pioneer in the field of computer gaming and artificial intelligence. Also, the synonym self-teaching computers was used in this time period. A representative book of the machine learning research during the 1960s was the Nilsson's book on Learning Machines, dealing mostly with machine learning for pattern classification. Interest related to pattern recognition continued into the 1970s, as described by Duda and Hart in 1973. In 1981 a report was given on using teaching strategies so that a neural network learns to recognize 40 characters (26 letters, 10 digits, and 4 special symbols) from a computer terminal.

### Artificial intelligence

Part of machine learning as subfield of AI or part of AI as subfield of machine learning



As a scientific endeavour, machine learning grew out of the quest for artificial intelligence. In the early days of AI as an academic discipline, some researchers were interested in having machines learn from data. They attempted to approach the problem with various symbolic methods, as well as what was then termed "neural networks"; these were mostly perceptrons and other models that were later found to be reinventions of the generalized linear models of statistics.[23] Probabilistic reasoning was also employed, especially in automated medical diagnosis.

Machine learning (ML), reorganized as a separate field, started to flourish in the 1990s. The field changed its goal from achieving artificial intelligence to tackling solvable problems of a practical nature. It shifted focus away from the symbolic approaches it had inherited from AI, and toward methods and models borrowed from statistics and probability theory.

### Data mining

Machine learning and data mining often employ the same methods and overlap significantly, but while machine learning focuses on prediction, based on known properties learned from the training data, data mining focuses on the discovery of (previously) unknown properties in the data (this is the analysis step of knowledge discovery in databases). Data mining uses many machine learning methods, but with different goals; on the other hand, machine learning also employs data mining methods as "unsupervised learning" or as a pre-processing step to improve learner accuracy.

### Optimization

Machine learning also has intimate ties to optimization: many learning problems are formulated as minimization of some loss function on a training set of examples. Loss functions express the discrepancy between the predictions of the model being trained and the actual problem instances (for example, in classification, one wants to assign a label to instances, and models are trained to correctly predict the pre-assigned labels of a set of examples).

### Statistics

Machine learning and statistics are closely related fields in terms of methods, but distinct in their principal goal: statistics draws population inferences from a sample, while machine learning finds generalizable predictive patterns. According to Michael I. Jordan, the ideas of machine learning, from methodological principles to theoretical tools, have had a long pre-history in statistics. He also suggested the term data science as a placeholder to call the overall field.

TYPES OF MACHINE LEARNING:

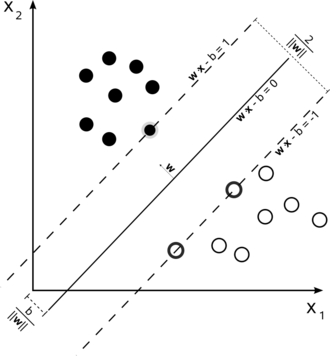
Machine learning approaches are traditionally divided into three broad categories, depending on the nature of the "signal" or "feedback" available to the learning system:

1. Supervised learning: The computer is presented with example inputs and their desired outputs, given by a "teacher", and the goal is to learn a general rule that maps inputs to outputs.
2. Unsupervised learning: No labels are given to the learning algorithm, leaving it on its own to find structure in its input. Unsupervised learning can be a goal in itself (discovering hidden patterns in data) or a means towards an end (feature learning).
3. Reinforcement learning: A computer program interacts with a dynamic environment in which it must perform a certain goal (such as driving a vehicle or playing a game against an opponent). As it navigates its problem space, the program is provided feedback that's analogous to rewards, which it tries to maximize.

### Supervised learning

Supervised learning algorithms build a mathematical model of a set of data that contains both the inputs and the desired outputs. The data is known as training data, and consists of a set of training examples. Each training example has one or more inputs and the desired output, also known as a supervisory signal. In the mathematical model, each training example is represented by an array or vector, sometimes called a feature vector, and the training data is represented by a matrix. Through iterative optimization of an objective function, supervised learning algorithms learn a function that can be used to predict the output associated with new inputs. An optimal function will allow the algorithm to correctly determine the output for inputs that were not a part of the training data. An algorithm that improves the accuracy of its outputs or predictions over time is said to have learned to perform that task.

Similarity learning is an area of supervised machine learning closely related to regression and classification, but the goal is to learn from examples using a similarity function that measures how similar or related two objects are. It has applications in ranking, recommendation systems, visual identity tracking, face verification, and speaker verification.



### Unsupervised learning

Unsupervised learning algorithms take a set of data that contains only inputs, and find structure in the data, like grouping or clustering of data points. The algorithms, therefore, learn from test data that has not been labeled, classified or categorized. Instead of responding to feedback, unsupervised learning algorithms identify commonalities in the data and react based on the presence or absence of such commonalities in each new piece of data. A central application of unsupervised learning is in the field of density estimation in statistics, such as finding the probability density function. Though unsupervised learning encompasses other domains involving summarizing and explaining data features.

### Semi-supervised learning

Semi-supervised learning falls between unsupervised learning (without any labeled training data) and supervised learning (with completely labeled training data). Some of the training examples are missing training labels, yet many machine-learning researchers have found that unlabeled data, when used in conjunction with a small amount of labeled data, can produce a considerable improvement in learning accuracy.

### Reinforcement learning

Reinforcement learning is an area of machine learning concerned with how software agents ought to take actions in an environment so as to maximize some notion of cumulative reward. Due to its generality, the field is studied in many other disciplines, such as game theory, control theory, operations research, information theory, simulation-based optimization, multi-agent systems, swarm intelligence, statistics and genetic algorithms. In machine learning, the environment is typically represented as a Markov decision process (MDP).

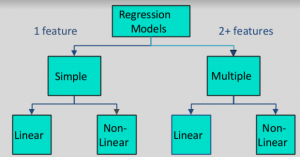
**REGRESSION**

A regression problem is when the output variable is a real or continuous value, such as “salary” or “weight”. Many different models can be used, the simplest is the linear regression. It tries to fit data with the best hyper-plane which goes through the points.

Which of the following is a regression task?

* Predicting age of a person
* Predicting nationality of a person
* Predicting whether stock price of a company will increase tomorrow
* Predicting whether a document is related to sighting of UFOs?

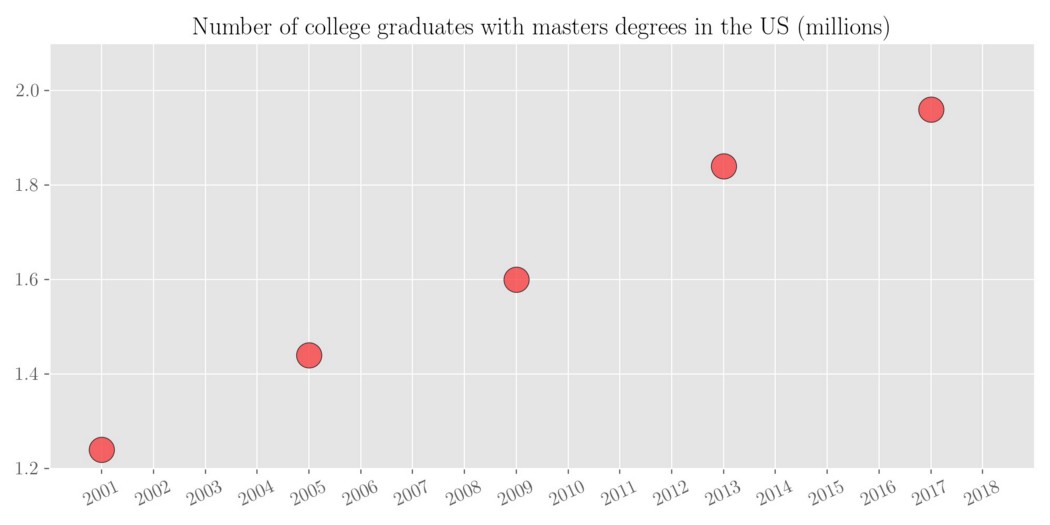
Solution: Predicting age of a person (because it is a real value, predicting nationality is categorical, whether stock price will increase is discrete-yes/no answer, predicting whether a document is related to UFO is again discrete- a yes/no answer).



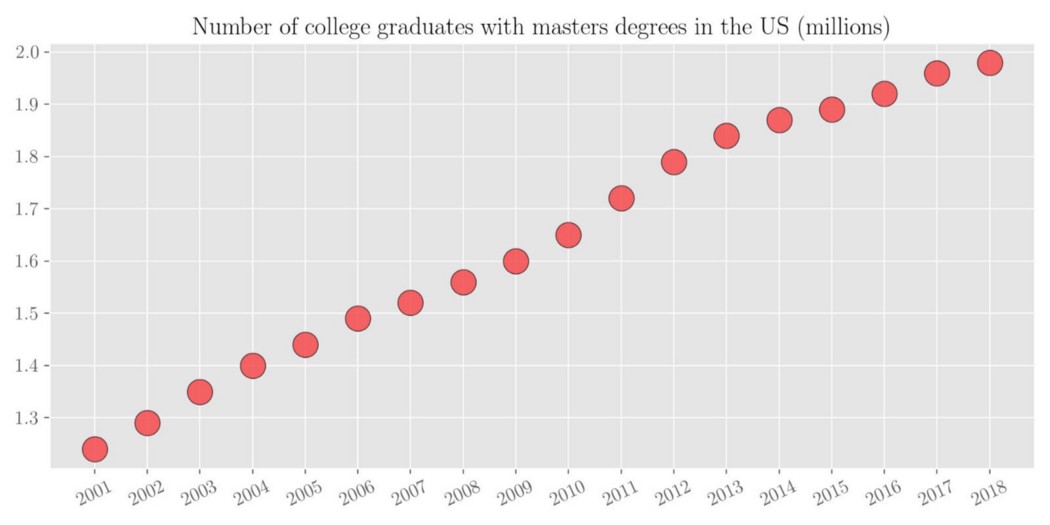
Using regression you fit a function on the available data and try to predict the outcome for the future or hold-out datapoints. This fitting of function serves two purposes.

1. You can estimate missing data within your data range (Interpolation)
2. You can estimate future data outside your data range (Extrapolation)

**Interpolation:**   
Let's assume we have access to somewhat sparse data where we know the number of college graduates every 4 years, as shown in the scatter plot below.



We want to estimate the number of college graduates for all the missing years in between. We can do this by fitting a line to the limited available data points. This process is called interpolation.

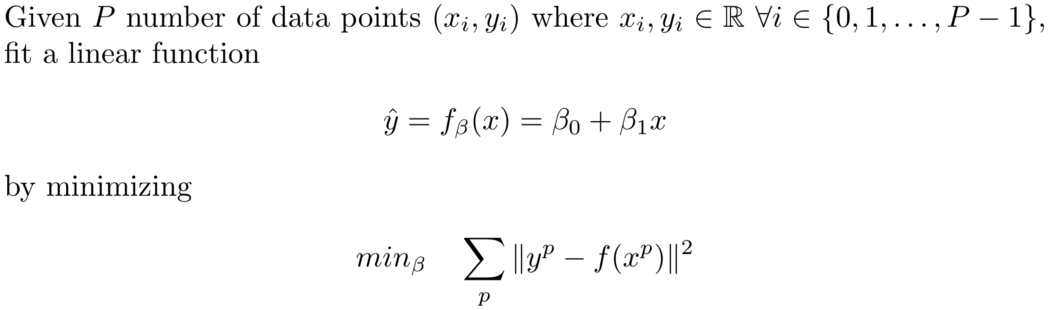


# **Types of regression analysis**

1. Linear Regression

In linear regression, the objective is to fit a hyperplane (a line for 2D data points) by minimizing the sum of mean-squared error for each data point.

Mathematically speaking, linear regression solves the following problem



Hence we need to find 2 variables denoted by beta that parameterize the linear function f(.). An example of linear regression can be seen in the figure 4 above where P=5. The figure also shows the fitted linear function with beta\_0 = -90.798 and beta\_1 = 0.046

## 2. Polynomial Regression

Linear regression assumes that the relationship between the dependant (y) and independent (x) variables are linear. It fails to fit the data points when the relationship between them is not linear. Polynomial regression expands the fitting capabilities of linear regression by fitting a polynomial of degree m to the data points instead.

## 3 Ridge Regression

Ridge regression addresses the issue of overfitting in regression analysis. To understand that, consider the same example as above. When a polynomial of degree 25 is fit on the data with 10 training points, it can be seen that it fits the red data points perfectly (center figure below).

**THE PROJECT**

This is a project which aims at finding the wine quality based on other ingredients and specifications of the wine. For the project we have some prerequisites. These are:

1. An Auto MPG dataset (provided by Antrix Academy)
2. Pandas Library
3. Numpy Library
4. Seaborn library
5. Matplotlib library
6. Sklearn Library

**The Libraries**

*Pandas: Pandas is a software library written for the Python programming language for data manipulation and analysis. In particular, it offers data structures and operations for manipulating numerical tables and time series.*

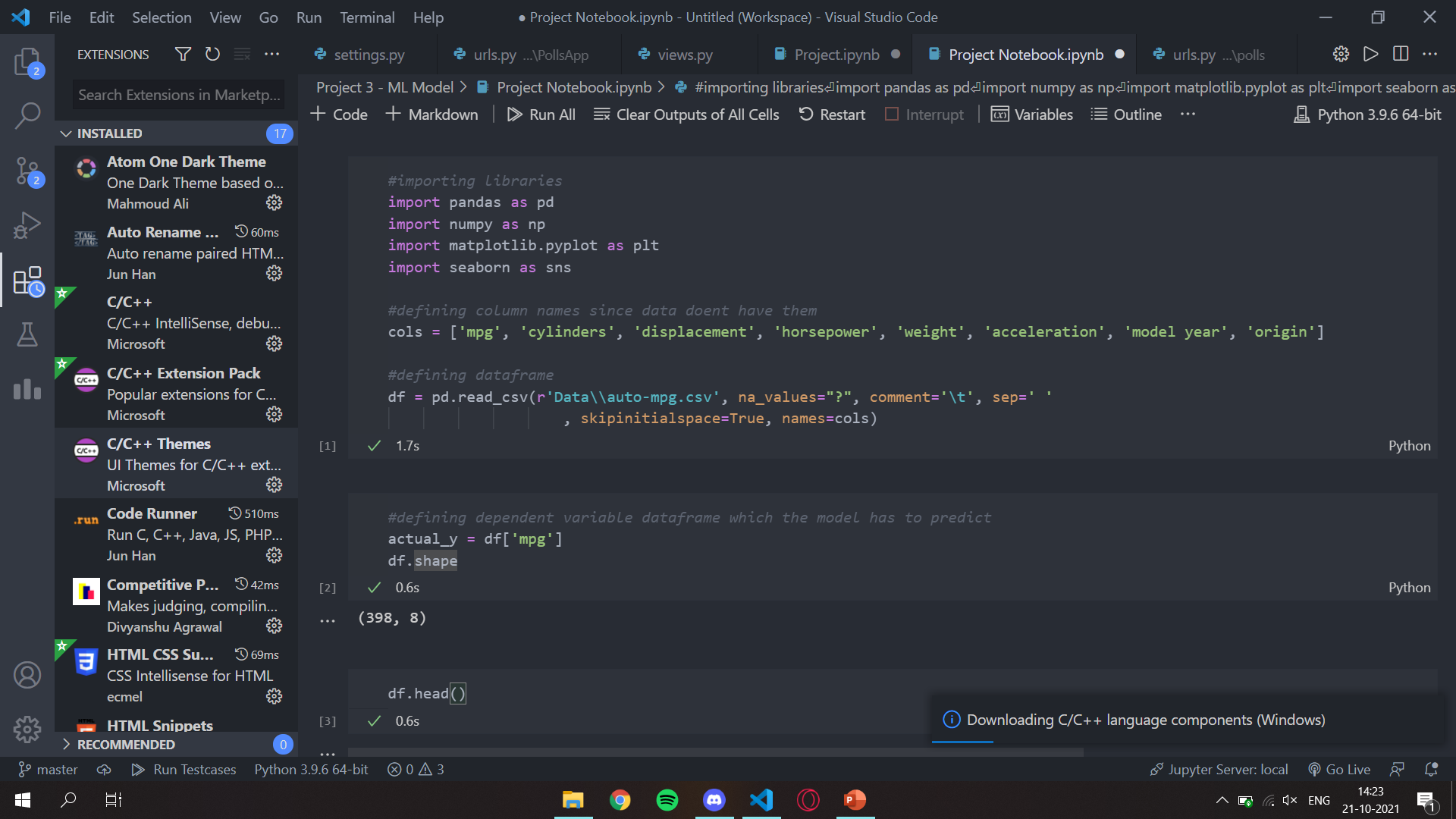
*Numpy: NumPy is a library for the Python programming language, adding support for large, multi-dimensional arrays and matrices, along with a large collection of high-level mathematical functions to operate on these arrays.*

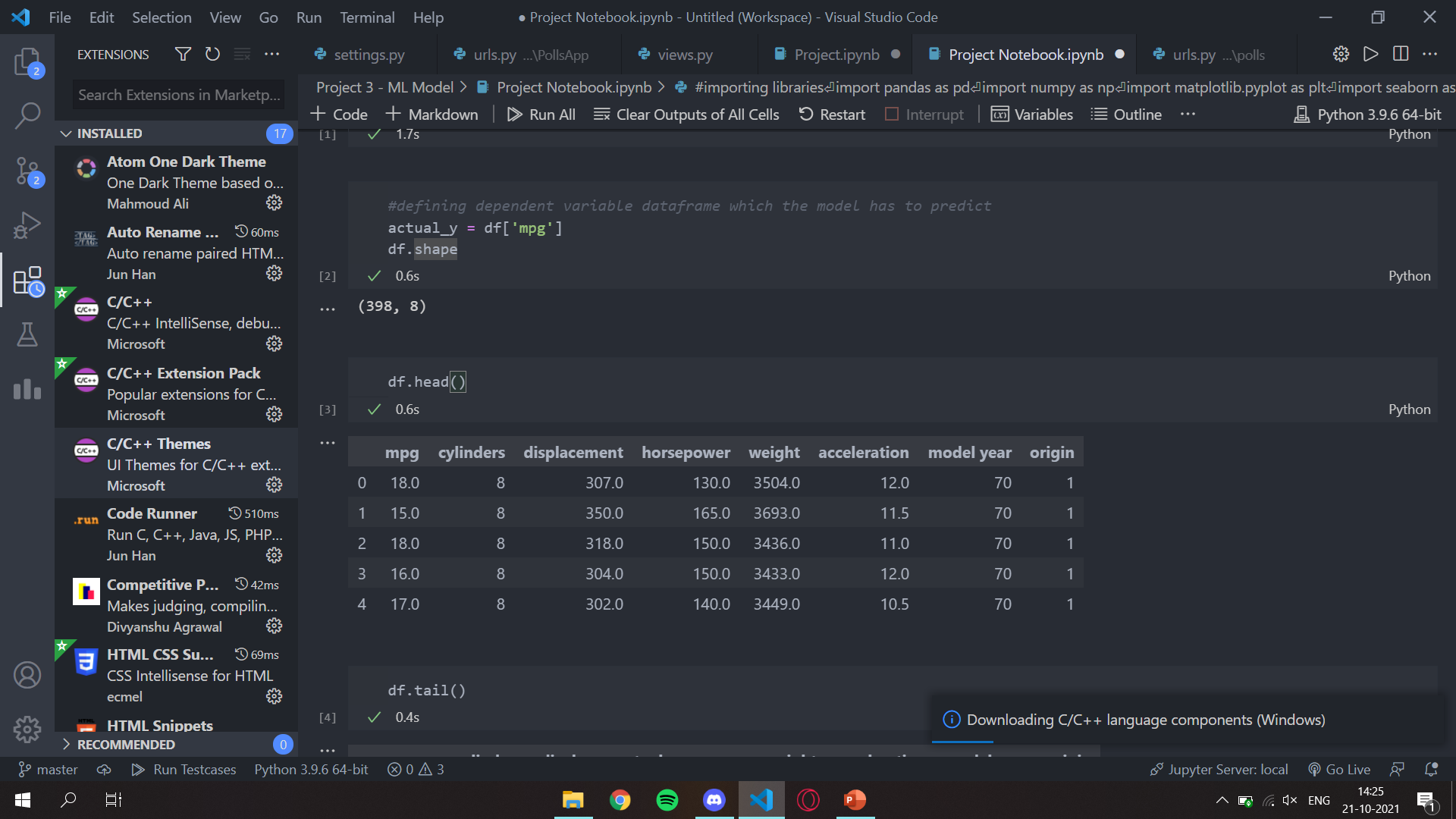
*Seaborn: Seaborn is a data visualization library built on top of matplotlib and closely integrated with pandas data structures in Python.*

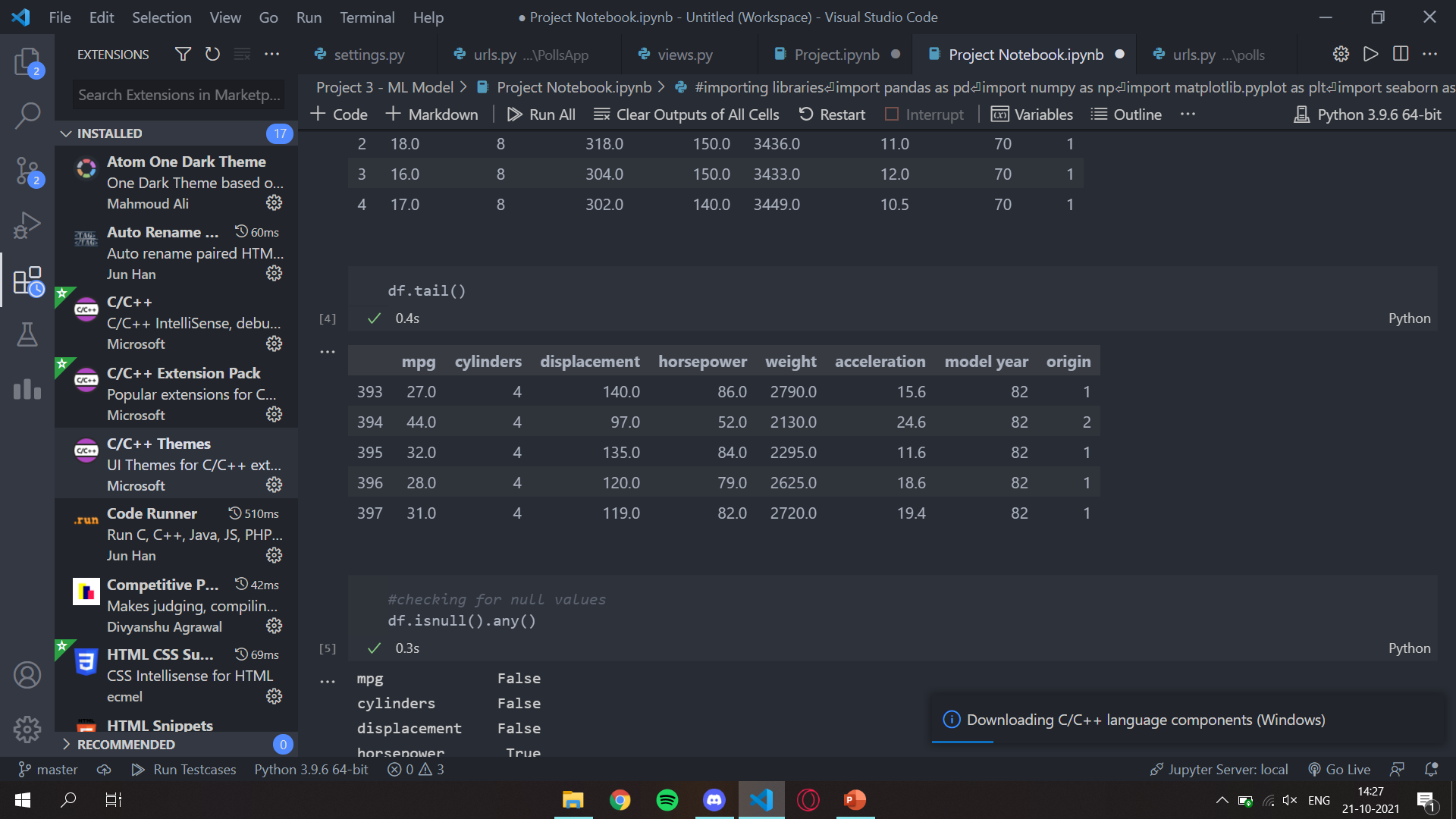
*Matplotlib:* *Matplotlib is a plotting library for the Python programming language and its numerical mathematics extension NumPy. It provides an object-oriented API for embedding plots into applications using general-purpose GUI toolkits like Tkinter, wxPython, Qt, or GTK.*

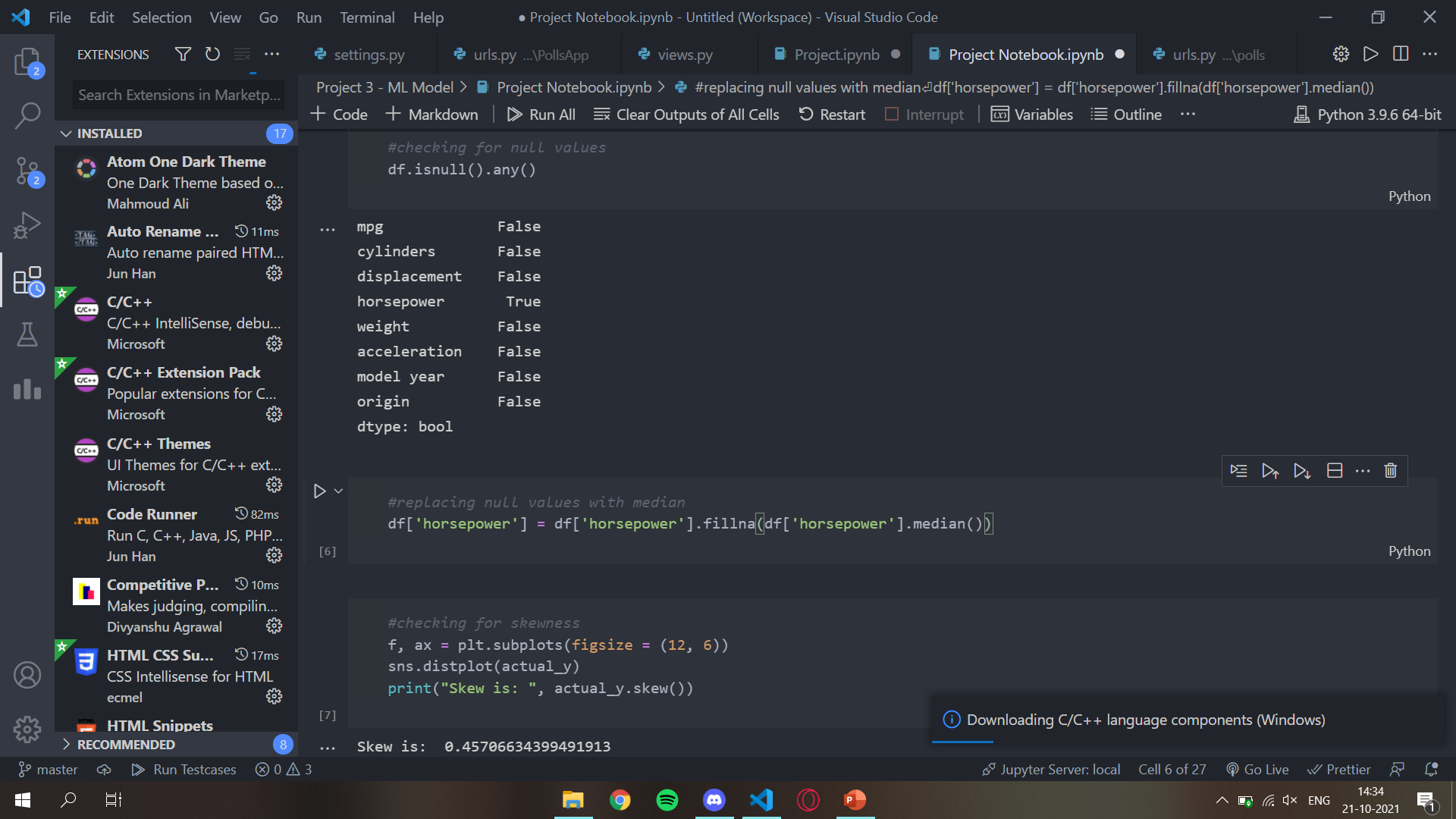
*Sklearn:* *Scikit-learn is a free software machine learning library for the Python programming language.[3] It features various classification, regression and clustering algorithms including support vector machines, random forests, gradient boosting, k-means and DBSCAN, and is designed to interoperate with the Python numerical and scientific libraries NumPy and SciPy.*

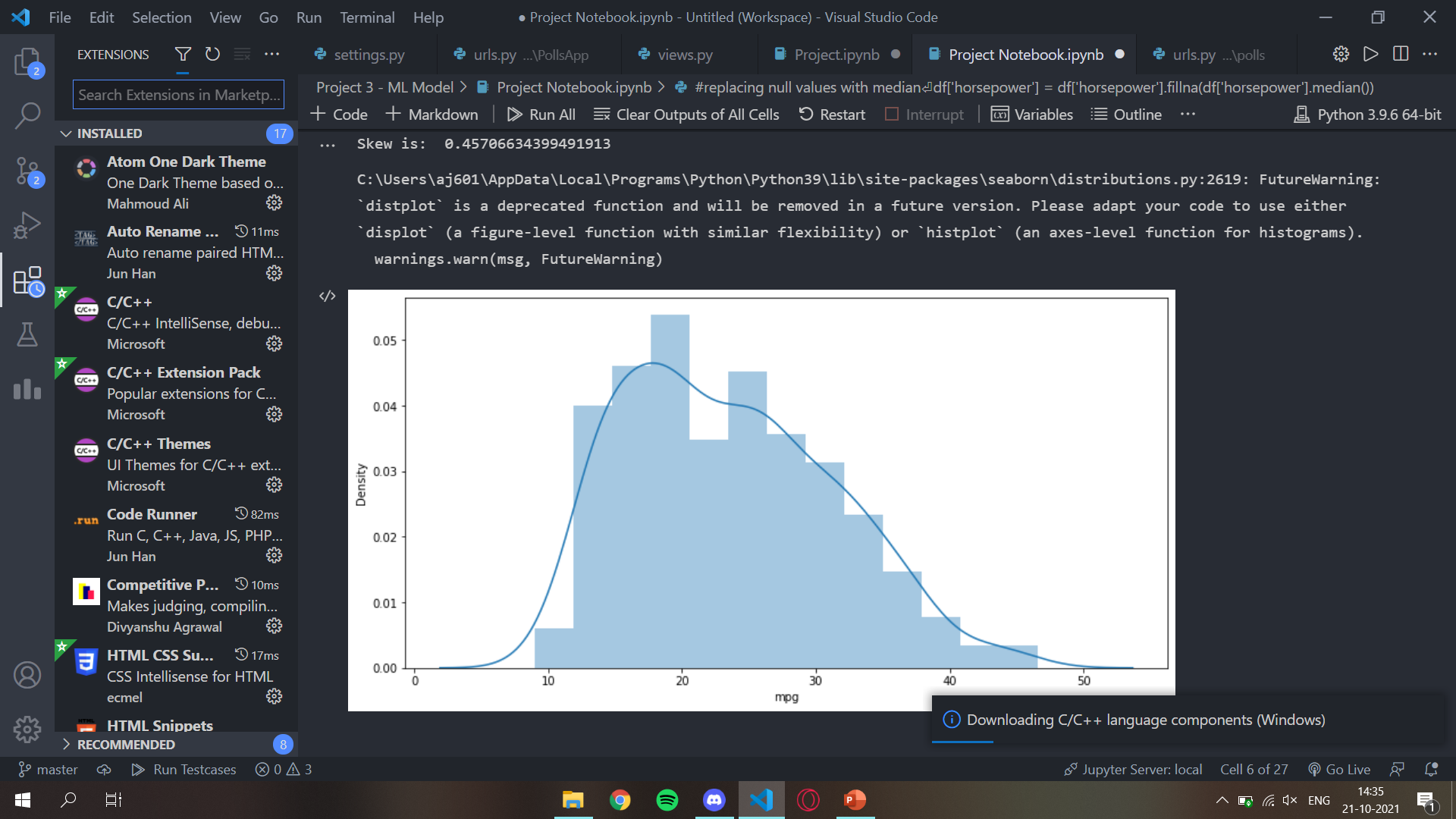
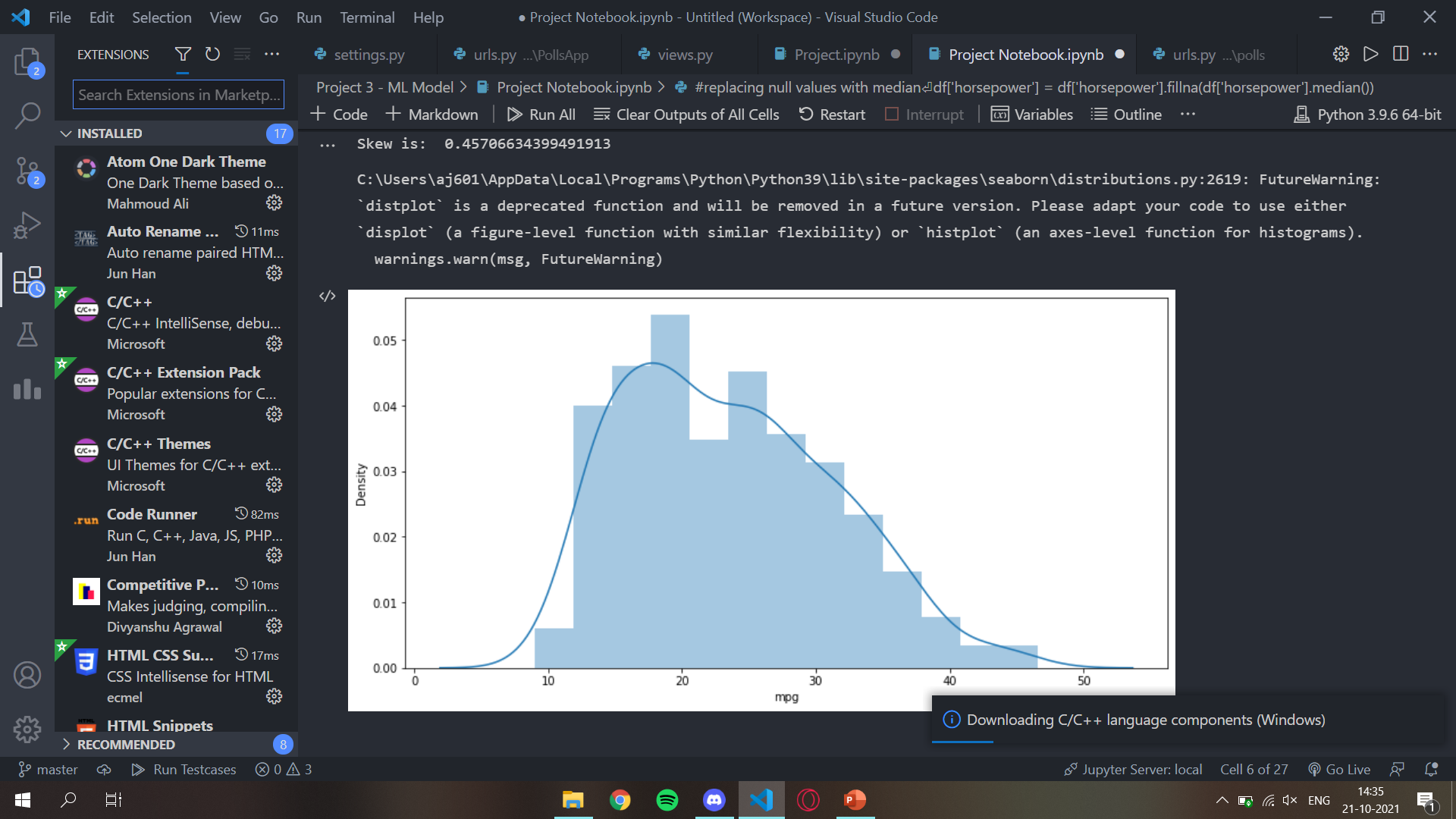
**SOURCE CODE**

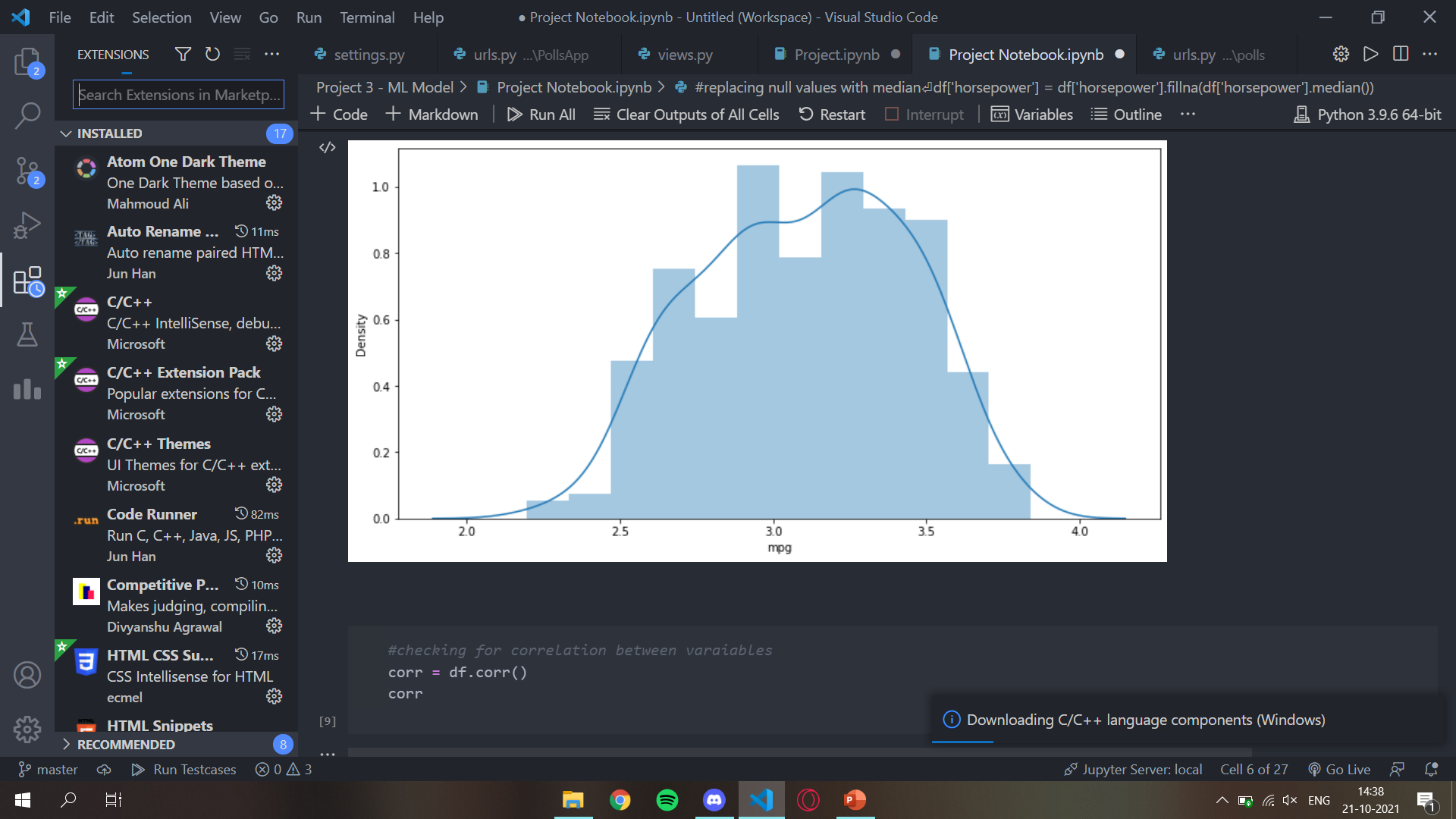
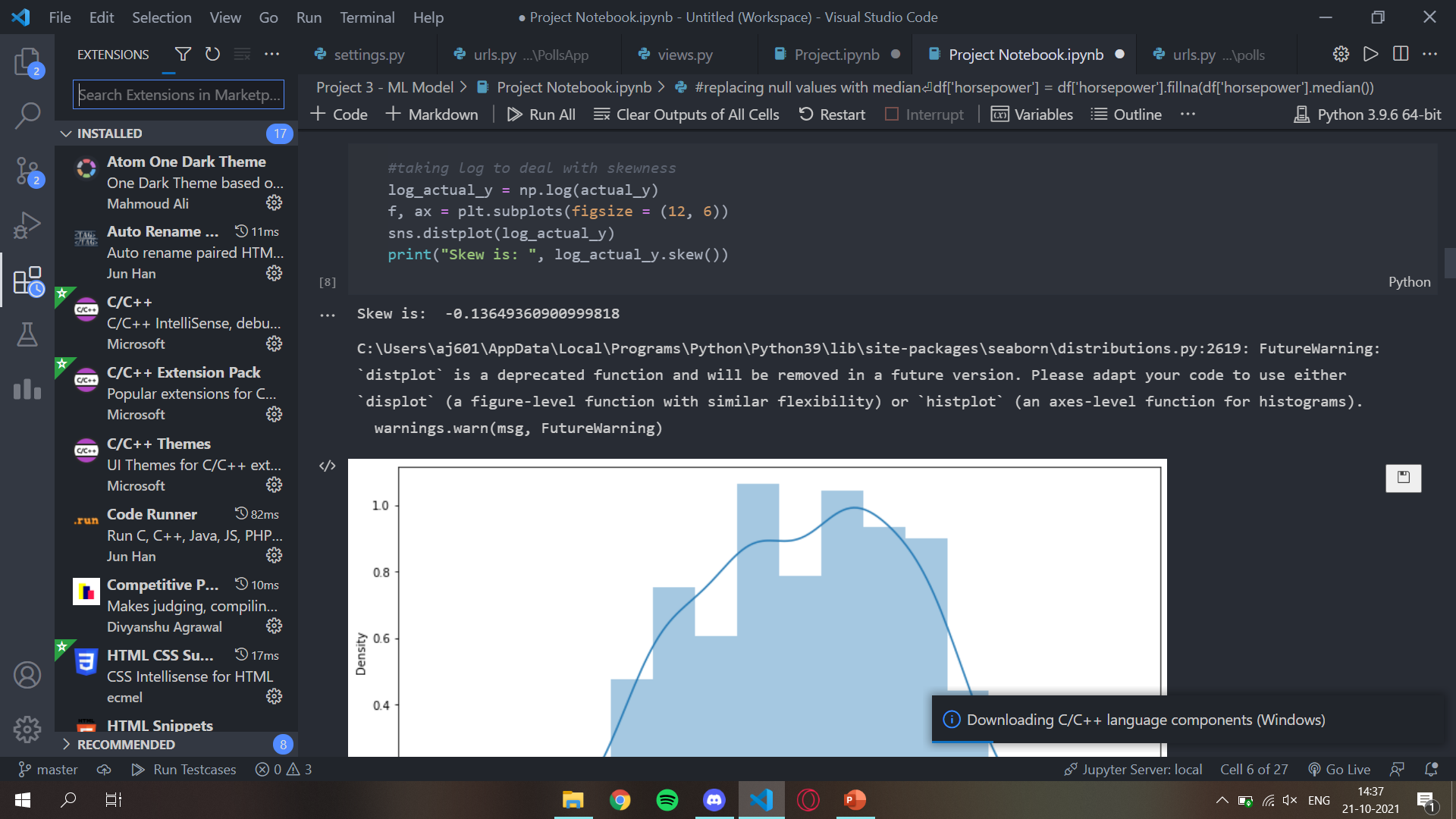




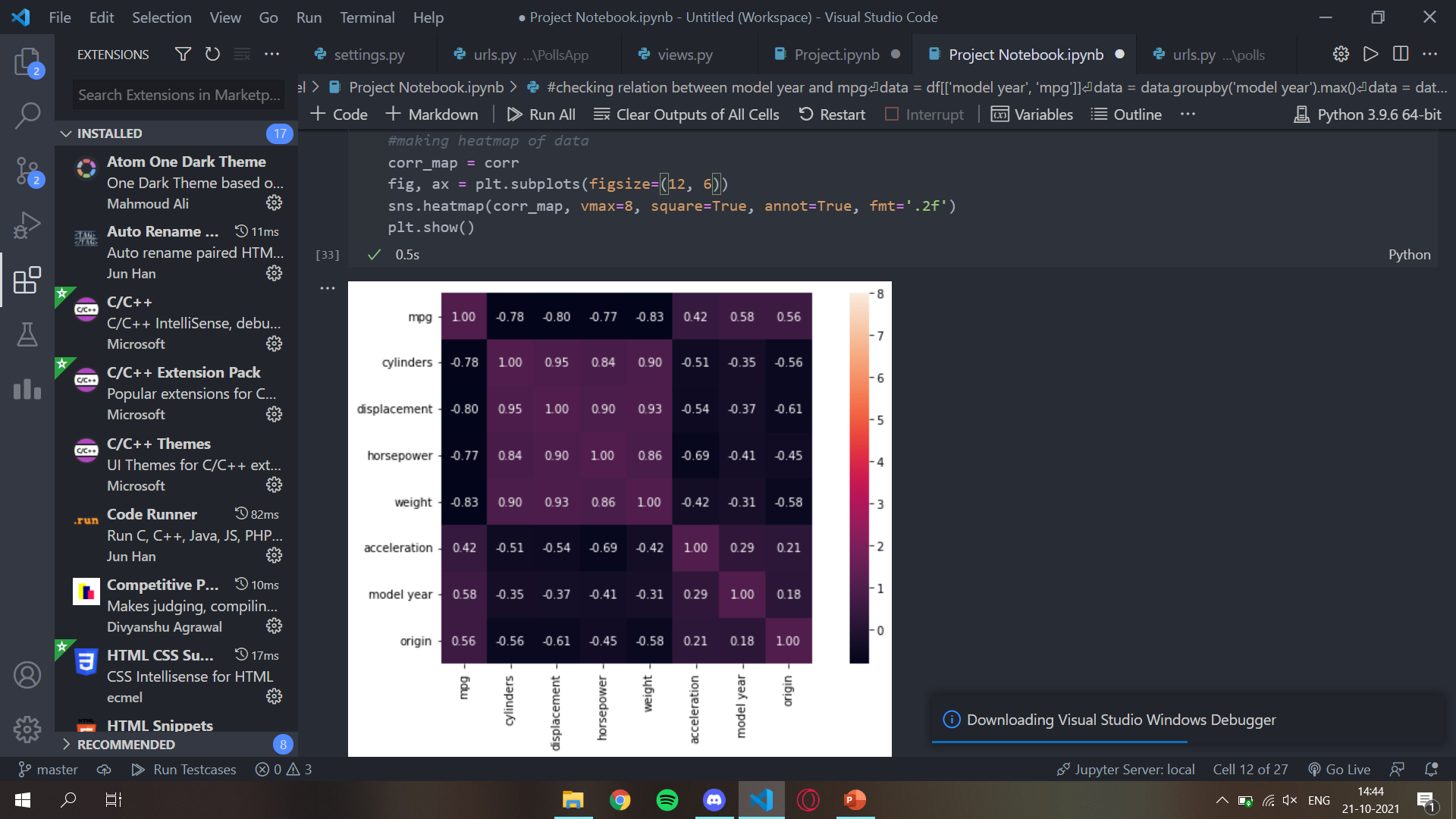


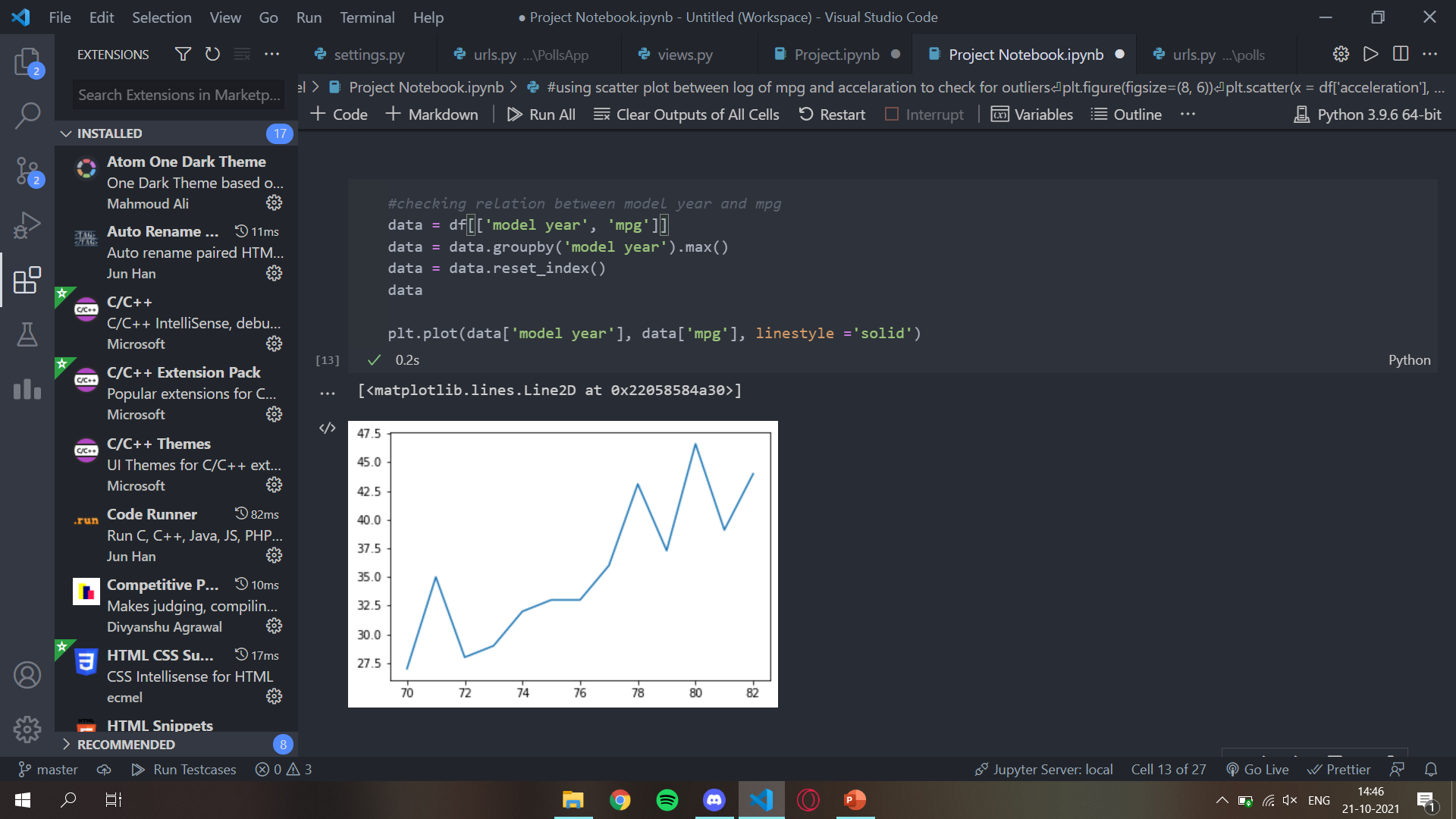


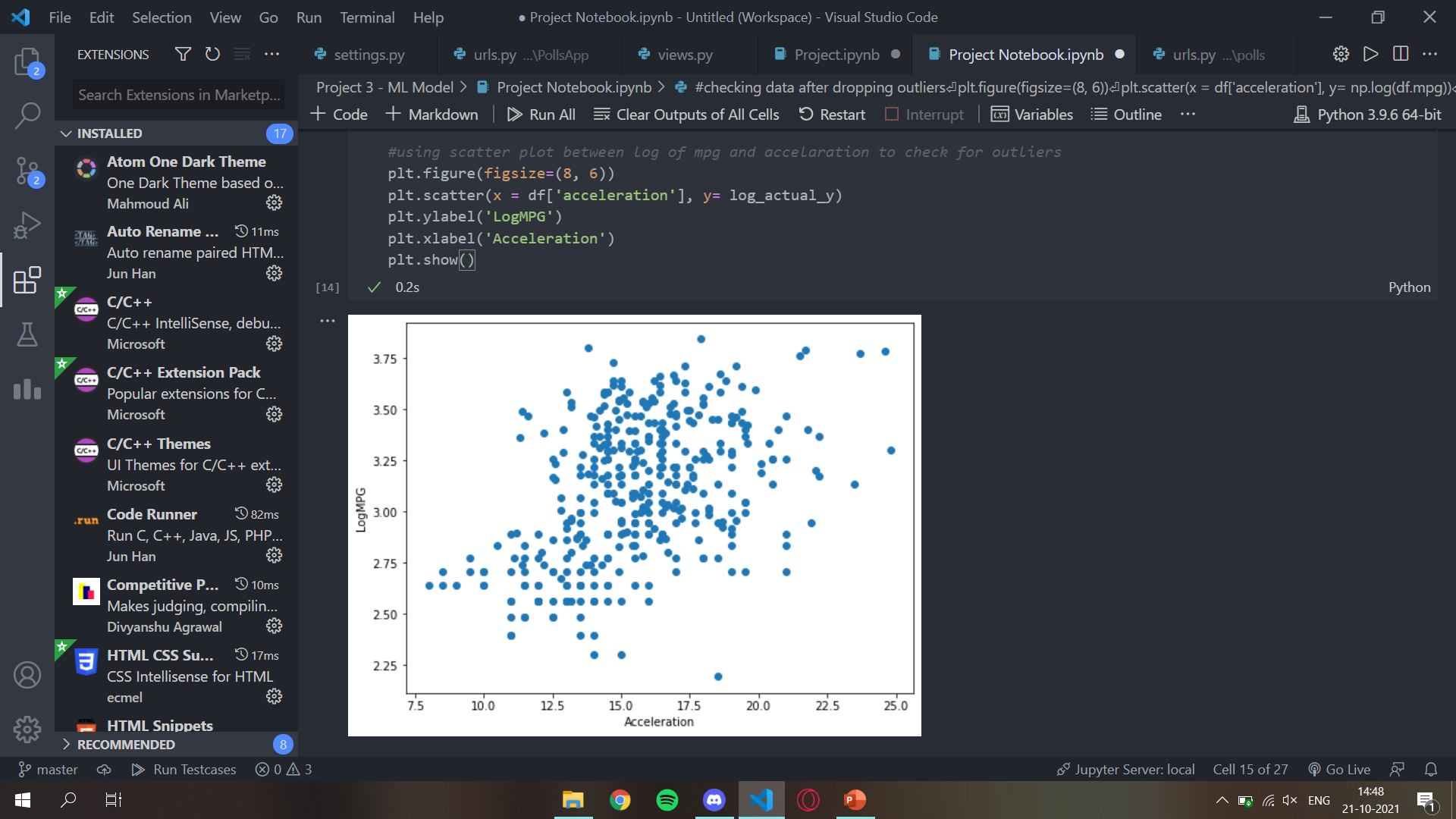


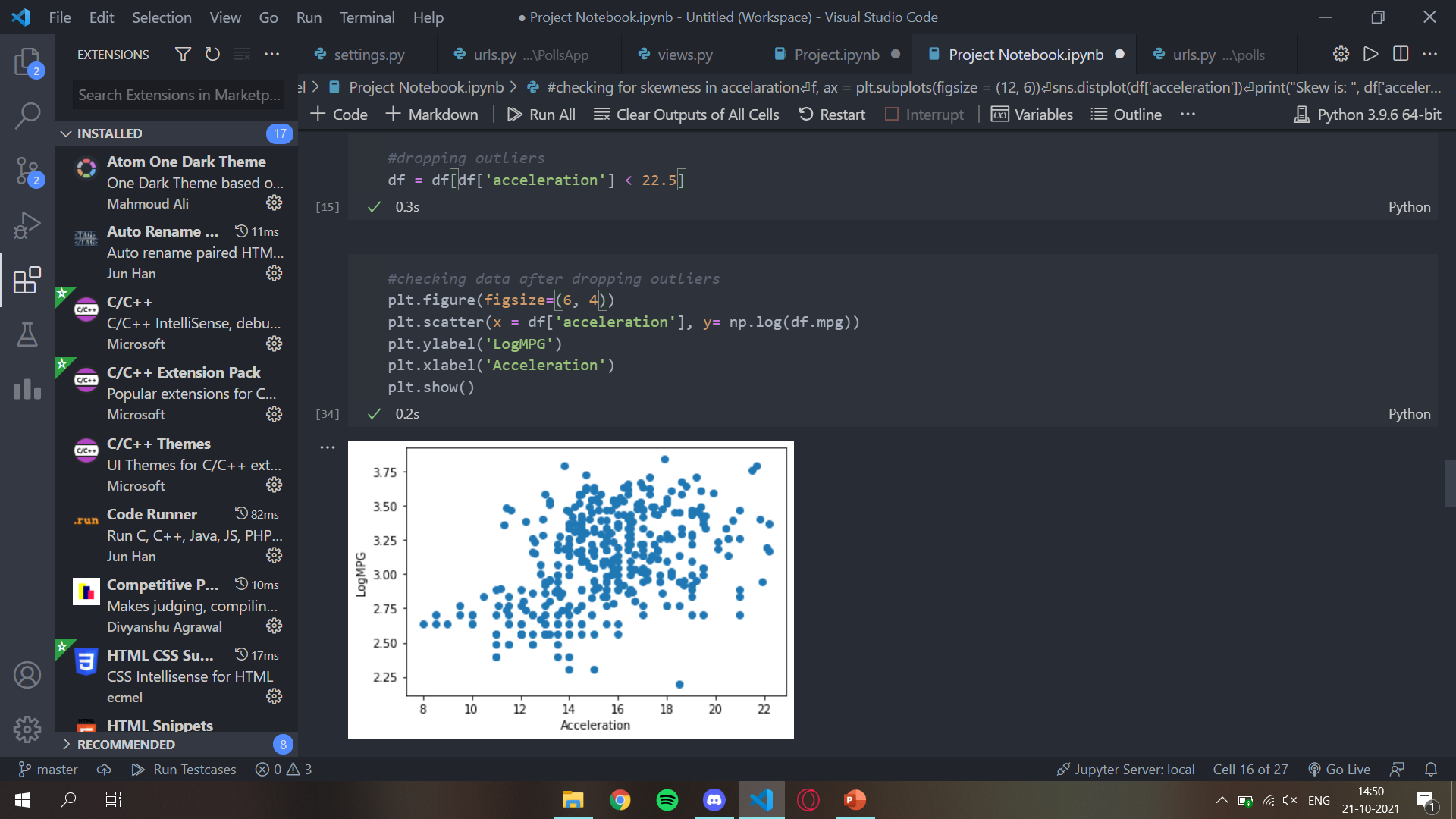


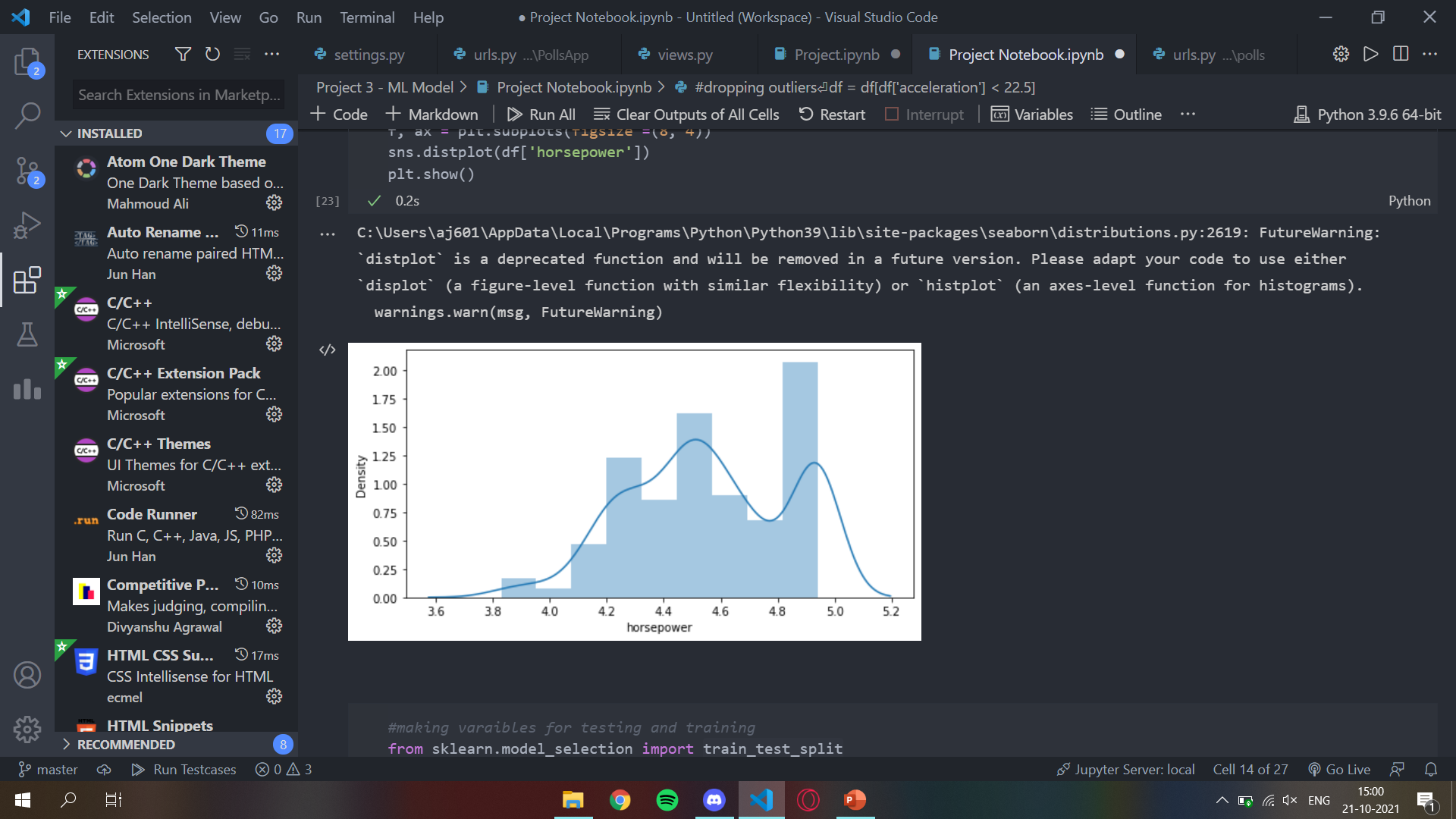
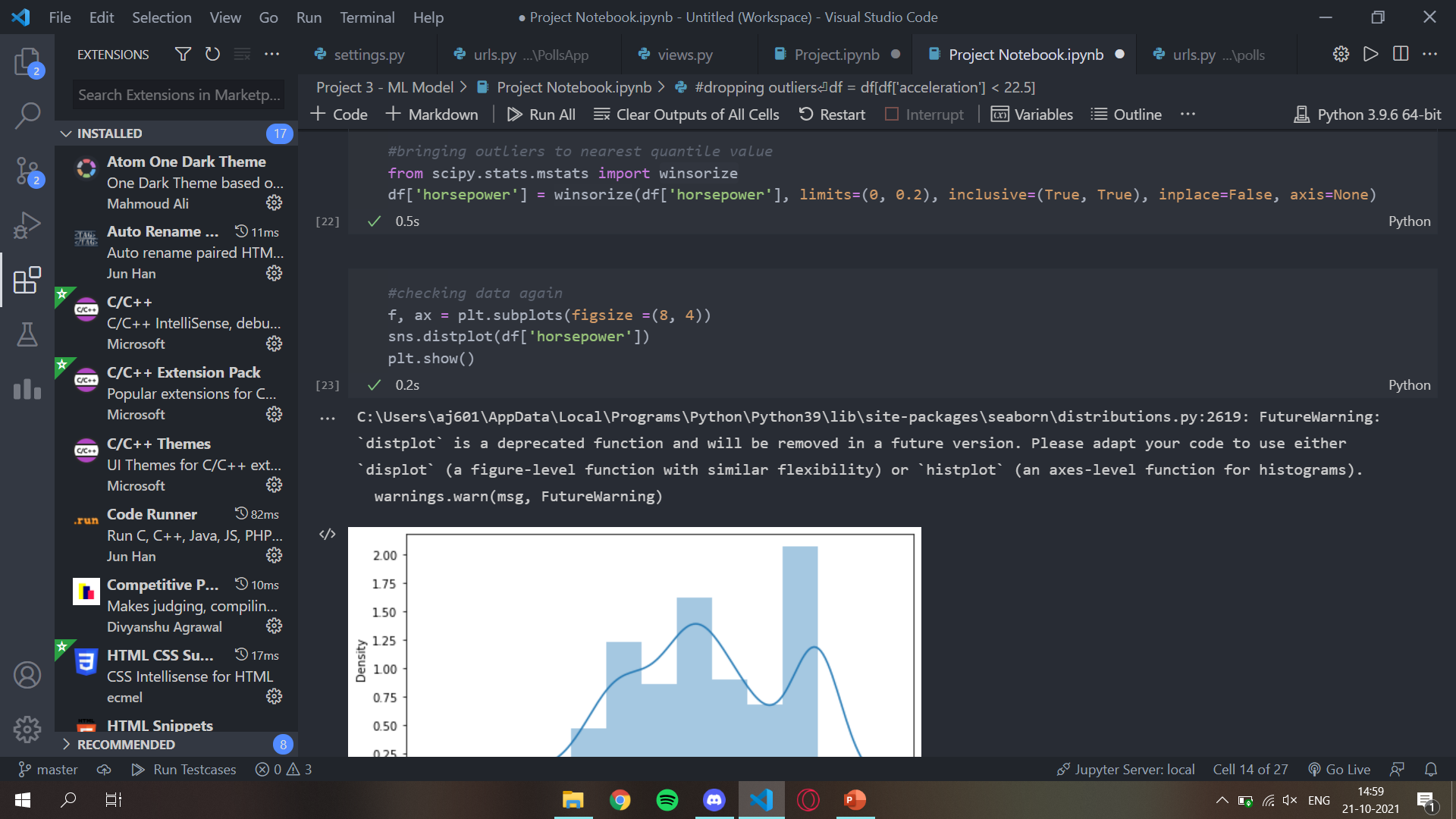


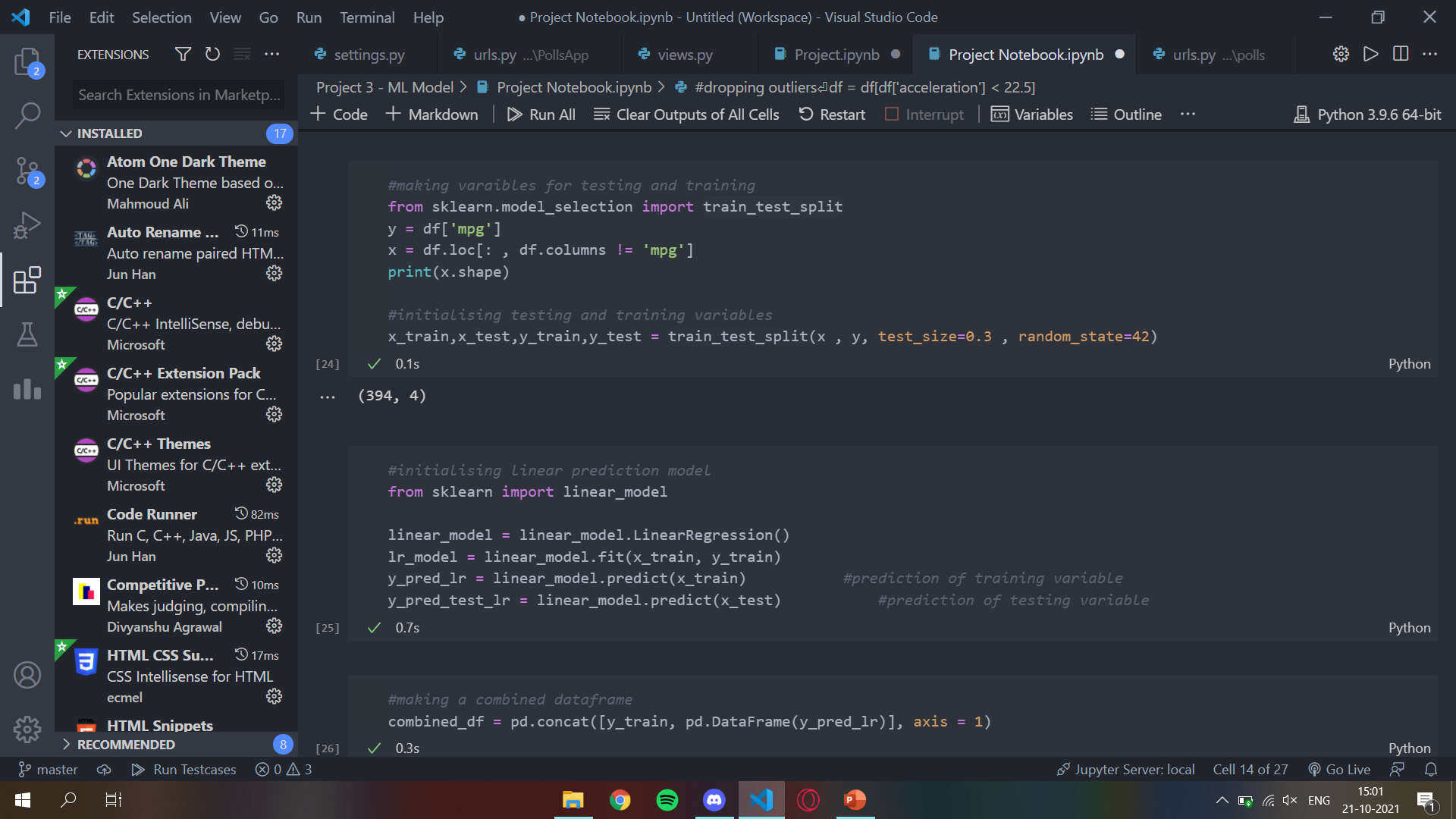


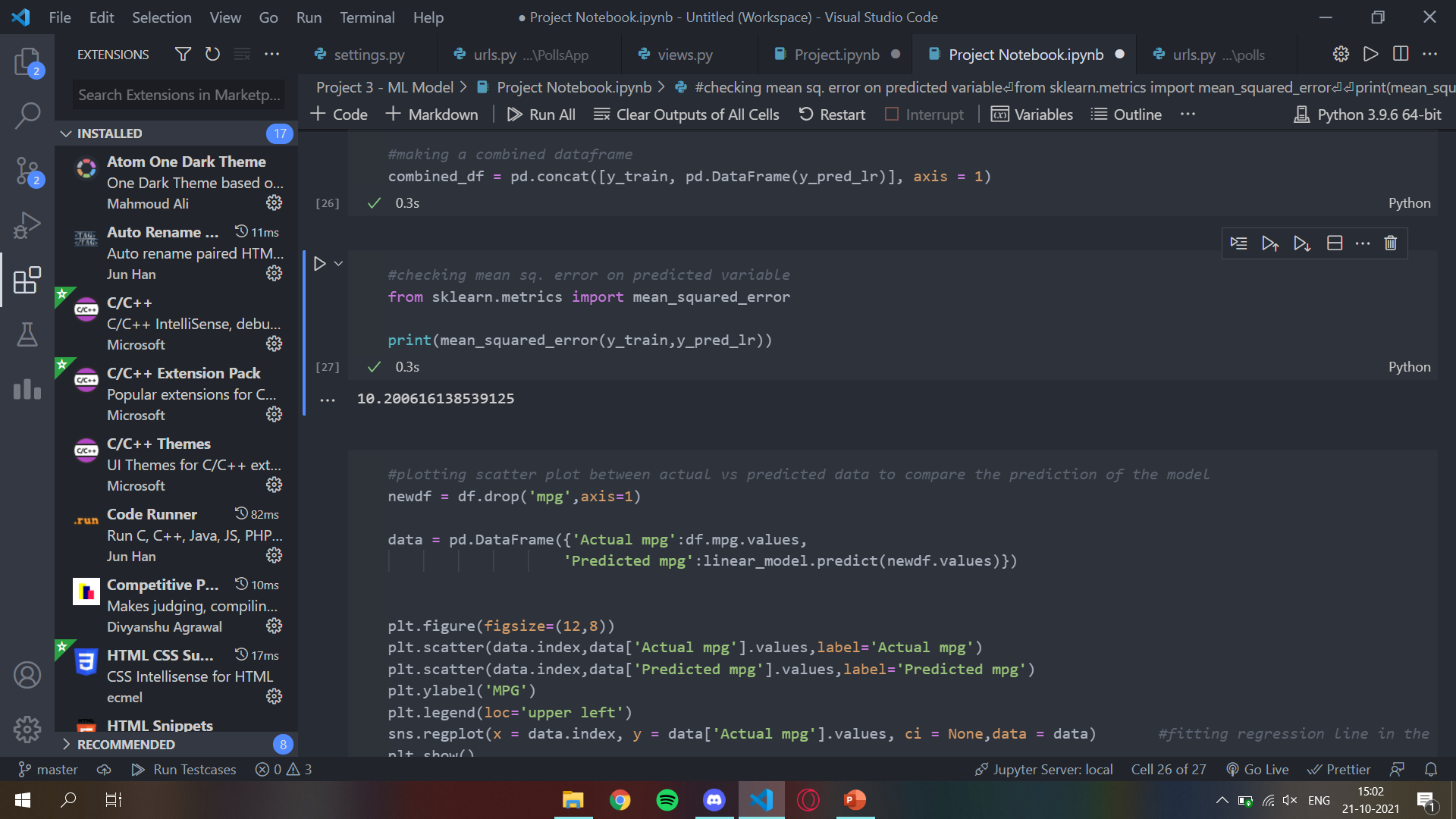


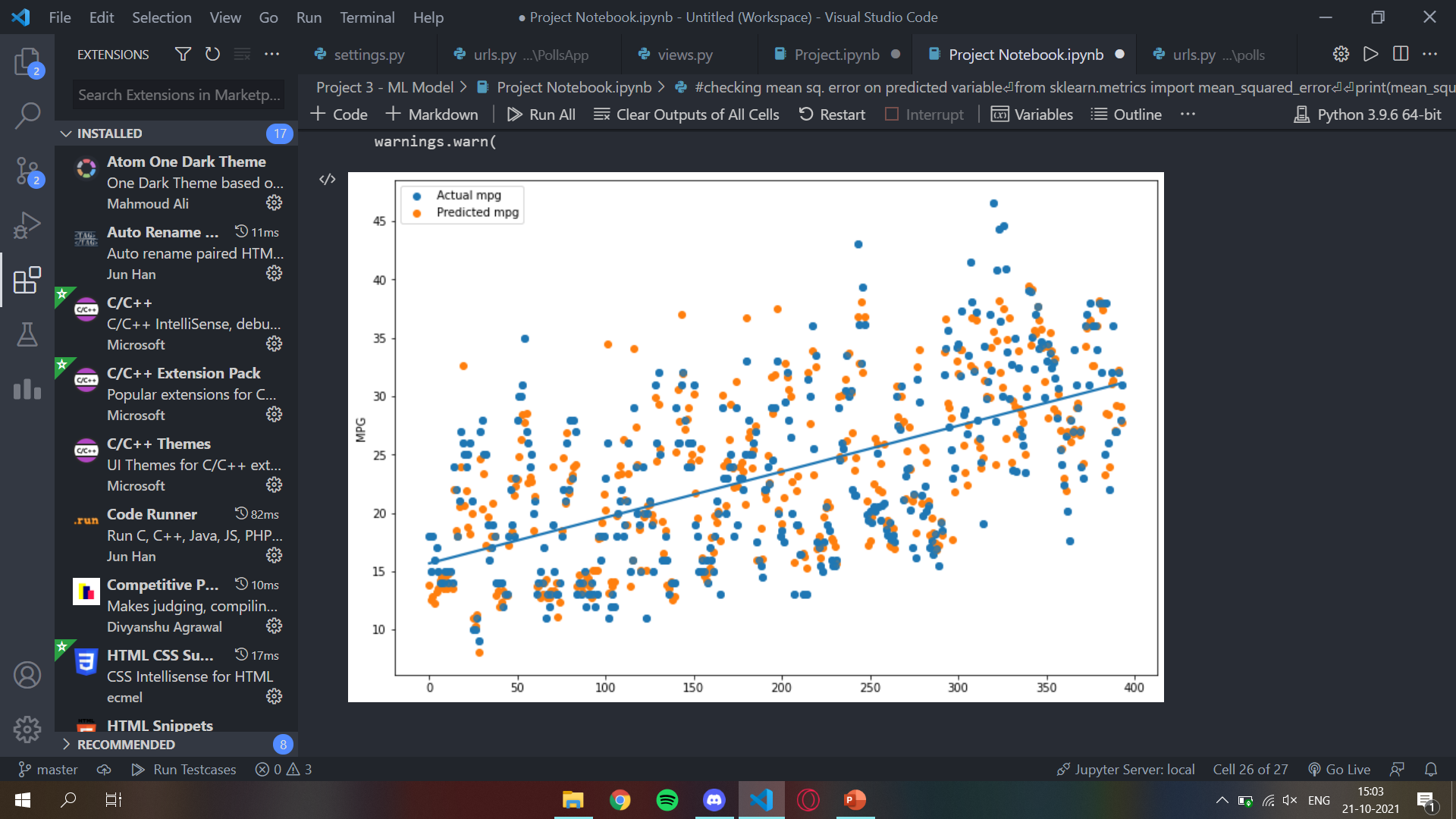












**Advantages of the Project**

1. Easily identifies trends and patterns
2. No human intervention needed (automation)
3. Continuous Improvement
4. Accuracy

**Limitations of the Project**

1. Data dependent/ Data Acquisition
2. High error-susceptibility
3. Time and Resources
4. Complex EDA cases may occur

**CONCLUSION**

Machine learning is quickly growing field in computer science. It has applications in nearly every other field of study and is already being implemented commercially because machine learning can solve problems too difficult or time consuming for humans to solve. Instead of relying on explicit programming, it is a system through which computers use a massive set of data and apply algorithms to "train" on--to teach themselves--and make predictions.

From the above project, we can see that using the scatter plot, we are able to visualise the predictions of the model. As seen, the output is close to the actual results when tested for pre-determined data, but also has some error. This means that there is scope for better predictions using either other models or more EDA. Hence, we can conclude that Machine Learning can be quite useful when it gets to getting accurate results as output on the basis of a series of specifications entered by the user.

REFERENCES

* Video lectures of the internship training programme
* <https://en.wikipedia.org/wiki/Python_(programming_language)>
* [www.w3schools.com/python](http://www.w3schools.com/python)
* <https://towardsdatascience.com/a-beginners-guide-to-regression-analysis-in-machine-learning-8a828b491bbf>
* <https://en.wikipedia.org/wiki/Machine_learning>
* <https://www.geeksforgeeks.org/regression-classification-supervised-machine-learning/>
* <https://data-flair.training/blogs/advantages-and-disadvantages-of-machine-learning/>