**Software Engineering Tools Lab**

**Assignment No-1**

**(Module 1- Introduction to OSS)**

**Batch – T8**

**Name: Sayali Yogesh Desai**

**PRN: 2020BTECS00206**

**1. Weka is a GUI workbench that empowers data wranglers to assemble machine learning pipelines, train models, and run predictions without having to write code. Using Weka tool perform below tasks such as data pre-processing, data classification (use any appropriate ML algorithm) and data visualization efficiently on given dataset.**

**Use the Iris dataset given**

[**https://drive.google.com/file/d/1A3Fxsfzm6BSfhFZGDrjI47RTe45bSgYP/view**](https://drive.google.com/file/d/1A3Fxsfzm6BSfhFZGDrjI47RTe45bSgYP/view)

**Note-provide screen shots for every task**

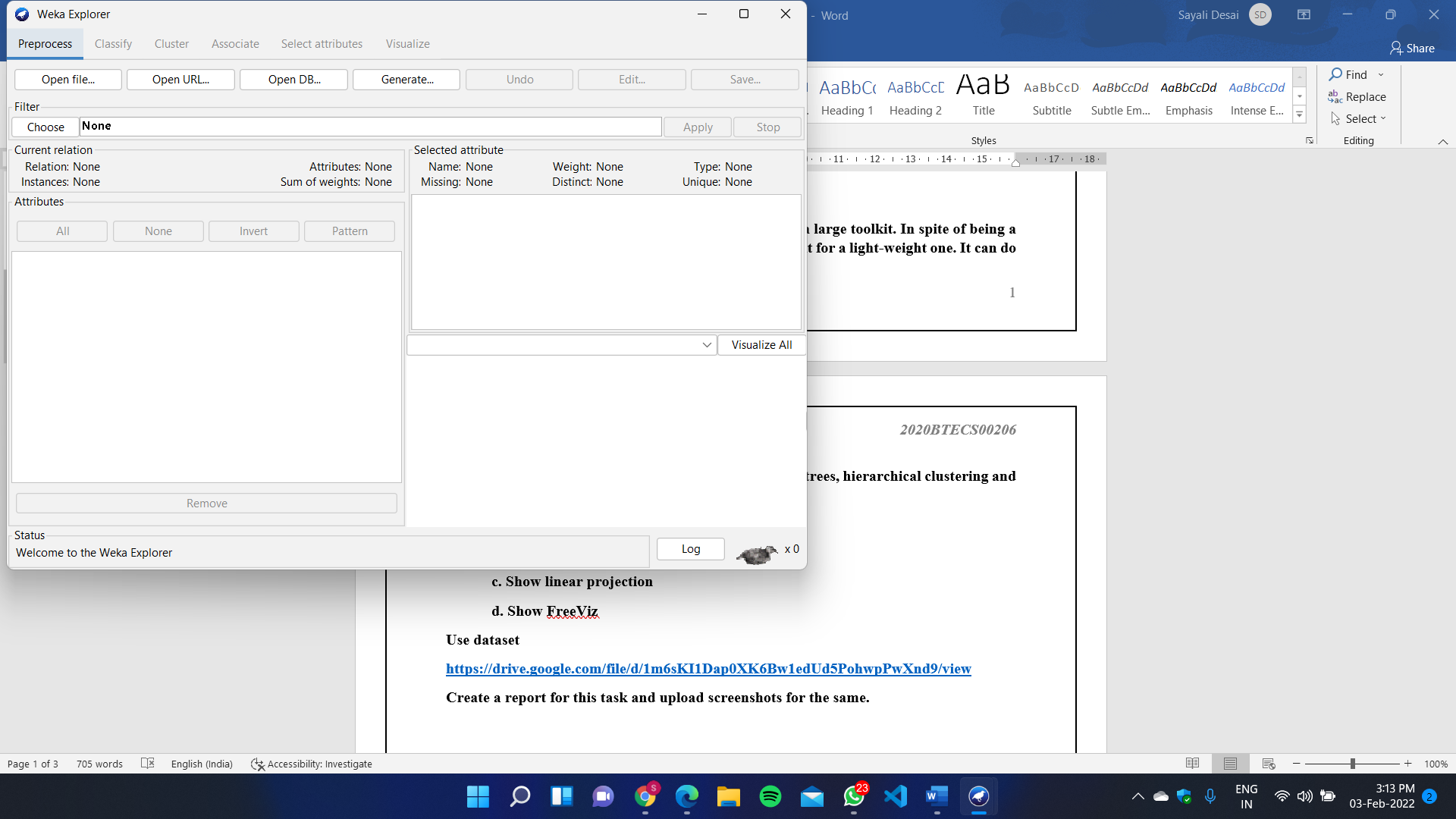
**Create a report which will illustrate the details of tasks performed**

**(For e.g to perform pre-processing of data provide details of navigation and selection of appropriate parameters)**

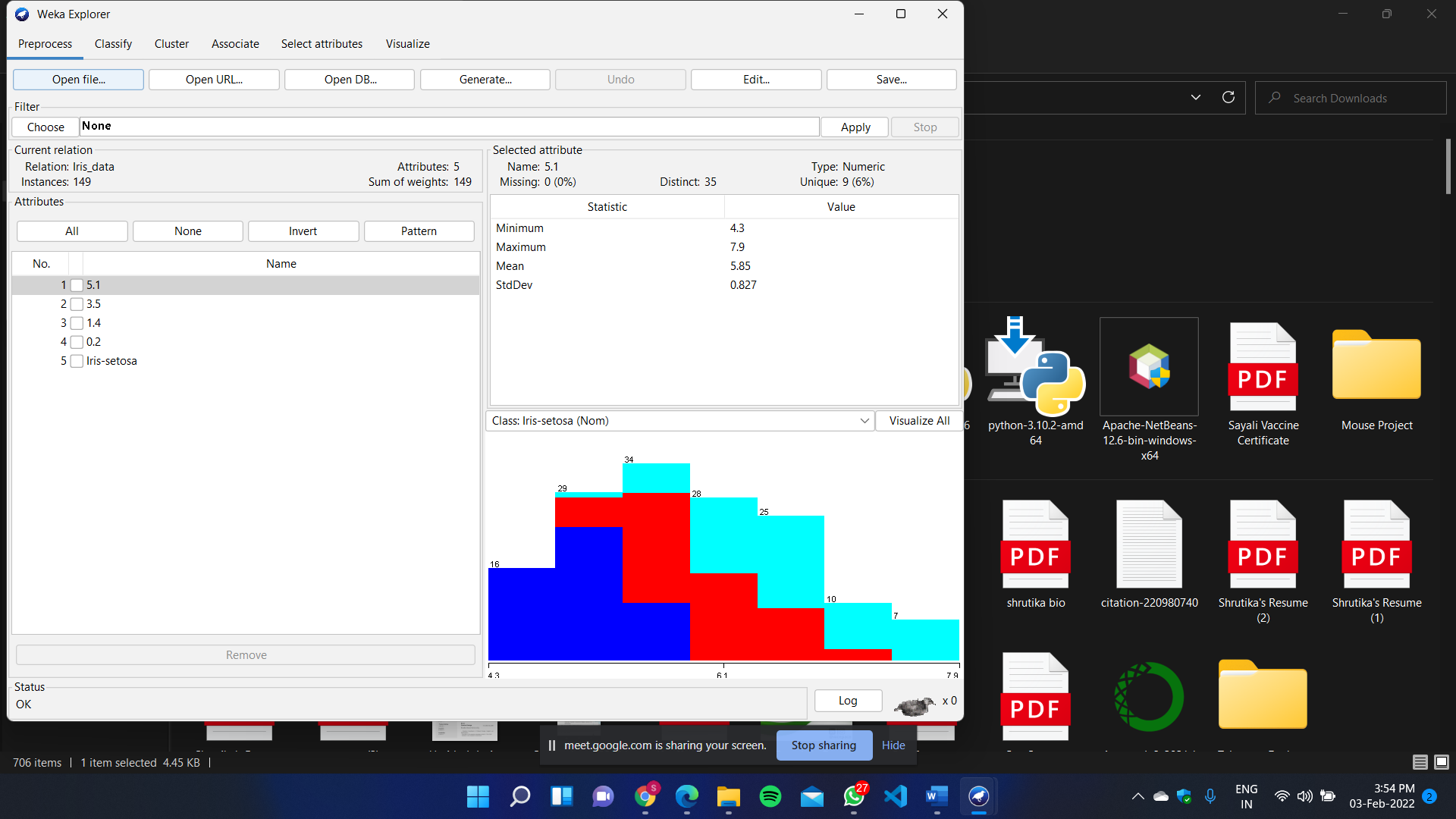
* Open Weka



* Select Explorer

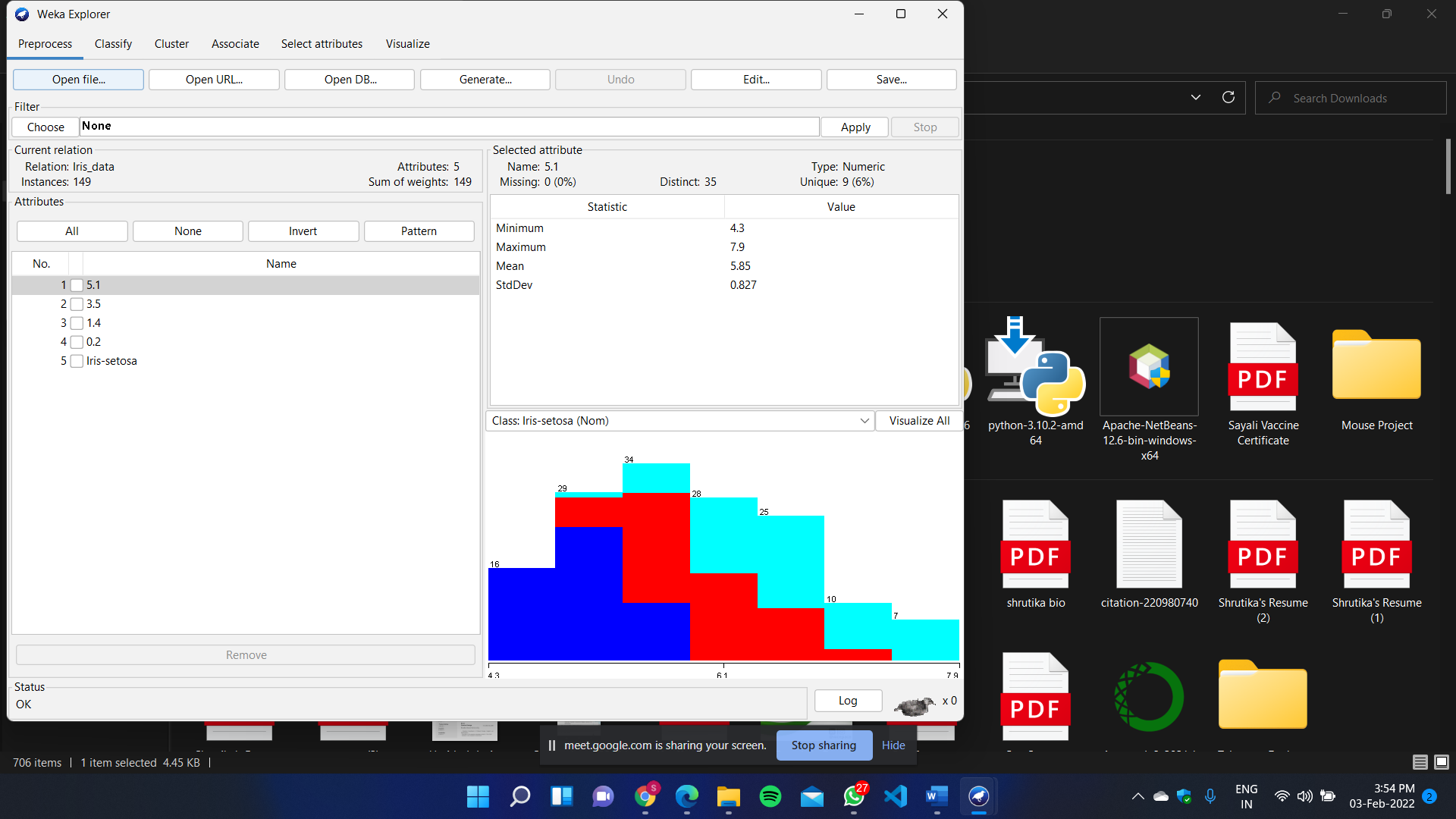


* Open iris\_data.csv file

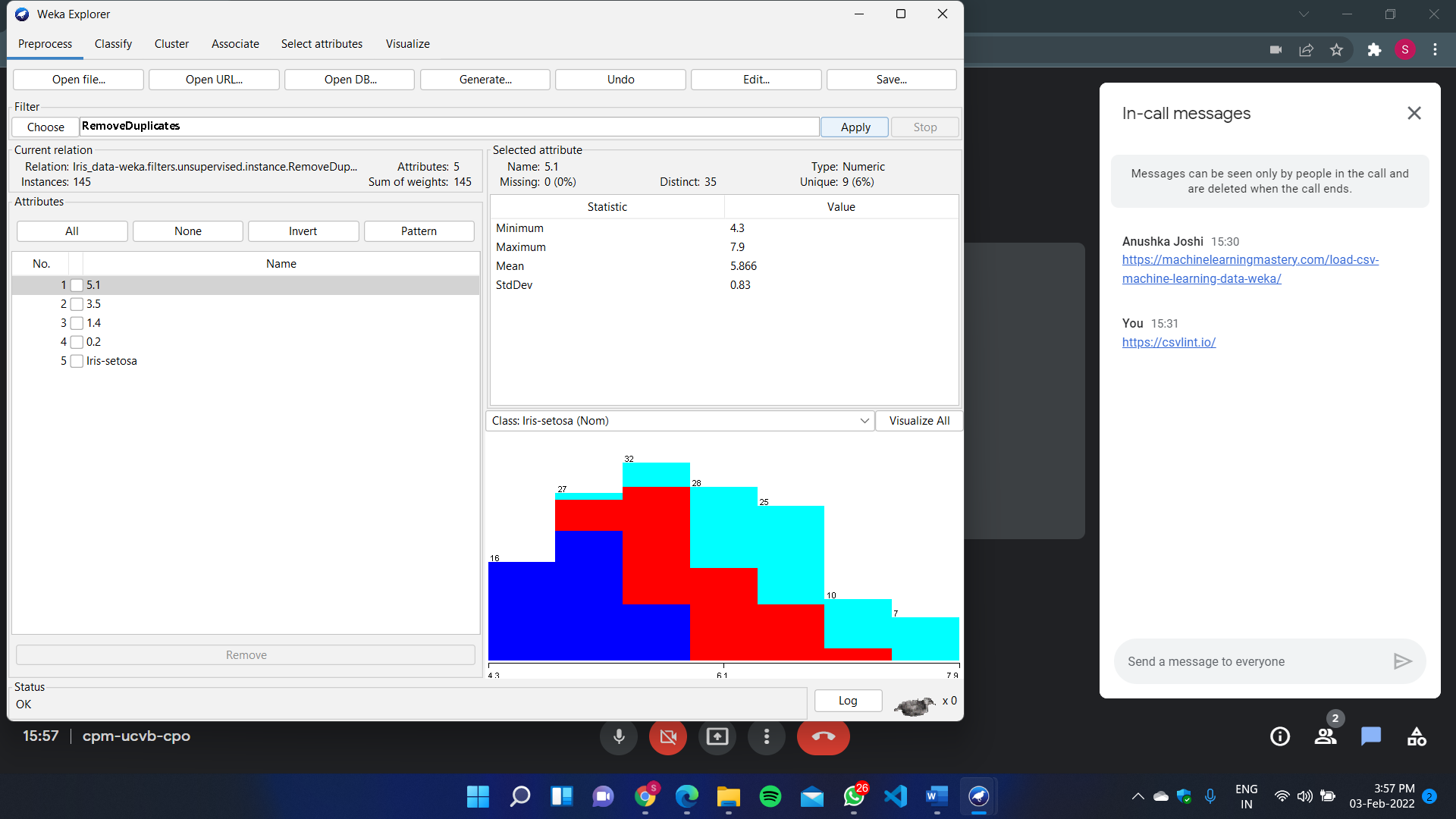


**Pre-processing**

* For adding pre-process filter, click on filter and select filter



* After Applying



**Classification**

* Select the classify option and select appropriate method from filter
* I have chosen Random Forest under Trees
* After that we can adjust the options and hit start to see result
* I have selected cross validation with 10 folds



**2. Orange is an easy-to-use data visualization tool with a large toolkit. In spite of being a GUI-based beginner-friendly tool, you mustn’t mistake it for a light-weight one. It can do statistical distributions and box plots as well as decision trees, hierarchical clustering and linear projections.**

**a. Install orange**

**b. Show data distribution**

**c. Show linear projection**

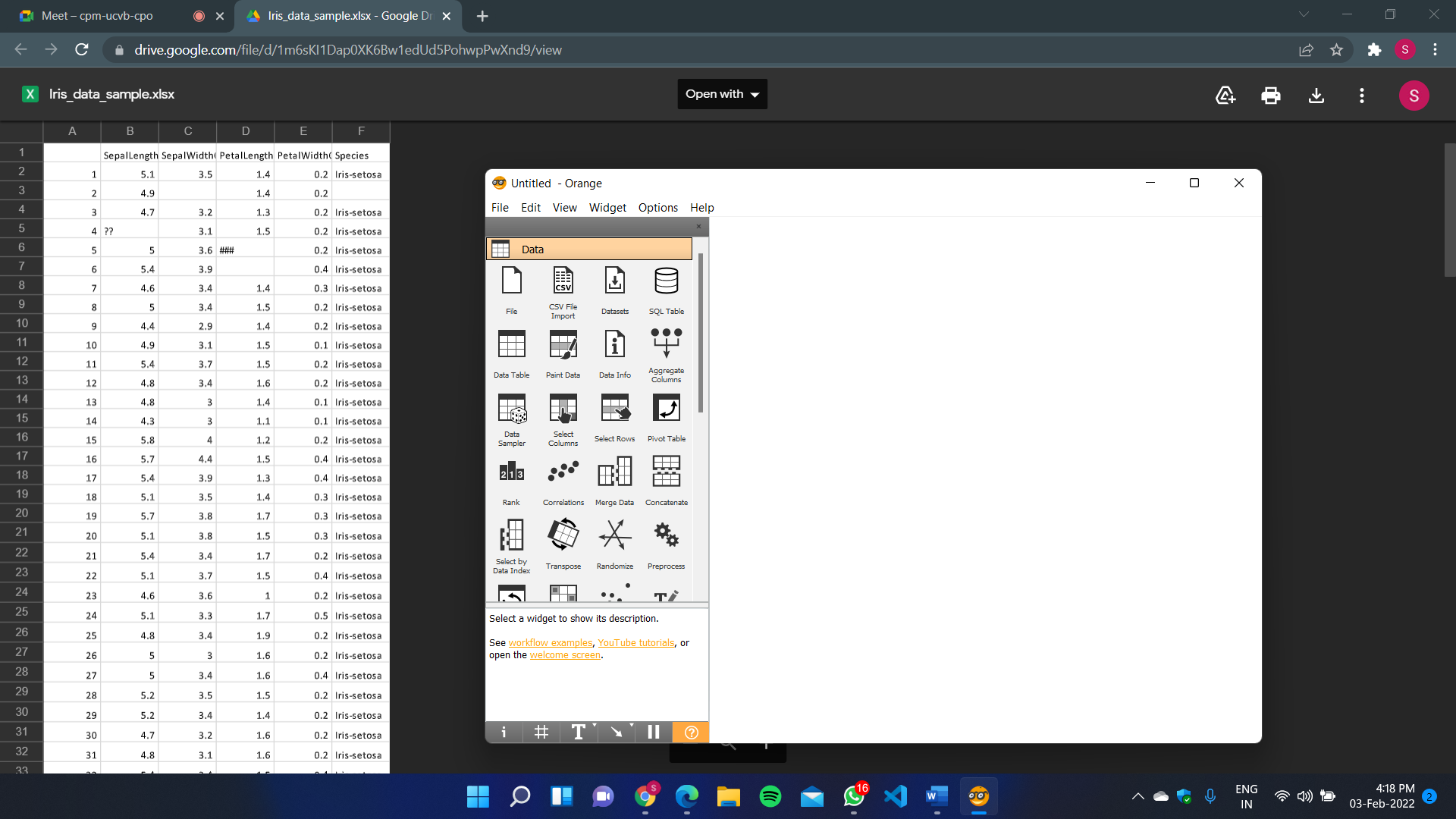
**d. Show FreeViz**

**Use dataset**

[**https://drive.google.com/file/d/1m6sKI1Dap0XK6Bw1edUd5PohwpPwXnd9/view**](https://drive.google.com/file/d/1m6sKI1Dap0XK6Bw1edUd5PohwpPwXnd9/view)

**Create a report for this task and upload screenshots for the same.**

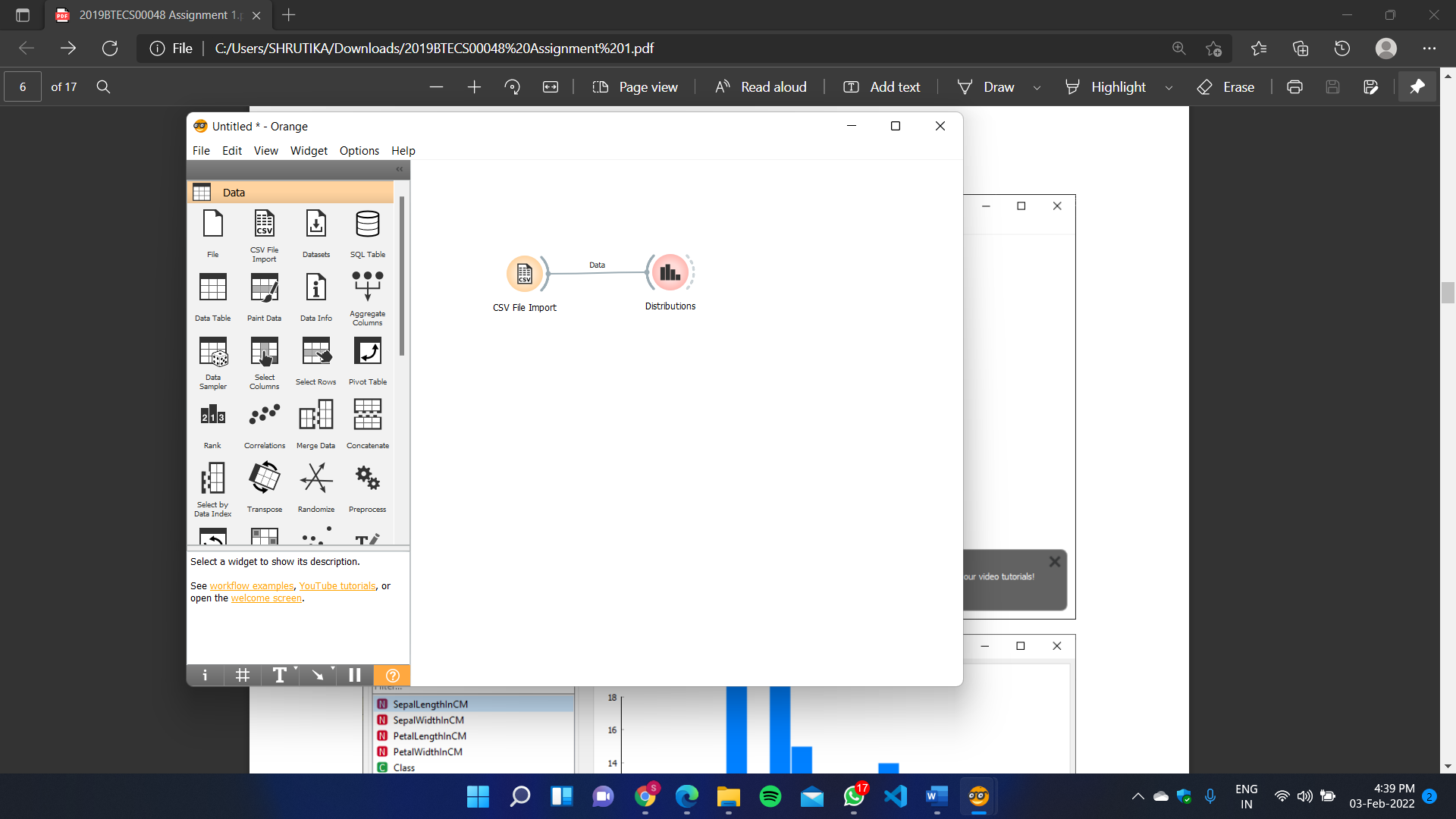
* Install and Open Orange

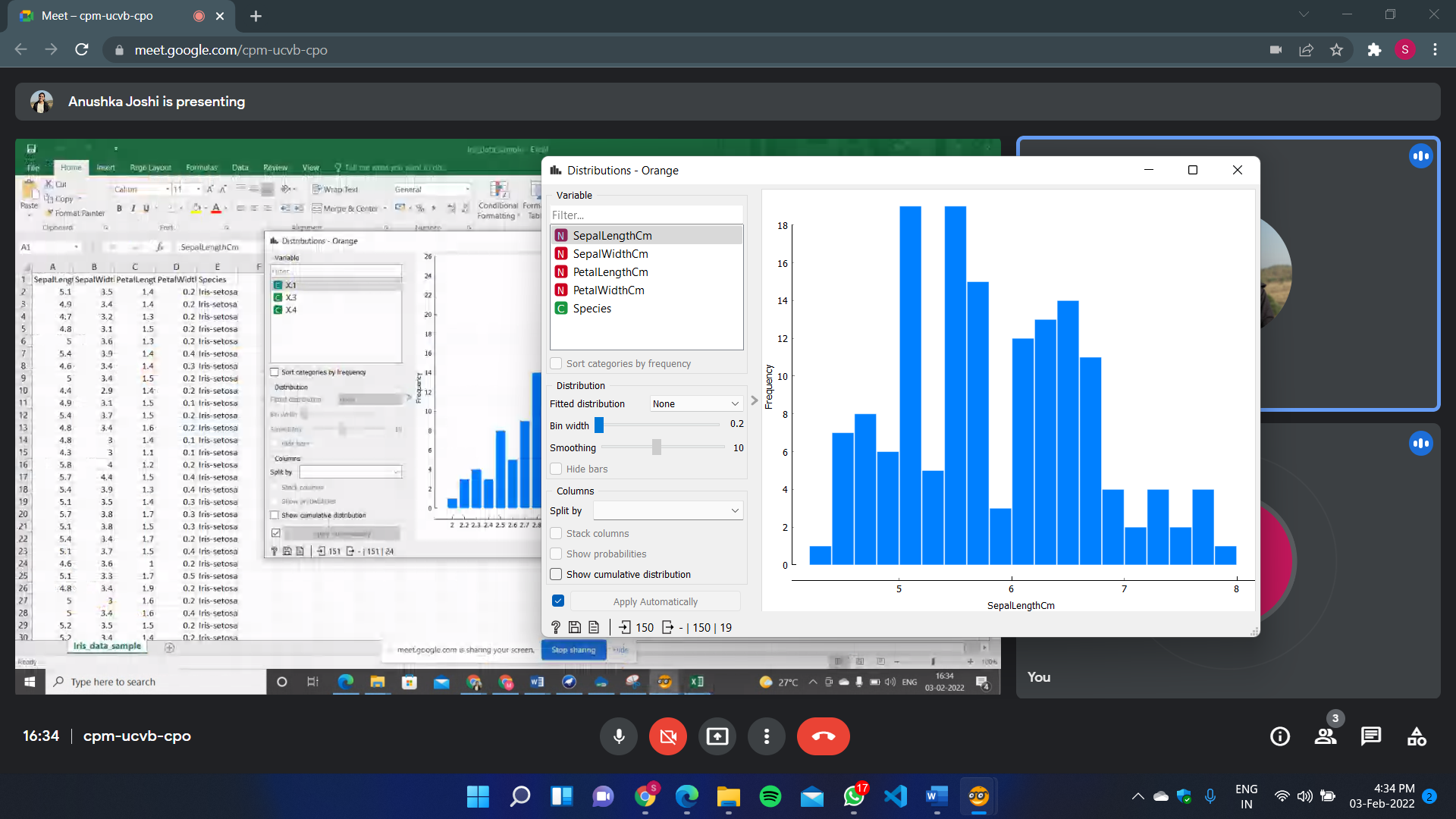


**Data Distribution**

Select CSV import and choose dataset file

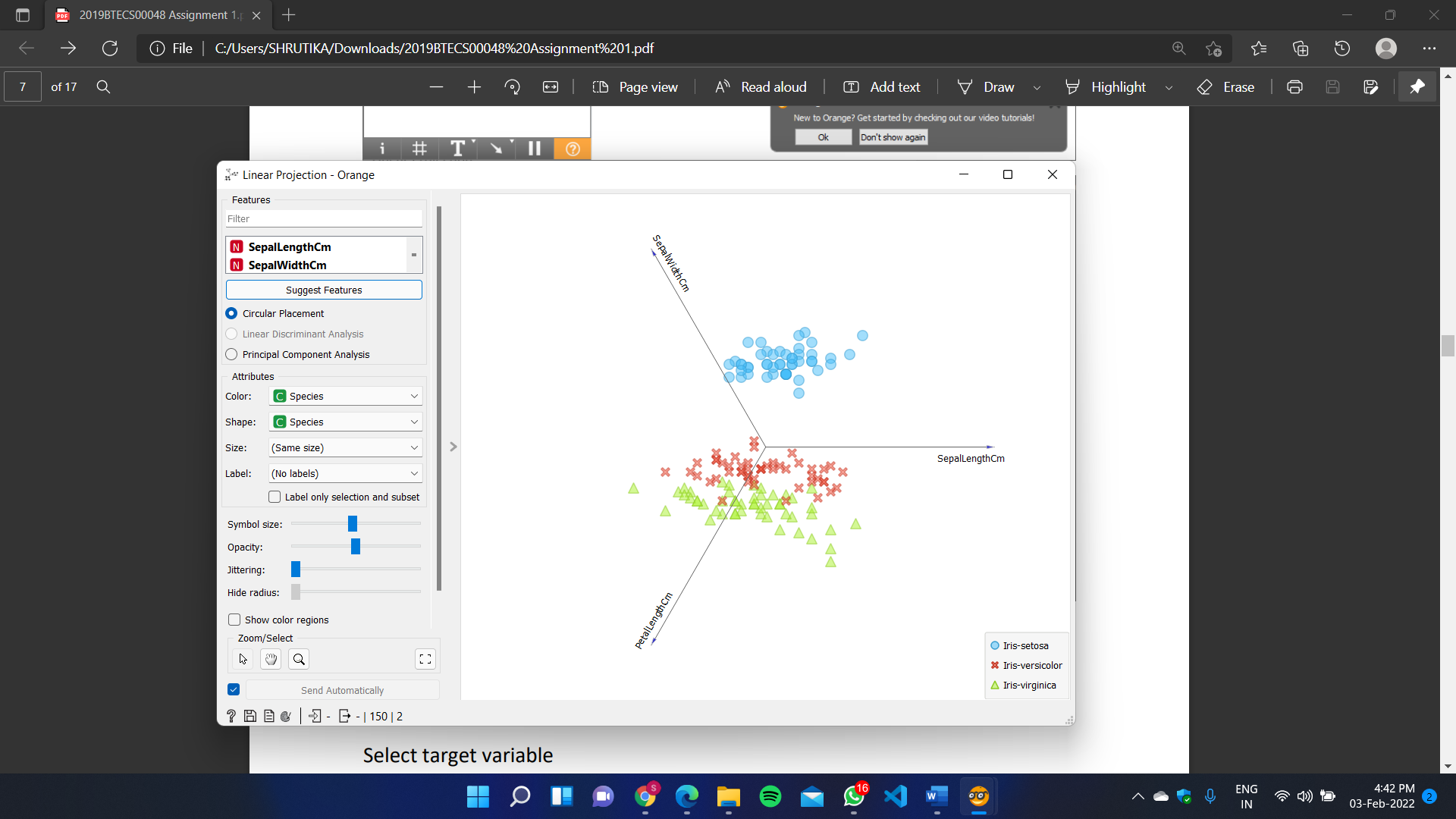
Select CSV import icon and then drag and search distribution.





**Linear Projection**

Select Import CSV file and drag and search projection

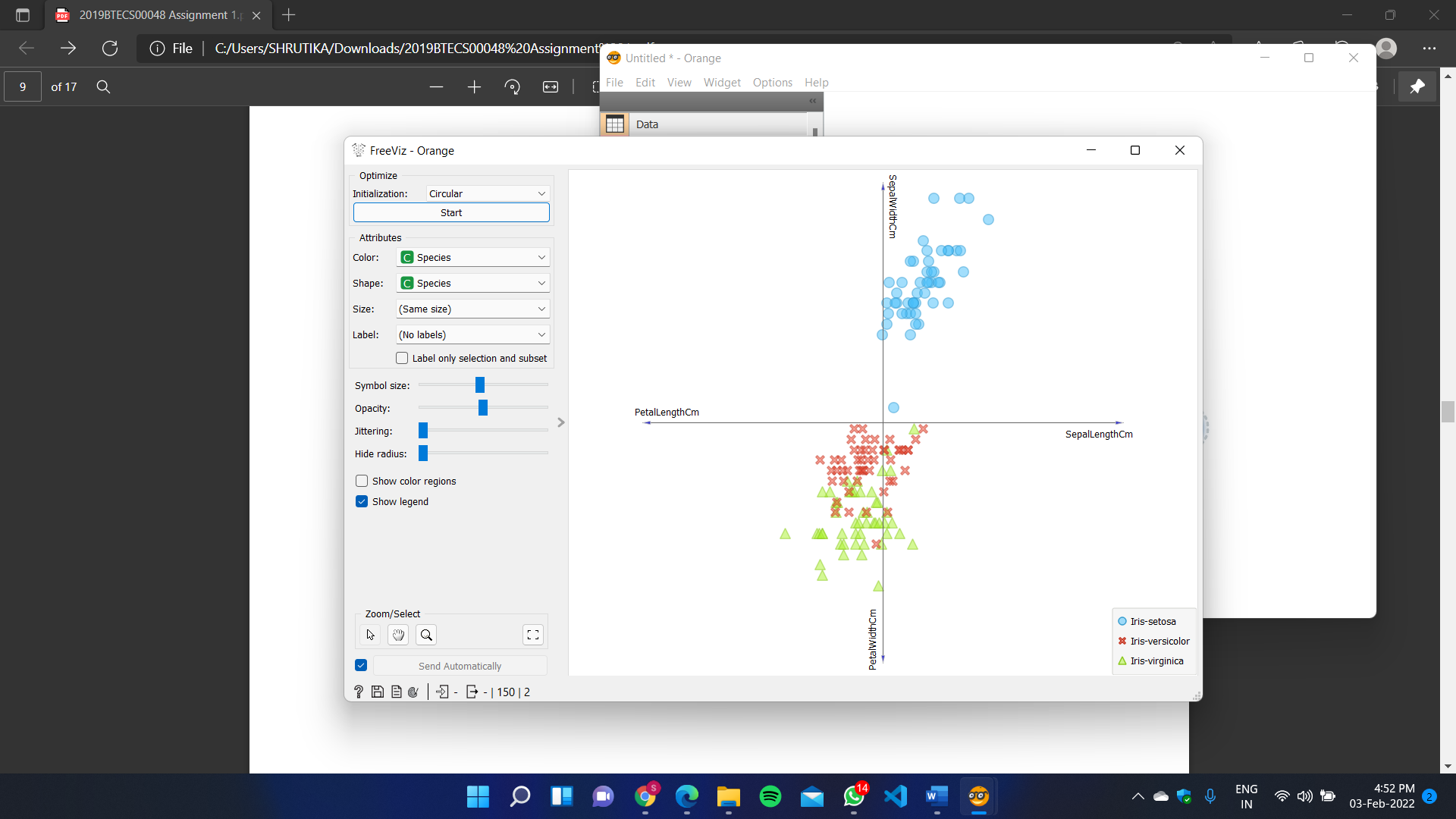


**FreeViz**

Drag from Import CSV and select Select Column

Select target variable

Drag from Select Column and search FreeViz



**3. Differentiate in between free software, Open-source software and proprietary software with respect to its properties.**

**1. Free Software:**

* Free software (or libre software is computer software distributed under terms that allow users to run the software for any purpose as well as to study, change, and distribute it and any adapted versions.
* Free software is a matter of liberty, not price; all users are legally free to do what they want with their copies of a free software (including profiting from them) regardless of how much is paid to obtain the program.
* Computer programs are deemed "free" if they give end-users (not just the developer) ultimate control over the software and, subsequently, over their devices.
* The right to study and modify a computer program entails that source code— the preferred format for making changes—be made available to users of that program.
* While this is often called "access to source code" or "public availability", the Free Software Foundation (FSF) recommends against thinking in those terms, because it might give the impression that users have an obligation (as opposed to a right) to give non-users a copy of the program.

**2. Open-source Software**:

* Open-source software is computer software whose source code is available openly on the internet and programmers can modify it to add new features and capabilities without any cost.
* Here the software is developed and tested through open collaboration.
* This software is managed by an open-source community of developers.
* It provides community support as well as commercial support if available for maintenance.
* We can get it for free of cost.
* This software also sometimes comes with a license and sometimes does not.
* This license provides some rights to users.

1. Software can be used for any purpose
2. Allows studying how the software works
3. Freedom to modify and improve the program
4. No restrictions on redistributions Some examples of Open source software includes Android, Ubuntu, Firefox, Open Office etc.

**3. Proprietary Software**:

* Proprietary software is computer software where the source codes are publicly not available only the company that has created can modify it.
* Here the software is developed and tested by the individual or organization by which it is owned not by the public.
* This software is managed by a closed team of individuals or groups that developed it.
* We have to pay to get this software and its commercial support is available for maintenance.
* The company gives a valid and authenticated license to the users to use this software.
* But this license put some restrictions on users also like.

1. the number of installations of this software into computers
2. Restrictions on sharing of software illegally
3. Time period up to which software will operate
4. Number of features allowed to use

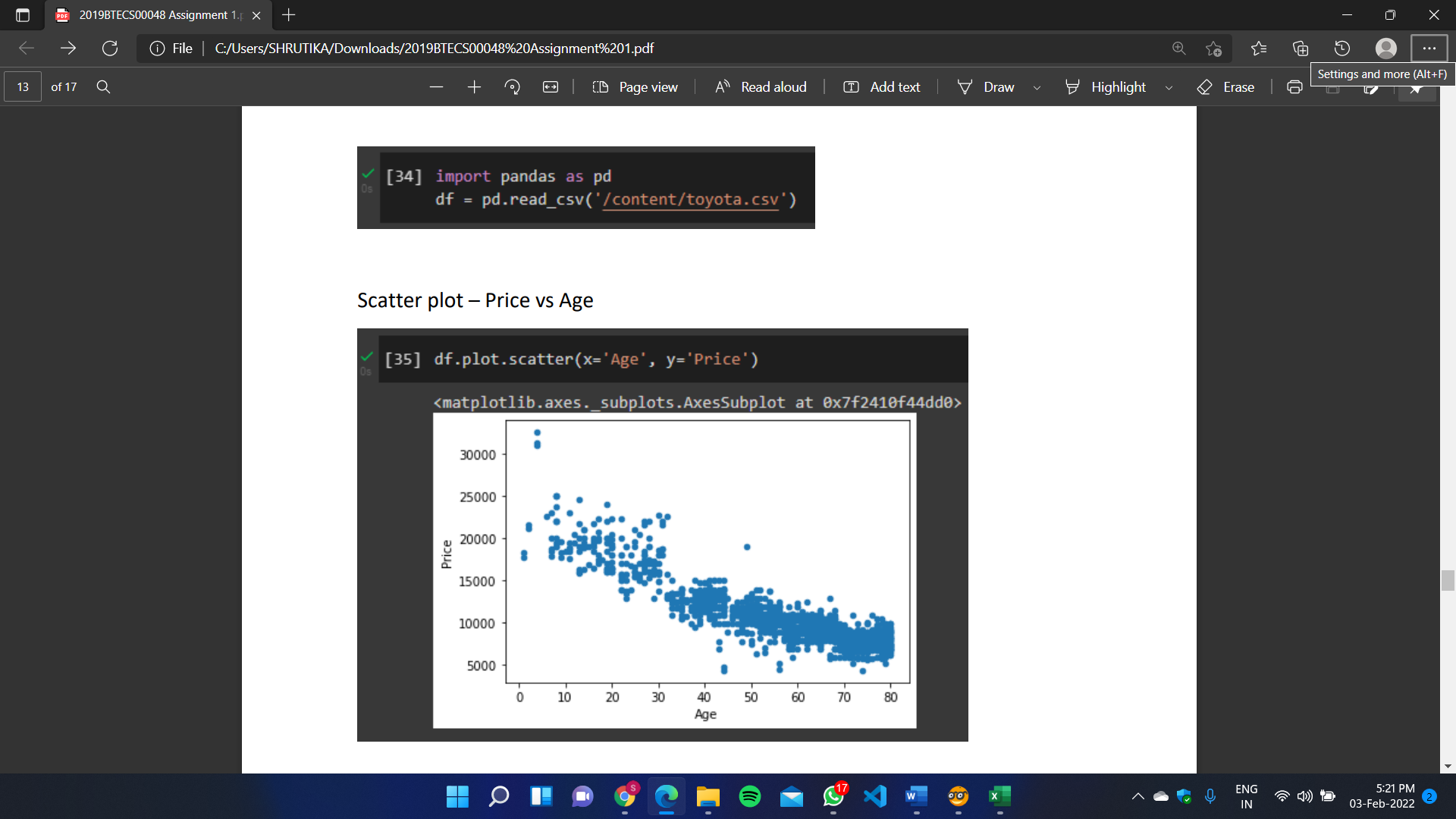
**4. Using Anaconda Python create Histogram, Scatter plot and Bar plot for the dataset given below.**

**Dataset-** [**https://drive.google.com/file/d/1i11BZFe8Xj9kNq7eeE9KOa\_Iz1KhEdXJ/view**](https://drive.google.com/file/d/1i11BZFe8Xj9kNq7eeE9KOa_Iz1KhEdXJ/view)

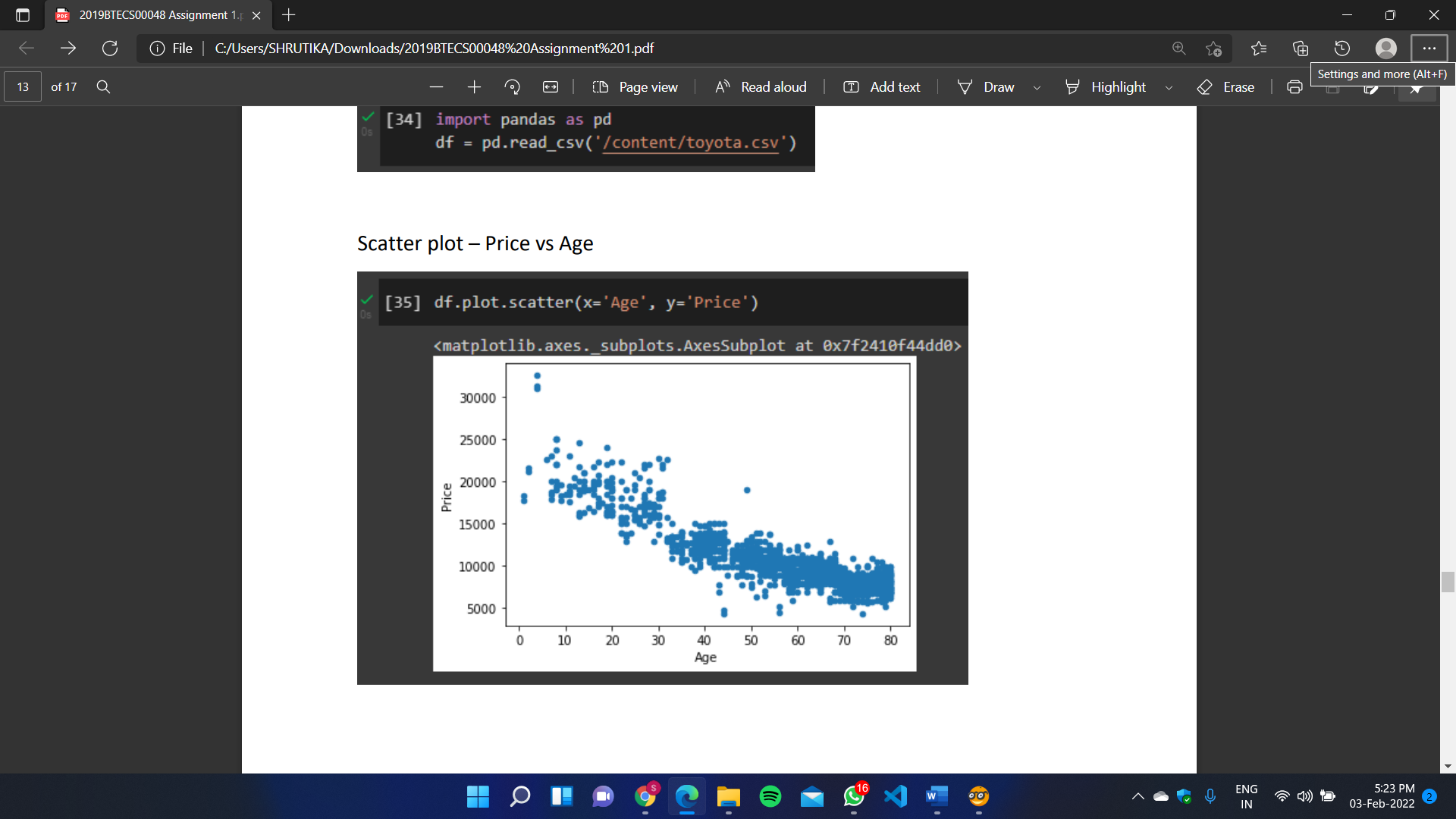
**a. Scatter plot- Scatter plot of Price Vs Age**

**b. Histogram- for Kilometre and CC**

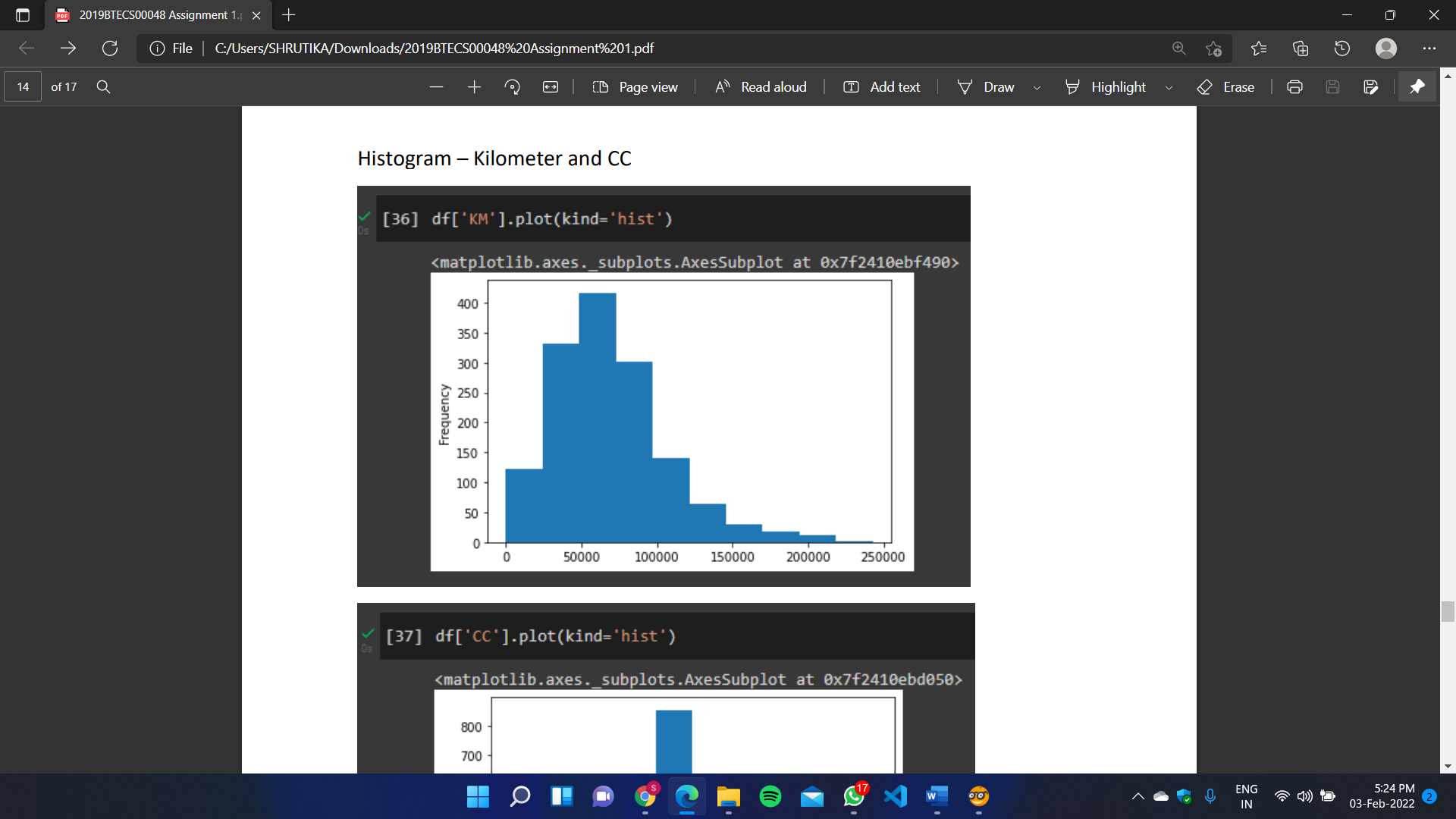
**c. Bar plot- Bar plot for different fuel types**

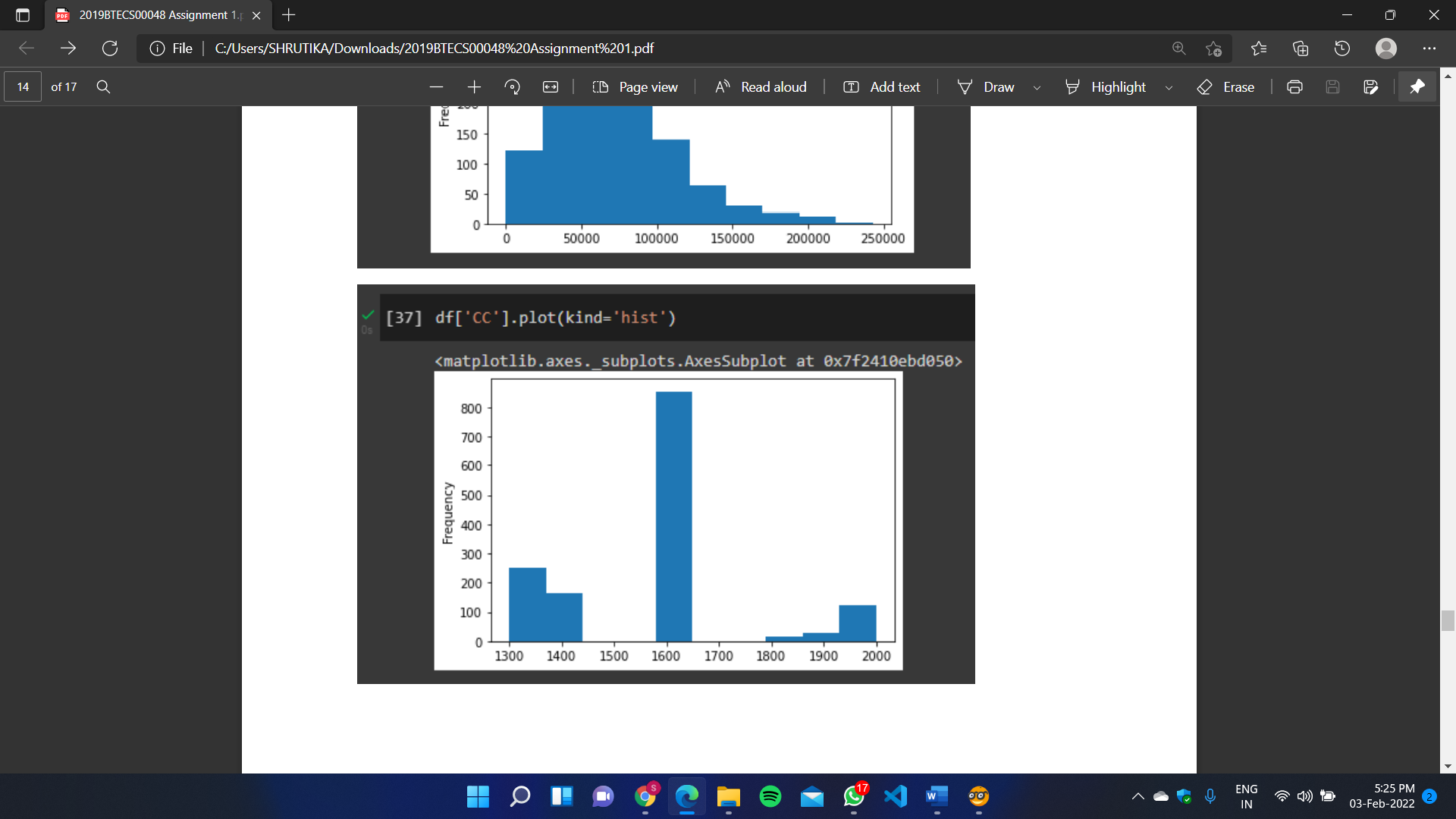


Scatter plot – Price vs Age

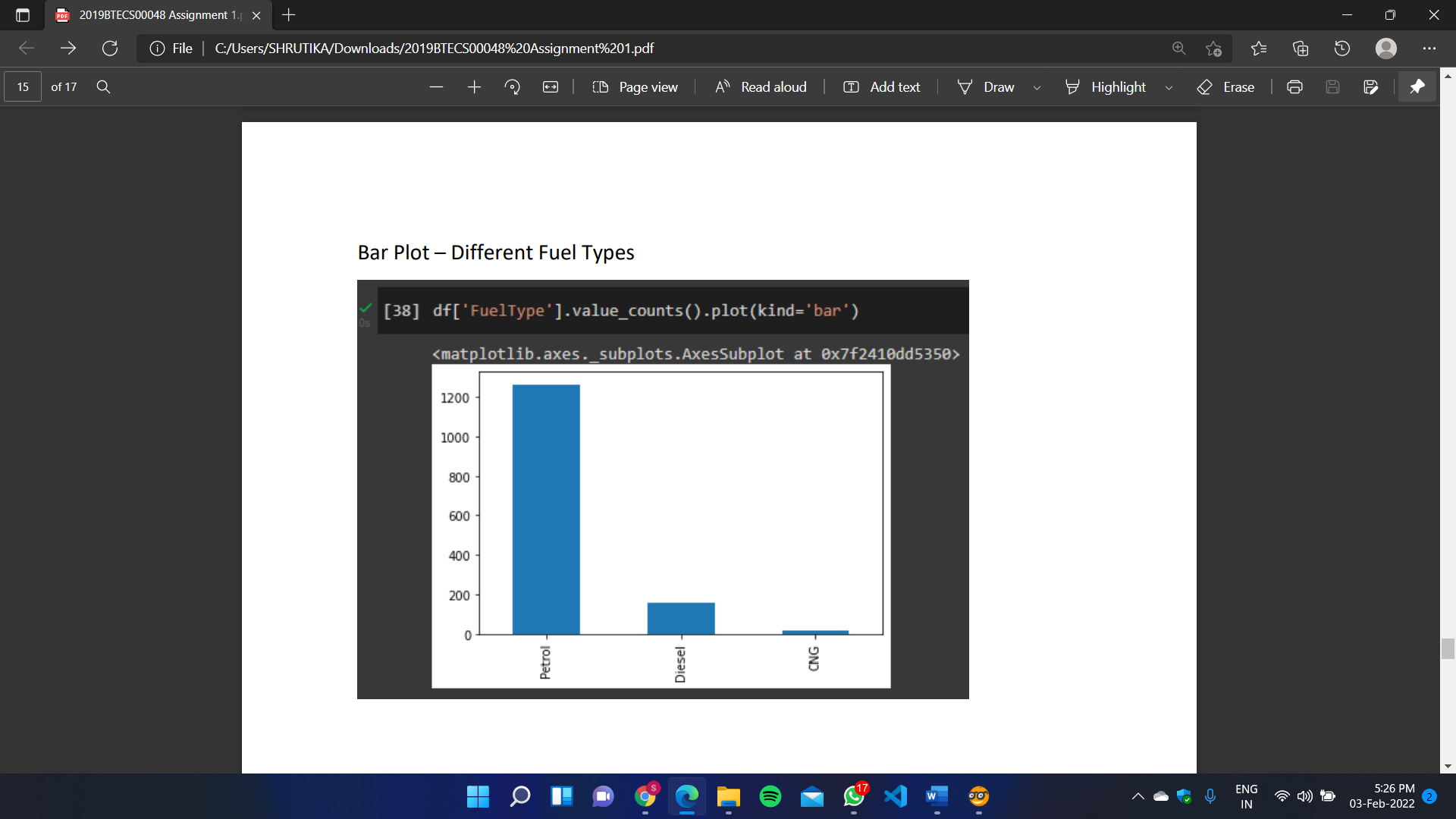


Histogram – Kilometer and CC





Bar Plot – Different Fuel Types



**5. Enlist some examples along with its purpose and properties (at least 10) of FOSS and proprietary software with respect to database.**

**1. PostgreSQL**

This relational database software has been around since 1997 and is the top choice in communities like Ruby, Python, Go, etc.

**2. MariaDB**

MariaDB was created as a replacement for MySQL by the same person who developed MySQL.

**3. CockroachDB**

The idea behind “cockroach” is that it’s an insect built for survival. No matter what happens — predators, floods, eternal darkness, rotting food, bombing, the cockroach finds a way to survive and multiply.

**4. ClickHouse**

It uses every hardware to its maximum potential to approach each query faster. The peak performance of processing a query usually remains more than two terabytes each second.

**5. Neo4j**

Support for transactional applications and graph analytics. Data transformation abilities for digesting large-scale tabular data into graphs. Specialized query language (Cypher) for querying the graph database Visualization and discovery features

**6. Redis**

When it comes to databases, it’s almost too easy to overlook the existence of Redis. That’s because Redis is an in-memory database and is mostly used in support functions like caching.

**7. SQLite**

SQLite is a lightweight C library that provided a relational database storage engine. Everything in this database lives in a single file (with a .sqlite extension) that you can put anywhere in your filesystem. And that’s all you need to use it! Yes, no “server” software to install and no service to connect to.

**8. Cassandra**

Cassandra belongs to what’s known as the “columnar” family of databases. The storage abstraction in Cassandra is a column rather than a row. The idea here is to store all the data in a column physically together on the disk, minimizing seek time.

**9. Timescale**

The timescale is a type of what’s called a “time series” database. It’s different from a traditional database in that time is the primary axis of concern, and the analytics and visualization of massive data sets is a top priority

**10. CouchDB**

It is a neat little database solution that sits quietly in a corner and has a small but dedicated following. It was created to deal with the problems of a net loss and eventual resolution of data, which happens to be a problem so messy that developers would instead switch jobs than deal with it.