



Shameer Ali

Cielo Cortés

Juan D. Novoa

Khatereh Najafi

Margarita Villar Vega

Prof.Steve Ataky, PhD

Institut Teccart

Montreal, 2022

Objective:

Democratize the access to AI and Machine Learning (ML) by developing a user-friendly web application accessible to everyone in which users can obtain descriptive statistics, run classification and regression models and make predictions over a dataset provided by them.

User’s need:

* Easy and understandable way to input a dataset to be analyzed.
* Generate a statistical report for better understanding of data’s behavior.
* Run Machine Learning algorithms over the data to make predictions.

Requirements:

* Upload a Dataset in CSV format
* Module for Data Statistics
* Module for Machine Learning Models

Solution:

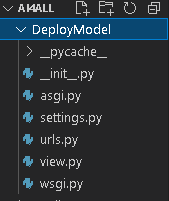
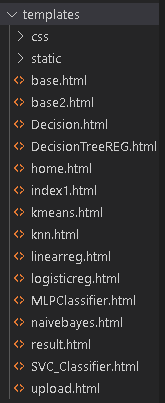
Introduction

It is the 21st century and thanks to technology advances world is more connected than ever before. Data is being collected in massive amounts and it is easily accessible by almost every individual or company. However, data itself is merely facts, numbers and figures. It needs to be analyzed. Data analysis organizes, interprets, structures and presents the data into useful information that provides context for better understanding of the object of study. Furthermore, thanks to AI and ML now data not only is to be interpreted but can be used to predict future behaviors. Whether the data is a sales report, the amount of likes in social media or the symptoms of thousands of patients, these tools can be used to predict with highly-accurate rates the outcome of a new input for instance knowing if a new product will be successful before being launched in the market or identifying if a patient has or is likely to have a disease based on its symptoms.

AI4All has is conceived to give access to such tools to everyone, making a basic analysis and predictions within reach without the need of having a degree in data science.

1. **Creating the web application.**

To create the web application the framework Django has been chosen based on its simplicity, flexibility, reliability and most of all because it is based in python, the same language in which the ML algorithms have been written, making the integration of the models and the web app much simpler. For mor information on Django visit [Django’s documentation.](https://docs.djangoproject.com/en/4.0/)  
AI4All source code can be found on github in: <https://github.com/juandanovo/AI4all>

* 1. **Web App structure:** Django structures the web apps in 3 main layers:
     1. **The model layer:**Django provides an abstraction layer (the “models”) for structuring and manipulating the data of the web application. For this project the layer has been named “DeployModel” and is organized as in the image below.  
          
        ****
     2. **The view layer:**Django has the concept of “views” to encapsulate the logic responsible for processing a user’s request and for returning the response. It is the case of the documents *view.py* which contains all the logic and methods to be used and *urls.py* which will be accessed when redirecting to a specific template.
     3. **The template layer:**The template layer contains the html files for rendering the information to be presented to the user. At the moment of presenting this document, the templates tree looks like this:  
          
        

Home page view


Figure Home Page view

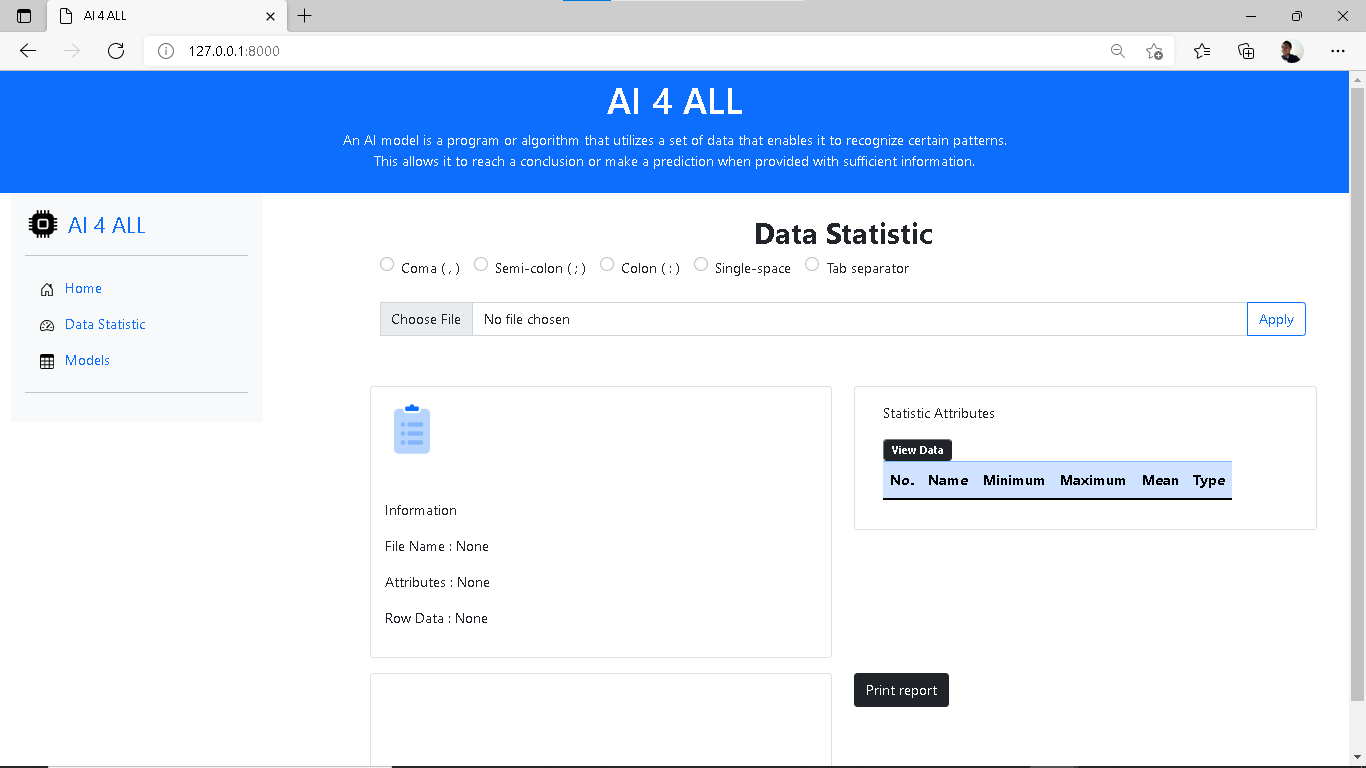


Figure Dashboard view

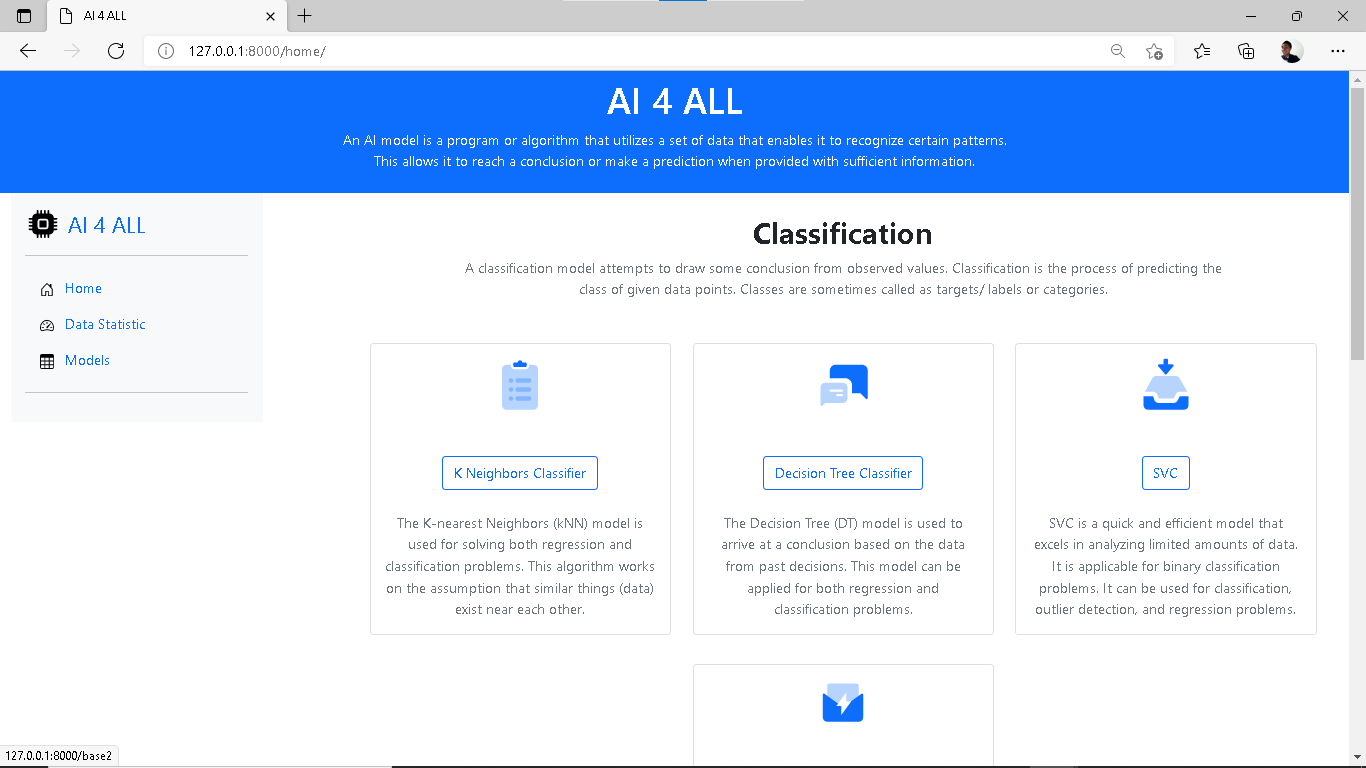


Figure Models view

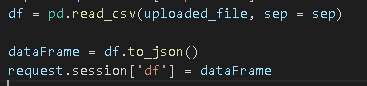
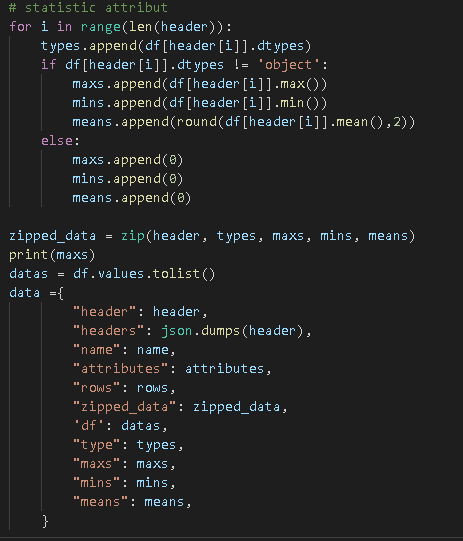
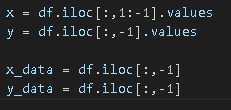
1. **Deploy Machine Learning Algorithms:**The process of making machine learning models available in production where web applications can use them is known as model deployment. Is thanks to this process the end user can consume the trained model by providing new data points and generating predictions.  
   Typically, machine learning models are created in order to anticipate an outcome. It can be a binary value i.e., 1 or 0 for Classification studies or continuous values for regressions and labels for clustering.  
   AI4All offers a set of well-known algorithms for classification and regression. The models that are currently available in the platform are:

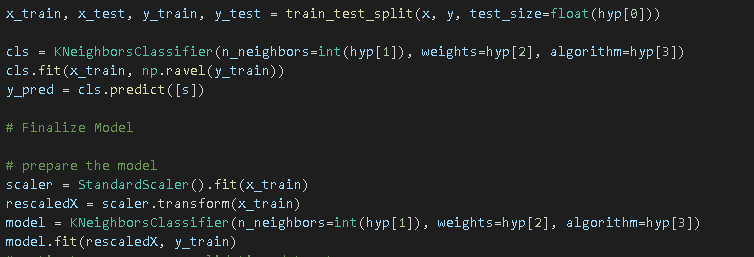
* K Neighbors Classifier
* Decision Tree Classifier
* C-Support Vector Classifier (SVC)
* Multi-layer Perceptron (MLP) classifier
* Naive Bayes
* Logistic Regression
* Linear Regression
* Kmeans
* Decision Tree Regressor

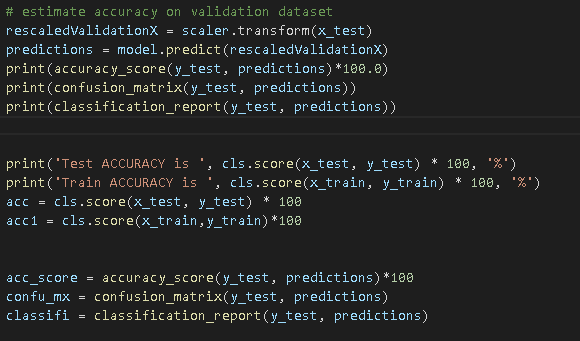
For documentation on ML models in python go to: [scikit-learn: machine learning in Python — scikit-learn 1.0.2 documentation](https://scikit-learn.org/stable/index.html)

**Note:** This section focuses on the deployment of models in the backend, to see what they look like on the front go to Users’s guide section.

**Libraries used for ML**:

* Numpy
* Pandas
* Matplotlib
* Plotly
  1. **Deployment steps:**Steps are explained for K Neighbors Classifier model, the steps are similar for the other models exchanging only the parameters to be entered.  
     1. **Reading the data:** once the user has uploaded the dataset via POST method it is read and save in the variable df and converted into a Session variable that can be used in any view.   
        **Note:** Data4All does not keep track of any data uploaded to its platform, session variables are destroyed with the session once the user quits the web app.
     2. **Visualizing data:** Once uploaded and read, data is printed to a list along with the type of data, max, mins and means for a better understanding of the user. 
     3. **Preprocessing:** Data is separated into features and target classes into variables X and Y   
        
     4. **Prepare the model:**  data is separated (split) into random train and test subsets, then normalized to values between 0 and 1 so they all have the same relevance thus avoiding biased predictions.

****

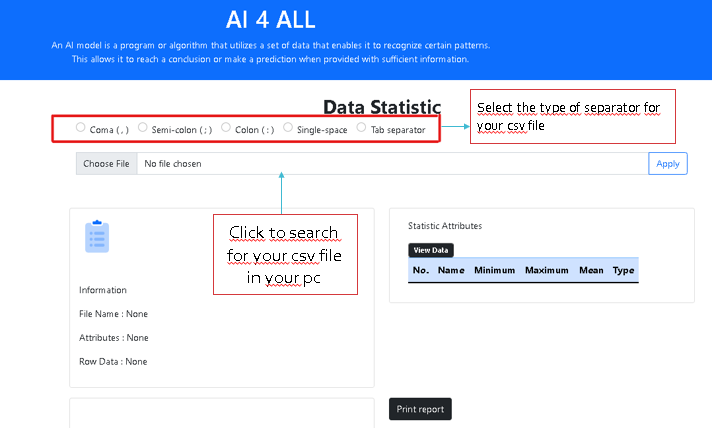
* + 1. **Running model:** Now that the input is ready, the next step is to run the model. 
    2. **Making predictions and testing accuracy:** with the model already run, is time to extract the information relevant for the user. It means the prediction and the accuracy of said prediction ****

1. **User guide:**

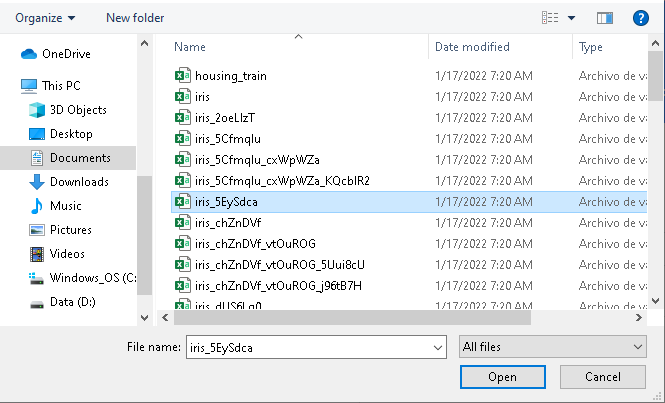
To show the performance a case of study is presented using the dataset: **iris\_5EySdca.csv**, it can be found in our [GitHub repository](https://github.com/juandanovo/AI4all/tree/main/AI4all/media) along with many other datasets that can be used to test the app ([*https://github.com/juandanovo/AI4all/tree/main/AI4all/media*](https://github.com/juandanovo/AI4all/tree/main/AI4all/media))

1. Uploading the dataset:

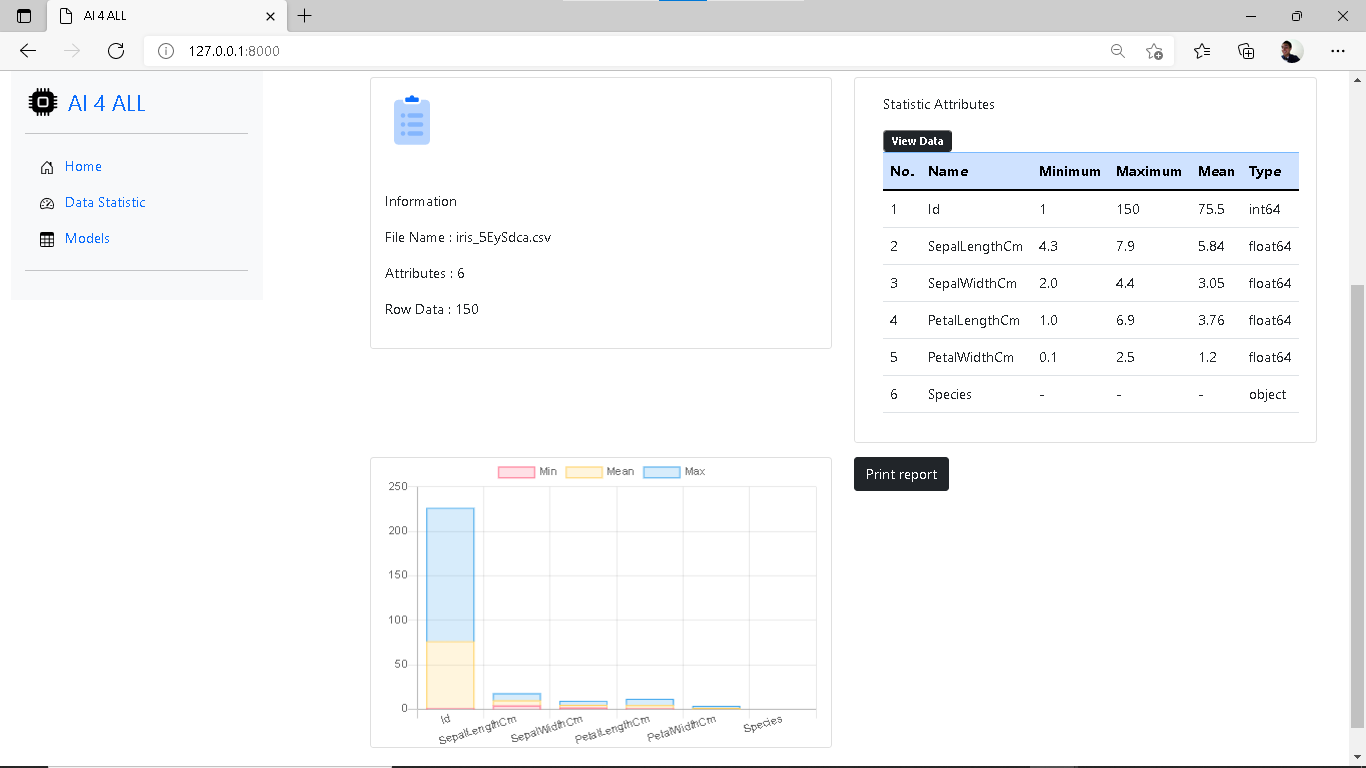
Select the type of separator, then click on “choos file”



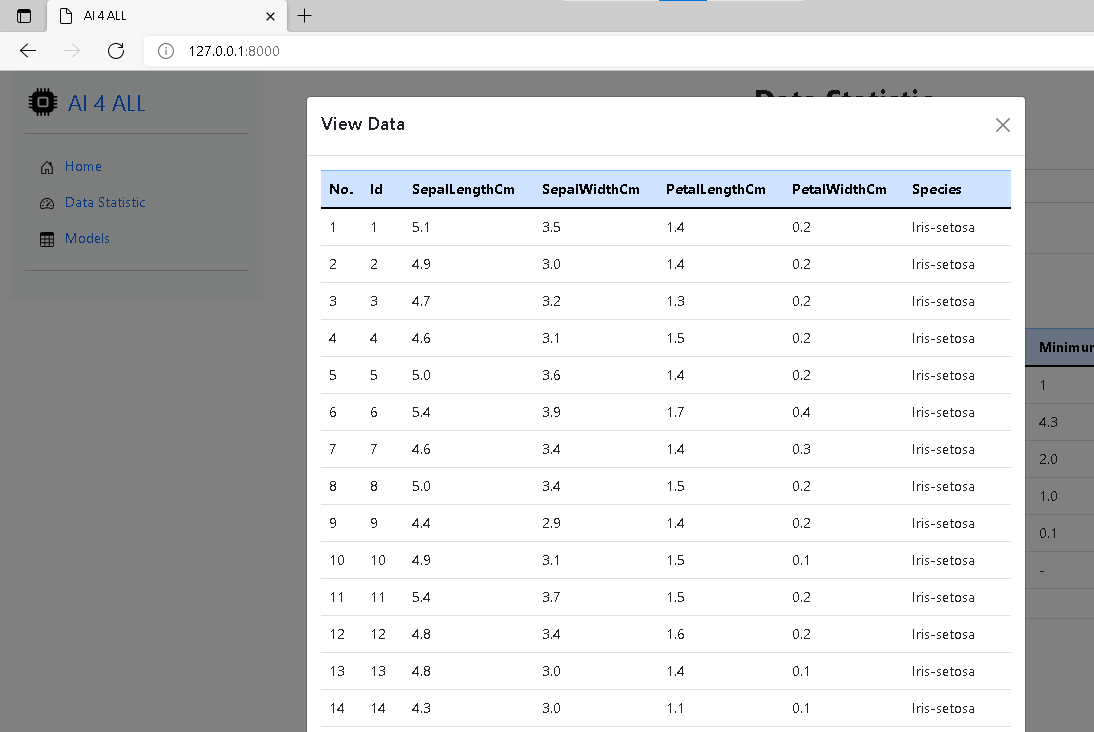
Select the desired dataset to use then click apply.



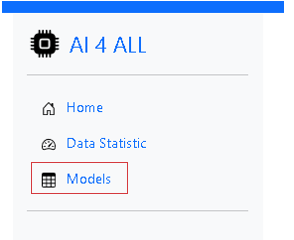
Once uploaded, the app will display the name of dataset, type of data, attributes and rows

****

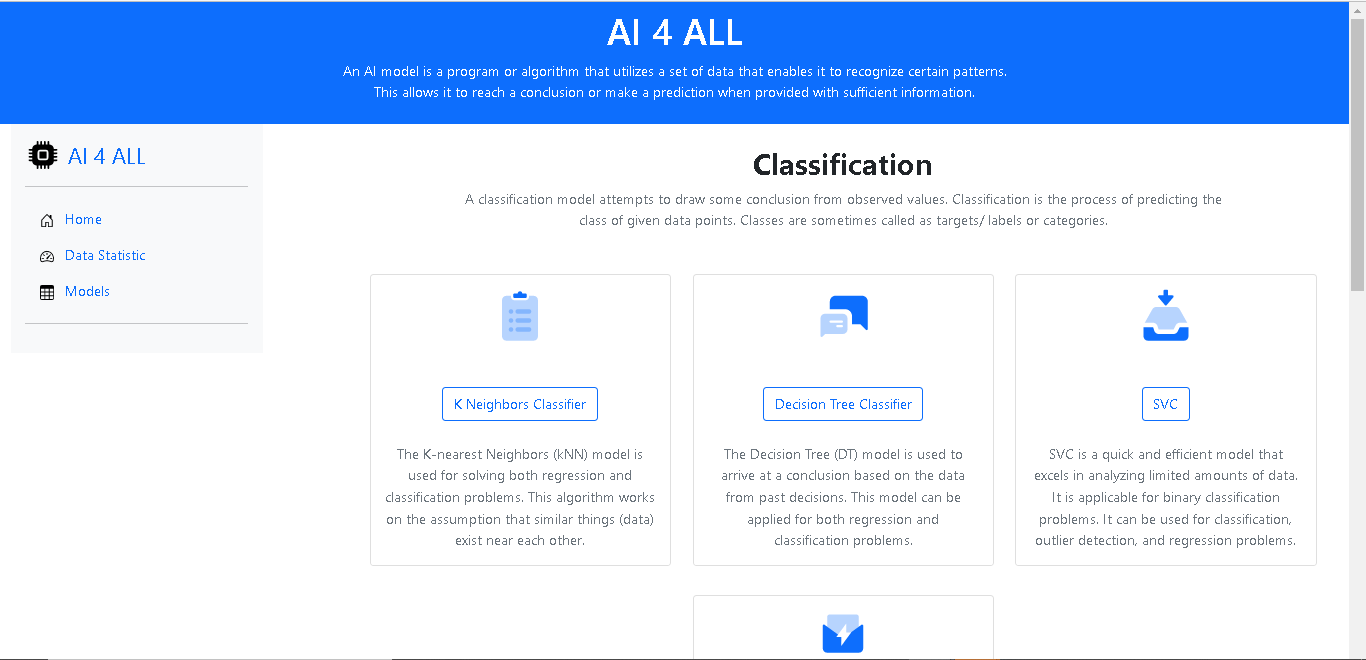
Click on “view data” to preview the dataset in the web app. It helps check if the upload was successful and the data keeps the same order.



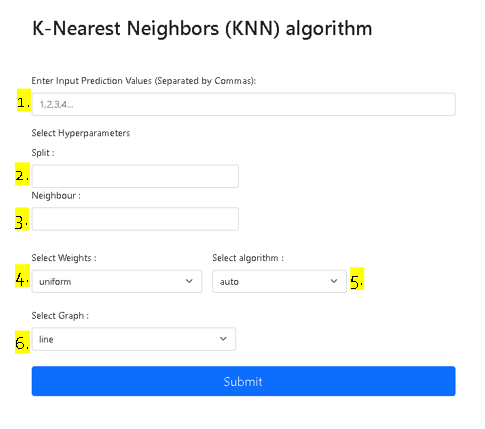
On the side bar click on “models” to see the list of models available.



It will lead you to the list of models

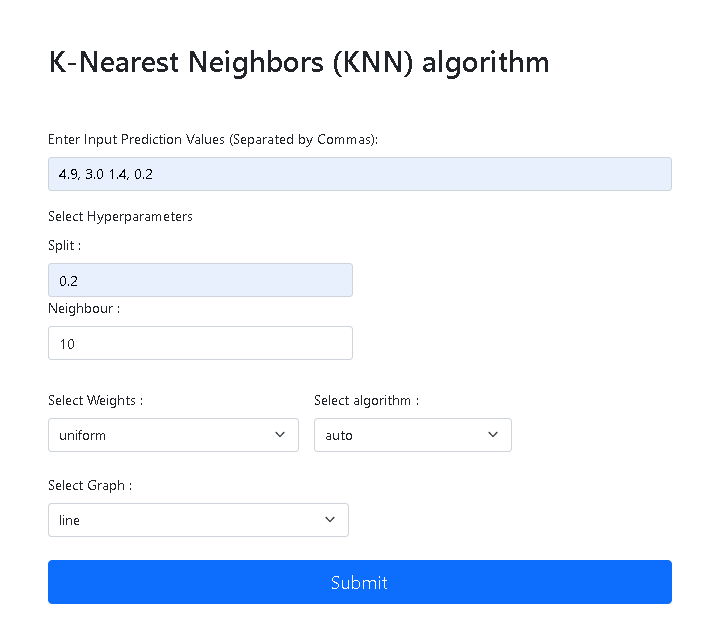


Select the model you would like to implement (for this example “K neighbors Classifier” is being used)

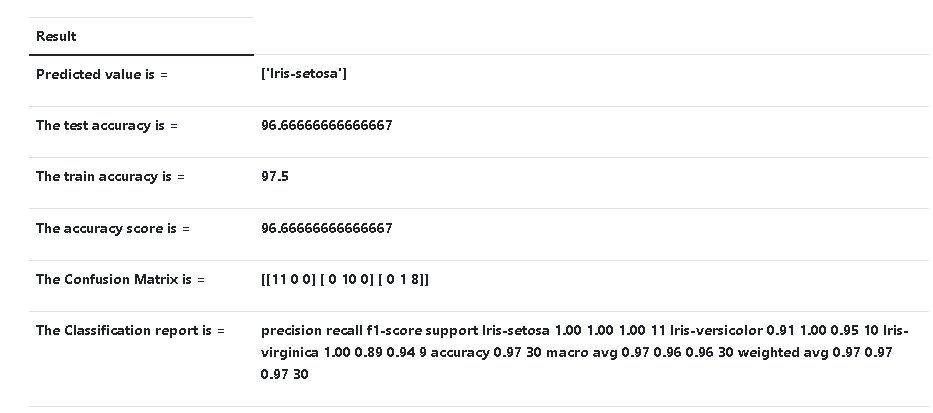


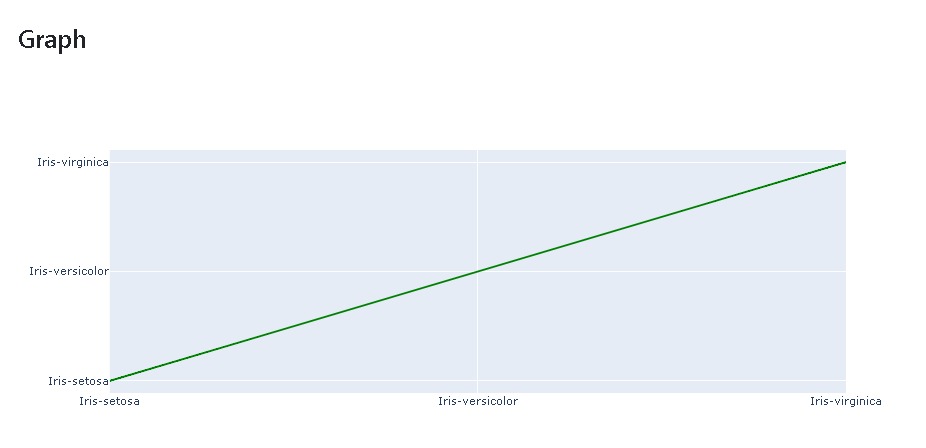
Insert the information required to run the model.

1. Values to make a new prediction
2. Size of split
3. Amount of K neighbors
4. Weight: uniform(default) or distance
5. Algorithm: auto (default)
6. Select the type of graph desired to represent the result.



**Results:**





Results display the class predicted for the values inserted in the form, the accuracy in train and test arrays, the accuracy of the prediction, the confusion matrix and a brief report based on the confusion matrix.

It is the final result, to try other models the user can go back to the “models” tab and choose a different one or try the same model with different values.

1. **Recommendations:**

Although **AI4All** platform can be used by anyone, ML algorithms will only return predictions and data in an understandable way. It is up to the user and its knowledge to take decisions based on the results given.

**Annex**

**Theory:**

For better understanding on key concepts of this project, a brief explanation is provided below.

**Clasiffication:**

Classification is **a machine learning technique used to categorize data into a given number of classes**. It will predict the class labels or categories for the new data. A decision tree is a supervised machine learning technique that predicts the class label of data objects.

In the field of data management, **data classification** as a part of the Information Lifecycle Management (ILM) process can be defined as a tool for categorization of data to enable/help organizations to effectively answer the following questions:

* What data types are available?
* Where are certain data located?
* What access levels are implemented?
* What protection level is implemented and does it adhere to compliance regulations?

When implemented it provides a bridge between IT professionals and process or application owners. IT staff are informed about the data value and management (usually application owners) understands better which part of the data centre needs to be invested in to keep operations running effectively. This can be of particular importance in risk management, legal discovery, and compliance with government regulations. Data classification is typically a manual process; however, there are many tools from different vendors that can help gather information about the data.

**Statistical Analysis : Classification of Data**

A picture containing timeline

Description automatically generated

there are four types of classification. They are: **(i) Geographical classification, (ii) Chronological classification,** **(iii) Qualitative classification, and (iv) Quantitative classification**.

**Geographical classification**

When data are classified on the basis of location or areas, it is called geographical classification

**Example:**Classification of production of food grains in different statesin India.

Table

Description automatically generated

**Chronological classification**

Chronological classification means classification on the basis of time, like months, years etc.

Table

Description automatically generated

**Qualitative classification**

In Qualitative classification, data are classified on the basis of some attributes or quality such as sex, colour of hair, literacy and religion. In this type of classification, the attribute under study cannot be measured. It can only be found out whether it is present or absent in the units of study.

**Quantitative classification**

Quantitative classification refers to the classification of data according to some characteristics, which can be measured such as height, weight, income, profits etc.

**Example:**The students of a school may be classified according tothe weight as follows

|  |  |
| --- | --- |
| Weighr (in kg) | No of students |
| 40-50 | 50 |
| 50-60 | 200 |
| 60-70 | 300 |
| 70-80 | 100 |
| 80-90 | 30 |
| 90-100 | 20 |

**Types of Diagrams:**

1. Bar chart  
2.Pie chart   
3.Pictograms or cartograms

Chart, diagram

Description automatically generated

**Regression model**

A regression model determines a relationship between an independent variable and a dependent variable, by providing a function. Formulating a regression analysis helps you predict the effects of the independent variable on the dependent one.

Example: we can say that age and height can be described using a linear regression model. Since a person’s height increases as its age increases, they have a linear relationship.

Regression models are commonly used as a statistical proof of claims regarding everyday facts.

## **Types of regression models**

### **Linear regression model**

A linear regression model is used to depict a relationship between variables which are proportional to each other. Meaning, the dependent variable increases/decreases with the independent variable.

In the graphical representation, it has a straight linear line plotted between the variables. Even if the points are not exactly in a straight line (which is always the case) we can still see a pattern and make sense out of it.

Example: As the age of a person increases, the level of glucose in their body increases as well.

A picture containing chart

Description automatically generated

### **Multiple regression model**

A multiple regression model is used when there is more than one independent variable affecting a dependent variable. While predicting the outcome variable, it is important to measure how each of the independent variables moves in their environment and how their changes will affect the output or target variable.

Example: Chances of a student failing their test can be dependent on various input variables like hard work, family issues, health issues, etc.

Chart, scatter chart

Description automatically generated

### **Non-linear regression model**

In the non-linear regression model, the graph doesn’t show a linear progression. Depending on how the response variable reacts to the input variable, the line will rise or fall showing the height or depth of the effect of the response variable.

To know that a non-linear regression model is the best fit for your scenario, make sure you look into your variables and their patterns. If you see that the response variable is showing not so constant output to the input variable, you can choose to use a non-linear model for your problem.

Example: A patient’s response to treatment can be good or bad depending on their body tendency and willpower.

**Source:**

* <https://cfml.se/blog/regression_using_keras/>
* <https://www.voxco.com/blog/regression-model-definition-types-and-examples/>
* <https://www.brainkart.com/article/Alfred-Marshall-s-Definition-(Welfare-Definition)_1509/>
* [scikit-learn: machine learning in Python — scikit-learn 1.0.2 documentation](https://scikit-learn.org/stable/index.html)
* [Django documentation | Django documentation | Django (djangoproject.com)](https://docs.djangoproject.com/en/4.0/)