1. **Introduction**

In agriculture, choosing the right crop to grow is an import decision and it depends on several factors like as the type of soil, weather and other natural conditions In this project, I use machine learning and python to help me with this analize and decision.

Based on that, the goal is to build a model that can suggest the best crop to plat based on sil and climate data.

This study is focused on the agriculture and food domain, one of the options available on the CA guidance for this assignment. The dataset I choose to use is called the Crop Recommendation Dataset, and it was found on Kaggle (<https://www.kaggle.com/datasets/atharvaingle/crop-recommendation-dataset>).

Just to highlight, it includes 2.2200 examples with 7 input features which are: levels of Nitrogen (N), Phosphorus (P) and Potassium (K) in the soil, temperature, humidity, pH and rainfall. The target column is the name of the recommended crop.

1. **OBJECTIVE:**

This is a classification problem, where I try to predict the crop type through using the features available and previous mentioned in my introduction. To solve this, I will test at least two different models, for example, Random Forest and Support Vector Machine (SVM) and also use cross0validation and hyperparameter tuning to check how well the the models perform and how reliable they are.

Based on that, another objective of this project is to show how machine learning can models can be used to sypport smart farming and better food production. By discussing all results with the help of graphs/charts and accuracy and performance scores (for example: accuracy and F1-score),

1. **EARLY DATA ANALYSIS**

Before starting with machine learning model it is important to undestante the entire dataset and also check the quality of the data. So, for this project, as I said in the introduction section I used the Crop Recommendation Dataset from Kaggle (shared by Atharva Ingle). It includes 2,200 rows and 8 columns, with different features that descbire soil and climate conditions and the last column is the target variable, which show the crop that should be recommended.

* + 1. The features in the dataset are:
* **N, P, K:** levels of Nitrogen, Phosphorus, and Potassium in the soil
* **temperature, humidity:** weather-related conditions
* **ph:** acidity level of the soil
* **rainfall:** average rainfall
* **label:** the name of the crop to be recommended

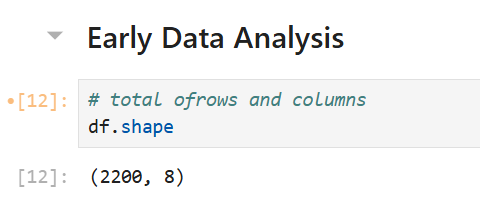
****

Figura 1 – Shape

Interface gráfica do usuário, Texto, Aplicativo

O conteúdo gerado por IA pode estar incorreto.

Figura 2 - Colum Names

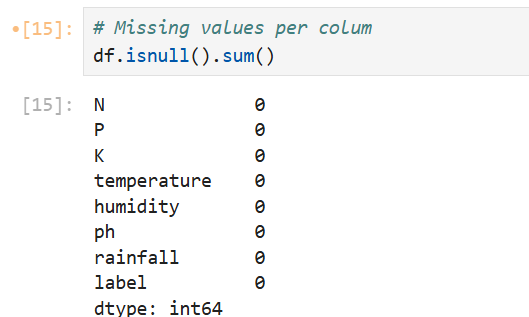


Figura 3 - Missing Values per Colum

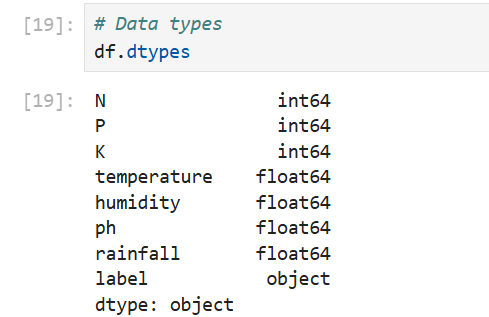


Figura 4 - Data Types

Texto, Tabela

O conteúdo gerado por IA pode estar incorreto.

Figura 5- First 5 rows

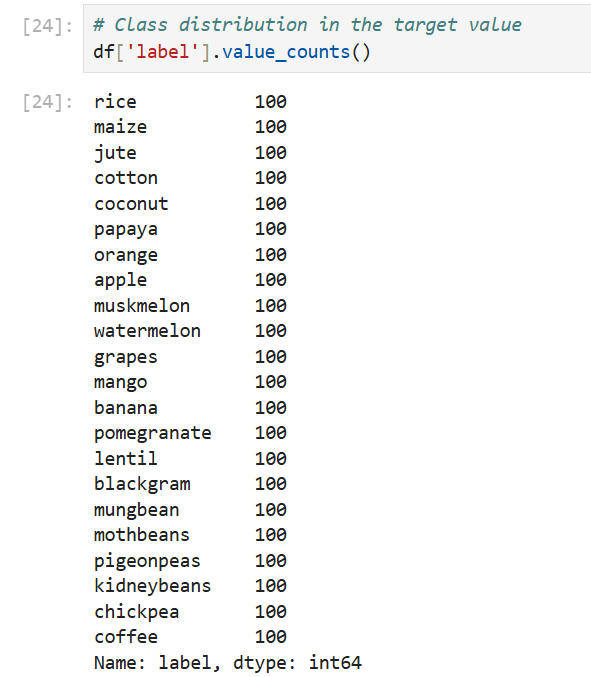


Figura 6 - Class Distribution

All the input features are numerical and the target is categorial (text). There were no missing values and the dataset was already clean.

* 1. Feature Distribution:

Gráfico, Histograma

O conteúdo gerado por IA pode estar incorreto.

Gráfico, Histograma

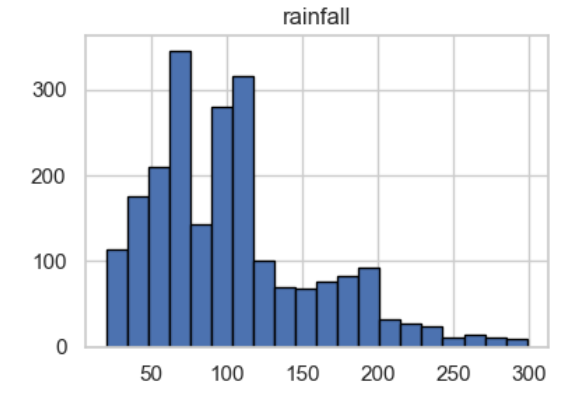
O conteúdo gerado por IA pode estar incorreto.

Figura 7 - Histogram for Numerical Features

I looked at the distribution of each feature by using histograms. The nutrient values (N,P,K) had more values on the lower end and some higher values as outliers, but, temperatures and humidity shows a normal distribution. The pH values was around 6 and 7, which is a neutral range for soil. Rainfall showed more variation, with a few cases of very high values.

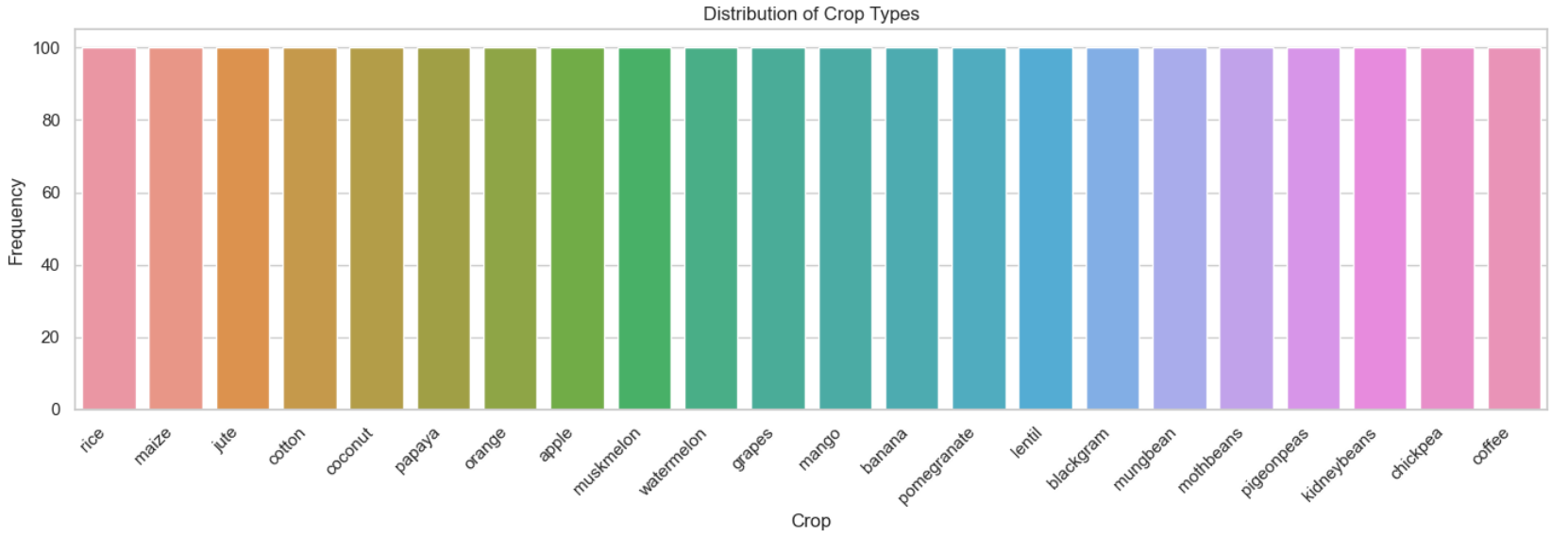


Figura 8 - Distribution of Crop Types

The target variable contains 22 different crop types, I checked how many times each crop appears in the data by using a bar chart and what I got is, the crops aver very balanced which means no class dominates..

Gráfico, Gráfico de mapa de árvore

O conteúdo gerado por IA pode estar incorreto.

Figura 9 - Correlation Heatmap of Numerical Features

I created a heatmap graph to see if any feature were strongly related, but, based on my interpretation most features had low or medium correlation with each other, which means they bring different kinds of information to the model, the strongest relation I found was between temperature and humidity, which is also weak in my point of view, but compared with others, make sense be the strongest correlation because warmer tempareatures usually come with higher humidity.

So, for this first analysis, I conclude that the dataset is in good condition and ready for machine learning models, the features are independent enough and there are no missing values and classes are balanced. Those aspects give us a strong base to proceed to build and process the models.

1. **Train-Test Split and Cross Validation**

To check how good and stable the model is, I used a Random Forest Classifier and tested it with 3 different samples of training and tests splits.I followed the instructions of the assignment and use 5%, 10% and 20% of the data for testing. On the training part, each split I applied 5 fold cross validation measure accurary.

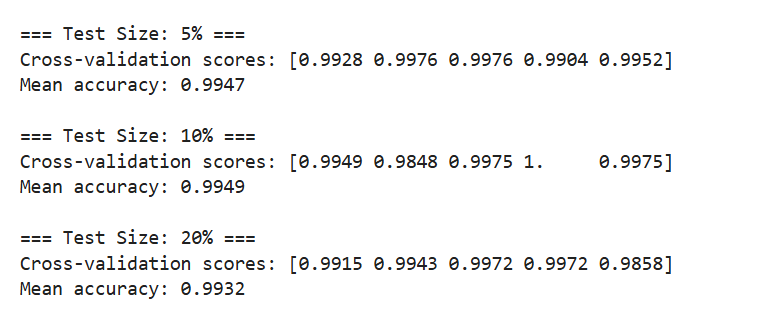


Figura 10 – Train/Test Split - Results

**Here are the results:**

* **Test Size**: 5%
  + - **Accuracy scores:** [0.9928, 0.9976, 0.9976, 0.9904, 0.9952]
    - **Mean accuracy:** 0.9947
* **Test Size:** 10%
  + - **Accuracy scores:** [0.9949, 0.9848, 0.9975, 1.0000, 0.9975]
    - **Mean accuracy: 0.9949**
* **Test Size: 20%**
  + - **Accuracy scores: [0.9915, 0.9943, 0.9972, 0.9972, 0.9858]**
    - **Mean accuracy: 0.9932**

The results show that the model works very well in all three cases. The accuracy got is always above 99%, which is very high. This means the model learns well from the training data and can make good predictions on new data. Furthermore, we can see that changing the test size (from 5% to 20%) does not affect the results too much. The accuracy stays almost the same. This is a good sign, because it shows the model is stable and can generalize to other situations.The scores from cross-validation are also very close to each other, which means that the model does not depend too much on a specific part of the data. In general the dataset is clean and well-balanced, and the Random Forest model gives excellent results for this classification task.

1. **Modeling**

In ths stage, as asked in this assignment I choose two different models to evaluate. As my goal is is to recommend the most suitable crop to plant based on soil nutrients and weather conditions, this characteristics makes this task a multiclass classification problem, where the target variable include 22 different types of crop. So, in order to solve this, I selected Random Forest Classifier and Support Vector Machine which are known to perform well with structured data.When Random Forest is a very strong and reliable model for classification tasks and known as working well with balanced data and can handle non linear relationships, SVM is also another powerful model which is known by performin well with smaller datasets.

* 1. Model Performance Results

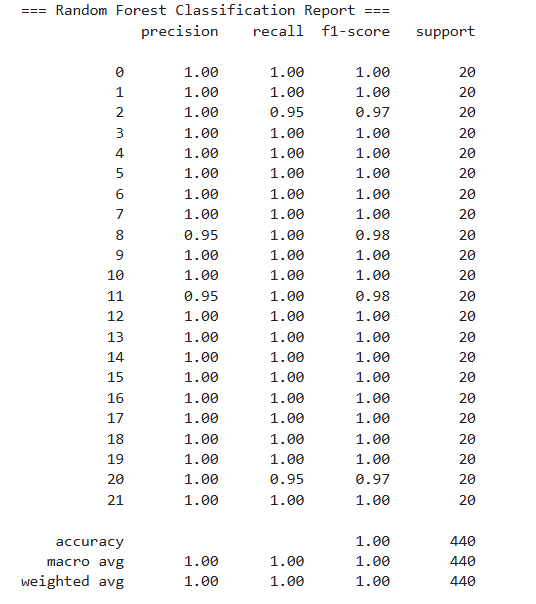


Figura 11 - Random Forest Report

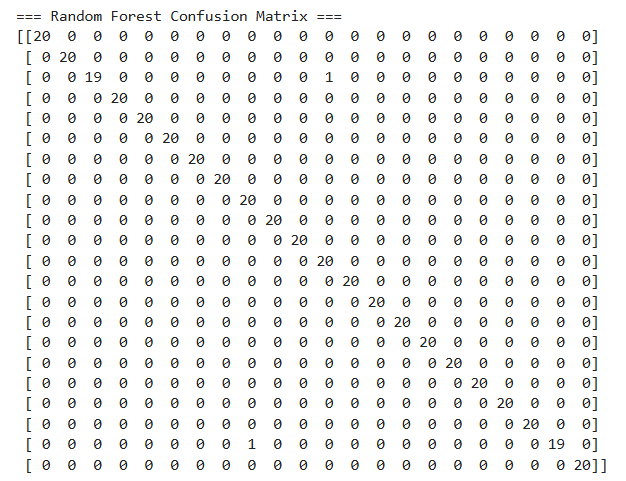


Figura 12 - Random Forest Confusion Matrix

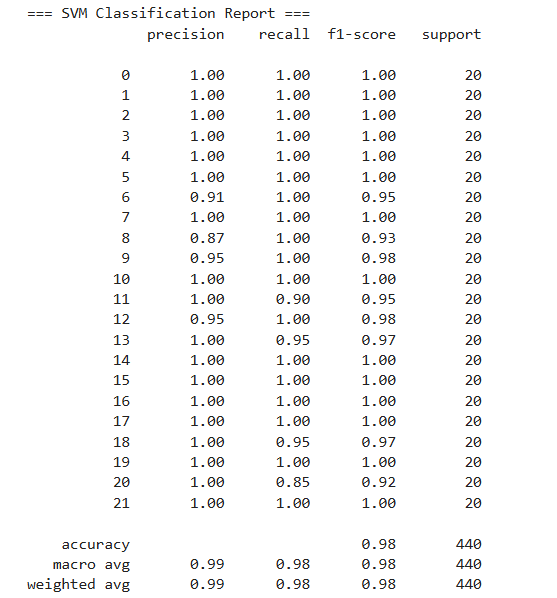


Figura 13 - SVM Classification Report

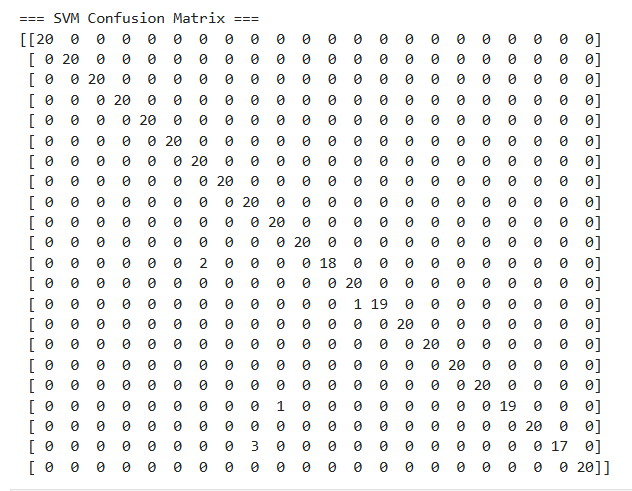


Figura 14 - SVM Confusion Matrix

**MODELS PERFORMANCE AND RESULTS:**

**After training both models using an 80/20 split, those are the results:**

**Random Forest Classifier:**

* **Random Forest Classifier:**
  + - **Accuracy:** 100%
    - **F1-Score:** Almost all classes scored 100%
    - **The confusion matrix** shows that nearly every prediction is correct with only 1 or 2 small mistakes.
* **Support Vector Machine:**
  + - **Accuracy:** 98%
    - A few crop types were misclassified such as class 20 and 8**.**
    - **F1-Score** for most all classes were still very high, but some were slightly below 1.00

Both models gave strong results, which shows that the dataset is clean, balanced and suitable for classification, however, Random Forest performed a bit better achieving nearly perfect scores across all metrics and the confusion matrix showed almost no errors. Because of this I choose Random Forest as the best model between both to move forward.

1. **Hyperparameter tuning**

Results:



Figura 15 -GridSearchCV Results

I applied GridSearchCV, a tool that tests all possible combinations of the chosen hyperparameters using cross-validation. This helps avoid overfitting and gives a more accurate idea of the model’s performance on unseen data.

The tune was done with: 5-fold cross validation, 72 different combinations and Random Forest was trained and tested 360 times.

Then the best parameters was found and it gives a slightly better result, so I decided to re-execute the model with the best parameters, these are the results: