**Artificial Intelligence for Prediction in Agriculture**

**Final Report**

**Executive summary:**

Agriculture is the main source of national income for most developing countries. So with the use of Artificial intelligence we created a model which will help the farmers to choose crops according their soil types and also choose best time interval to sow using weather prediction.

Main goals of our project are:

* To find out why farmers are facing the complication during sowing time? Complication like unpredicted weather, poor soil condition, timing etc.
* Create the model using AI technologies which help farmer to analyse weather/soil/health of crop etc and save time and allow farmer to grow right crop in each season that has best yield.

To make a dataset, first we gathered information about crops in different regions of our country. From these data we made a dataset by our self and performed data pre-processing on that. To understand this dataset more closely, we made different visualisations by using various features of our dataset. After that, we performed EDA on our dataset to analyse the data. Then we applied different model approaches like K-Nearest Neighbours, Support Vector Machine, Naive Bayes, Decision Tree, Random Forest, Logistic Regression and Stochastic Gradient Decent to check which model suits better on our data. By studying results from above testing, we have concluded that K-Nearest Neighbours (KNN), Support Vector Machine, Decision Tree and Random Forest suit best on our data. The project has been created using PYTHON.

Our plan was to make this project precise, reliable and user friendly. For that we focussed on:

* Creating a database which includes features like soil type, weather reports, water availability, temperature etc.
* Creating an AI model which can be used in prediction of type of crops, time interval for sowing, water requirement and weather condition.

**Rationale Statement:**

The type of soil and nutrition of soil are some of the important factors that have a direct impact on crop yield and quality. With the change in climate condition and increasing pollution, it has become difficult for farmers to determine the right time for sowing seed. So with the help of Artificial Intelligence and Machine Learning, farmers can plan about the type of crop that can be grown and when should they start sowing the seeds.

**Problem Statement:**

In agriculture climate factors such as rainfall, temperature and humidity play an important role. Climate change can disrupt food availability, reduce access to food, and also affect food quality. So it has become difficult for farmers by using traditional methods of farming to make decisions about selection of type of crop and time to sow.

So in this application, we will get data from farmers about their soil type and then by using these data, our AI based model will provide information to them regarding which plant they should plant, how much water and fertilizer should provide, when to plant according to future weather condition.

**Data Requirement:**

The necessary data is not readily available based on our variables and prerequisites. Hence we will be making our own data sets to perform the task of agricultural prediction. During the whole process in formulating the Algorithm following variables with data values will be required:

1. Crop: This is one of the important aspects we are aiming to predict based on our model. We will predict the type of crop based on the temperature and many other variables which can be cultivated in that particular area.
2. Soil type: Different crops needs different soil types. If farmers plant crops in suitable soil types, then they will get good yield and hence good profit.
3. Area: It is one among the very important aspects for farming. Higher the area higher the plantation and so very high chance of good yield. This can be considered as a relatively linear relation. It will be in the form of numerical values in the units of Sq. Kms.
4. Requirement of Water: Based on this feature we will be able to train our model to decide that which crops require relatively less water or which crops can survive without much water intake. Its units will be in thousand litres based on area of the plot.
5. Temperature: This feature will help us to differentiate between the hot climate crops or cold climate crops. The units of this factor will be in the terms of ○Celsius.
6. Yield: This will be based on the all the above mentioned factors and the ultimate yield calibration in the units of tonnes. We aim to predict a much accurate amount of yield by maintaining its relationship with factors and ultimately higher the yield along with its quality better it is.
7. Investment: This must give as an input from the user based on which computing it along the line with all other factors a certain amount can be predicted and compared to user input in order to give suggestion the user as of which crop to invest in.

**DATA:**

For all the prediction purpose the key factor that does all the work is the relevant history associated with the database. The idea is to identify the type of crop which can efficiently be grown based on number of inputs. Therefore, our dataset included following variables as key features:

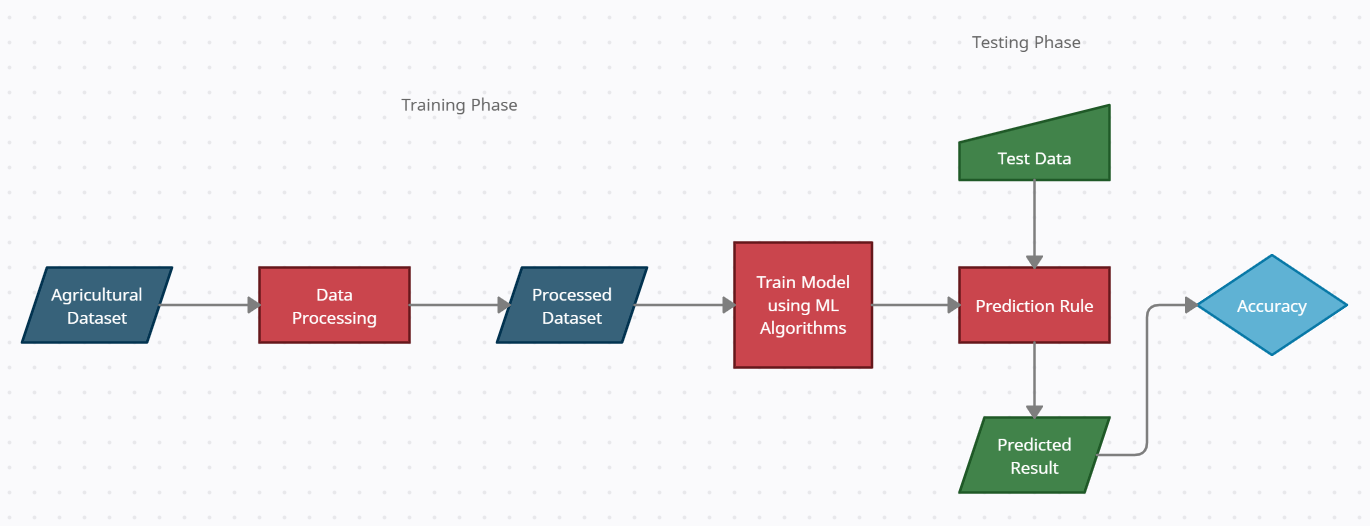
|  |  |
| --- | --- |
| **Field** | **Description** |
| **Crop** | Variable stating the type of crop |
| **Area** | Area in sq Kms with associated crop |
| **Soil type** | This shows different soil types for crops |
| **Water requirements** | Amount of water required in litres for crop cultivation |
| **Temperature** | Temperature determining the climate |
| **Yield** | Amount of output in quintal so that yield can be measured and predicted |
| **Investment** | Overall cost required including all the costs based on which output can be predicted against better costing |

There are certain datasets available on the kaggle platform and other, from which we will be taking reference.

1. [Dataset 1](https://data.world/thatzprem/agriculture-india)
2. [Dataset 2](https://www.kaggle.com/kamalkhumar/tamilnadu-crop-production-analysis?select=Tamilnadu+agriculture+yield+data.csv)
3. [Dataset 3](https://www.kaggle.com/srinivas1/agricuture-crops-production-in-india)

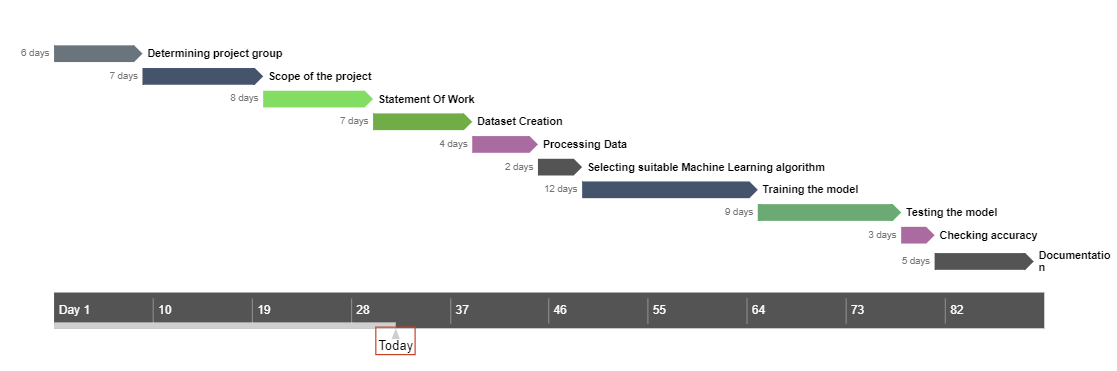
Above mentioned are the links to the datasets we will be referring to in order to form one proper dataset of our own. Moreover, certain data will be taken from govt. Websites and other trusted articles in order to achieve a clean data.

**Model Approach:**



First, we made agricultural dataset containing weather forecast, soil condition, water requirement, crop type etc. After processing the data, we generated the processed dataset which can be used in training purpose. Then we trained the data using machine learning algorithms and made prediction rule. We took the data to test. Finally, using prediction rule of trained model we predicted the result and we checked the accuracy of the model.

**Project Plan:**



In the above gannt chart, project plan timeline has been shown. Our team has worked with responsibility to complete this project alongside the timeline. Each of the team members has contributed equally to complete this project.

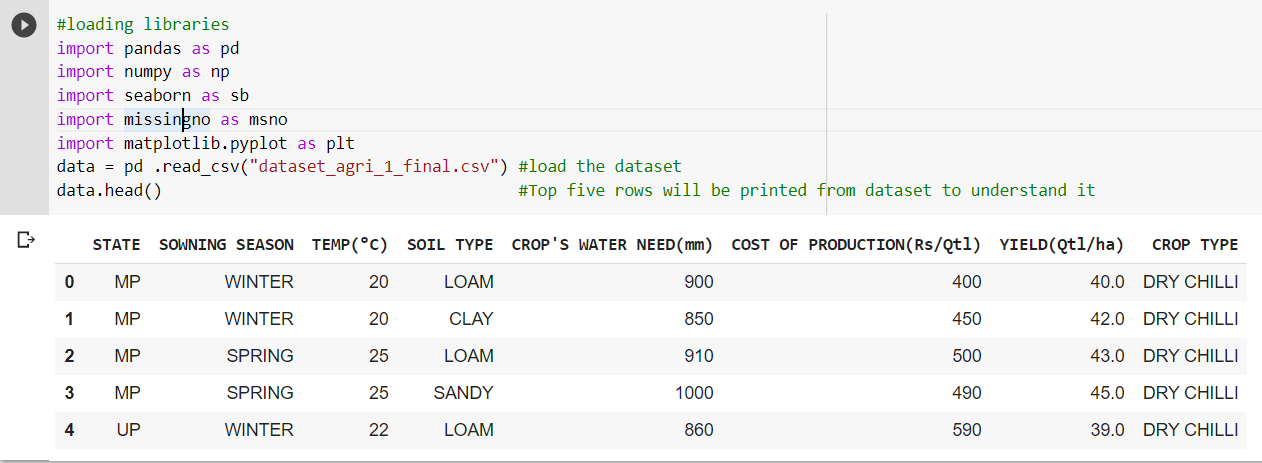
**Data Pre-processing**

Dataset is created manually from analysing the previous years’ agriculture data. Datasets has features like state, sowing season, temperature, soil type, crop’s water need, cost of production, yield and target crop type. With these features we are going to predict which crop type is best suited for give test data.

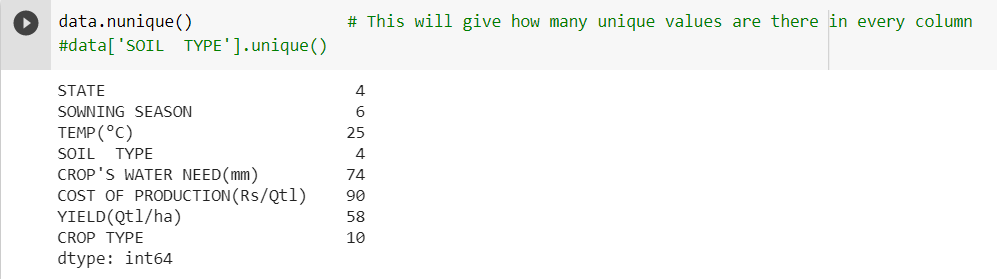
Data pre-processing include steps like data cleaning, data provisioning and data preparation.

1. **Data provisioning:**

* We have loaded our agriculture dataset in the python file. To know about data, we have printed top 5 rows of the dataset. So by seeing it any person can understand the dataset.

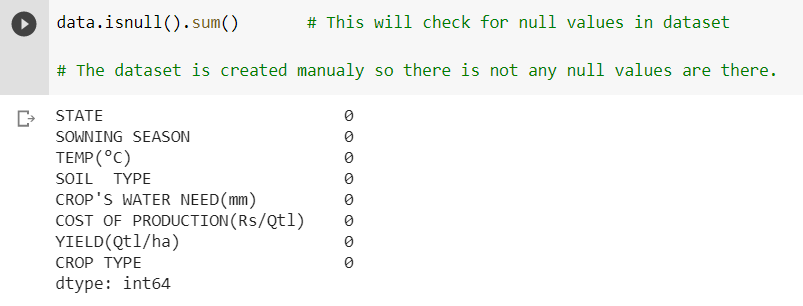


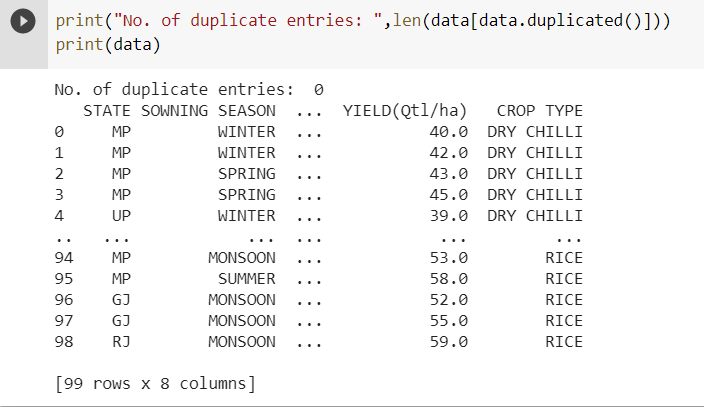
* We have also checked for how many unique values are there in every column.



1. **Data cleaning:**

* Since the data is created manually there are not any null values or duplicate values presented in dataset. Still we have check it for null values and duplicate values.





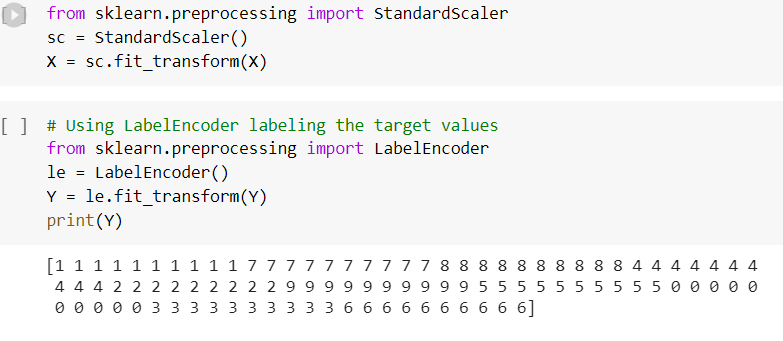
We have also printed whole dataset.

1. **Data preparation:**

* There are columns like sowing season, state and soil type were containing string values which are not used for predictions. So we have converted string values into integer values using onehotencoder.

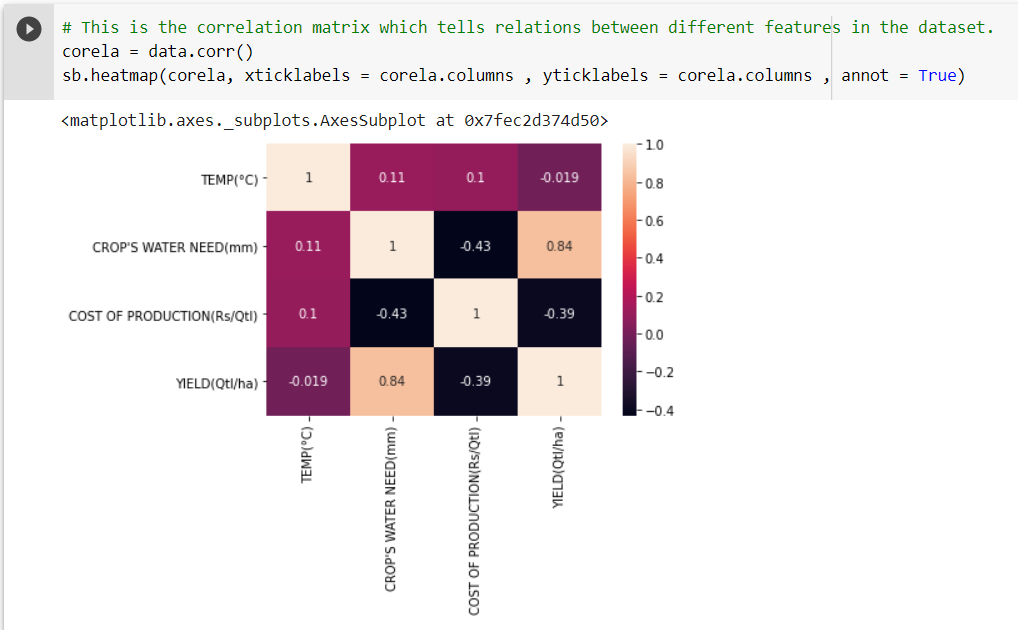


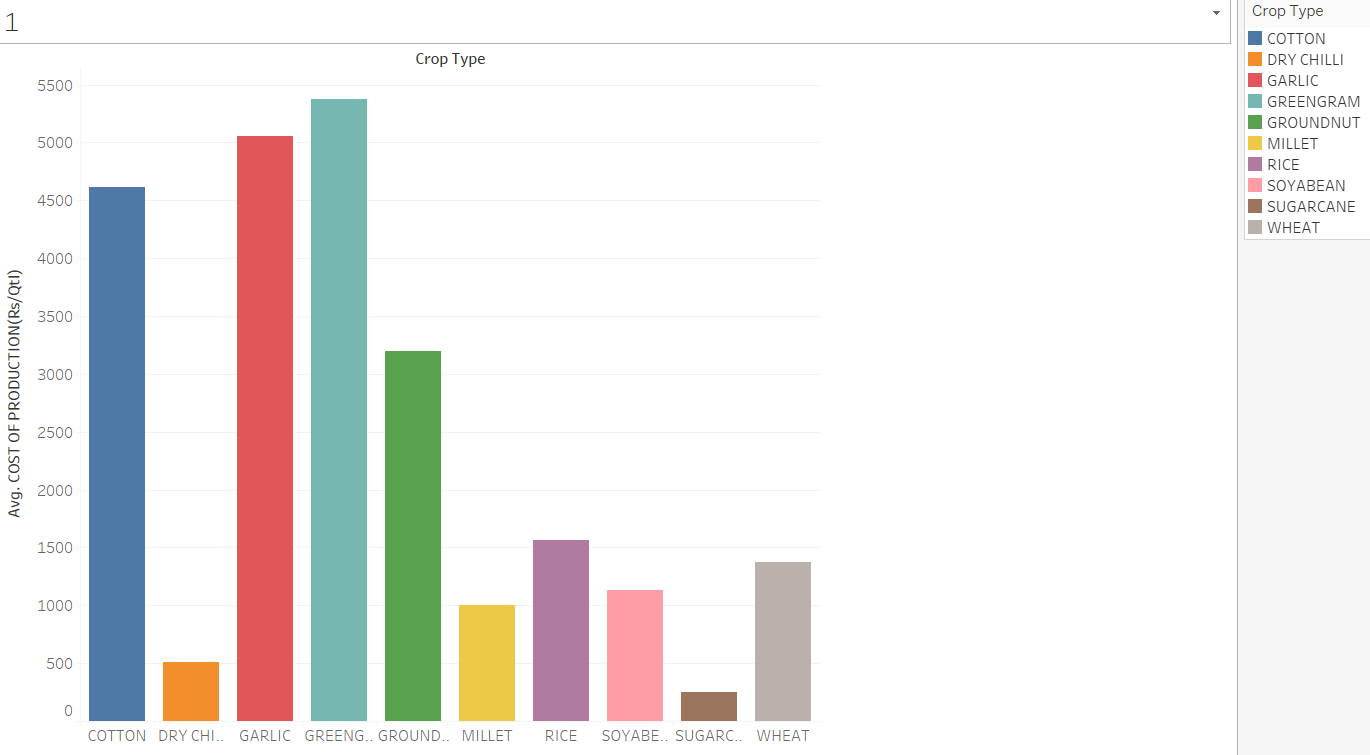
* We have converted target values into labels using label encoder.



**EDA: Exploratory Data Analysis**

Exploratory data analysis (EDA) is an approach to analyse and understand datasets to summarize their main features. It is often done by using statistical graphics and other data visualization methods.

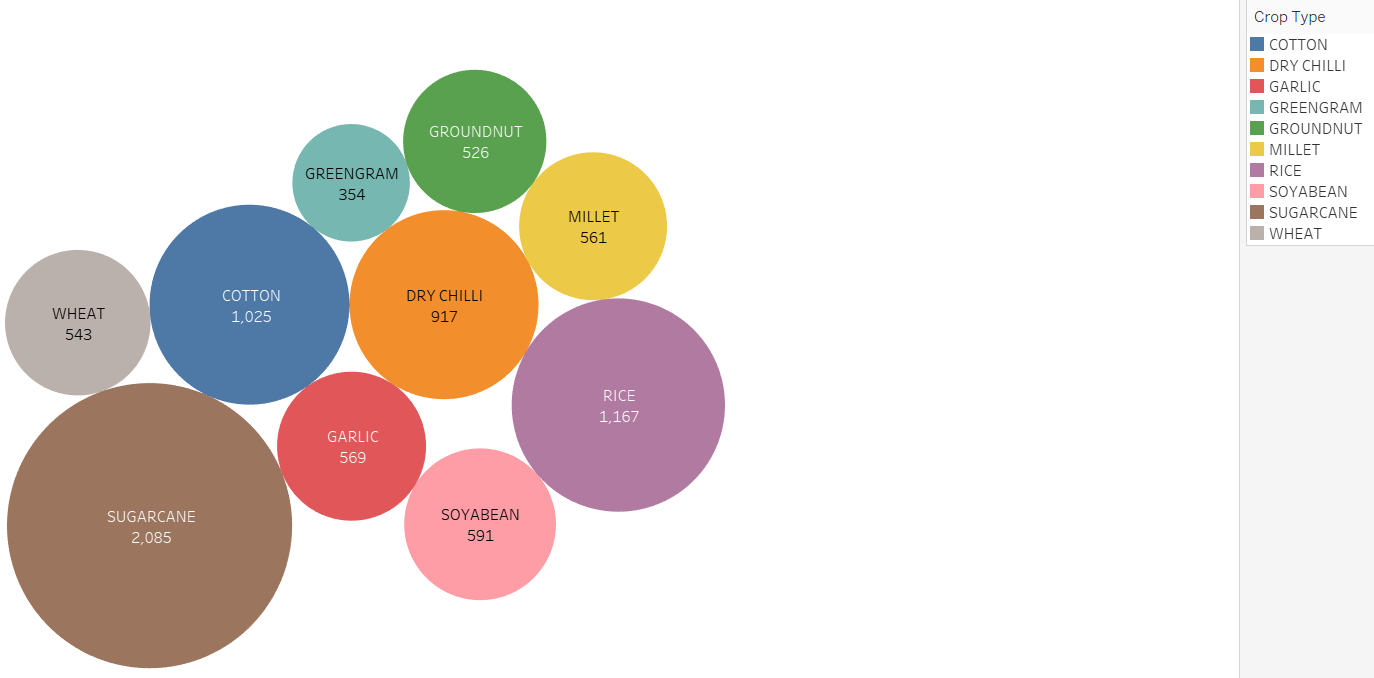
1. Correlation: Correlation is a measure of how two or more features are related to each other, in that sense, a change in one can cause a change in the other in the same or opposite directions. Its value is between -1 and 1.
2. We will see that, what is the cost of production for every crop type? So farmer can know which crop he should grow with his budget. The graph is created using tableau public.



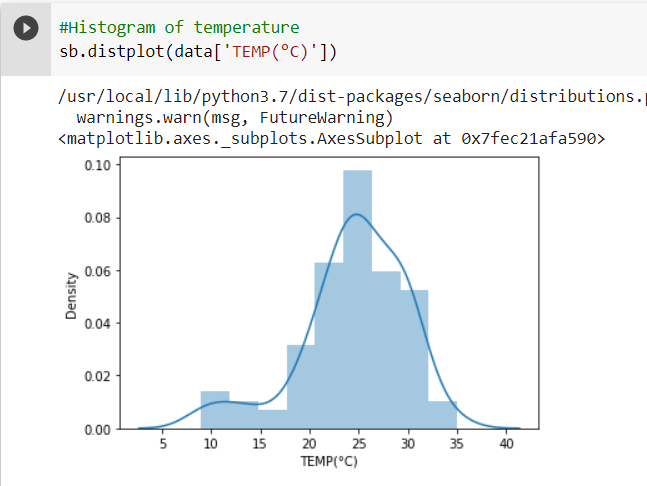
1. Following graph is the relation between sowing season and temperature for crop types. It is created using tableau public.

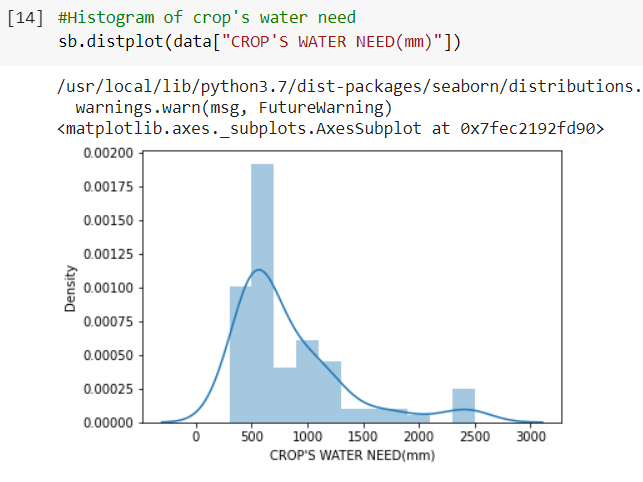


1. This graph is for the crop water need of every crop type.

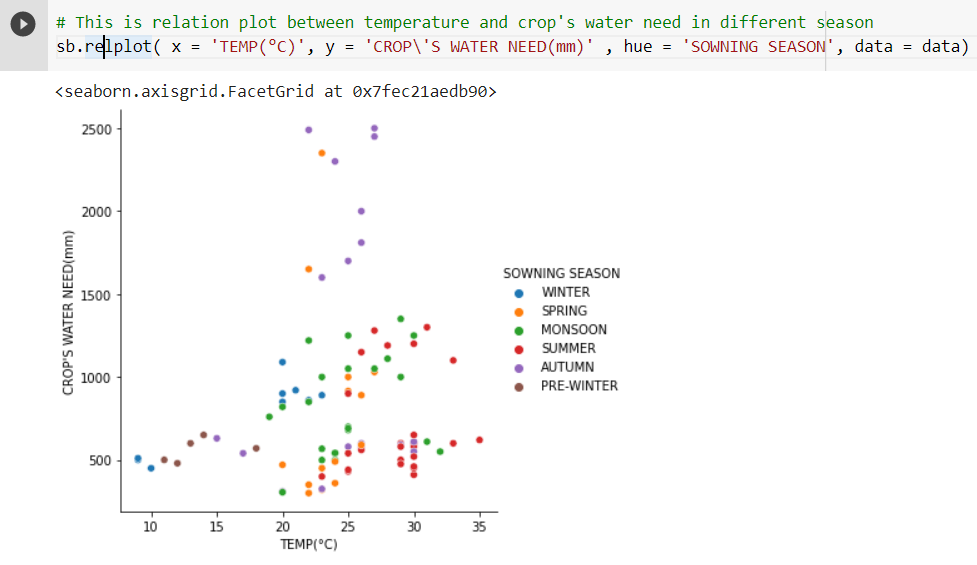


1. These are the histograms of temperature and crop water need. Graph is created using seaborn library in python.





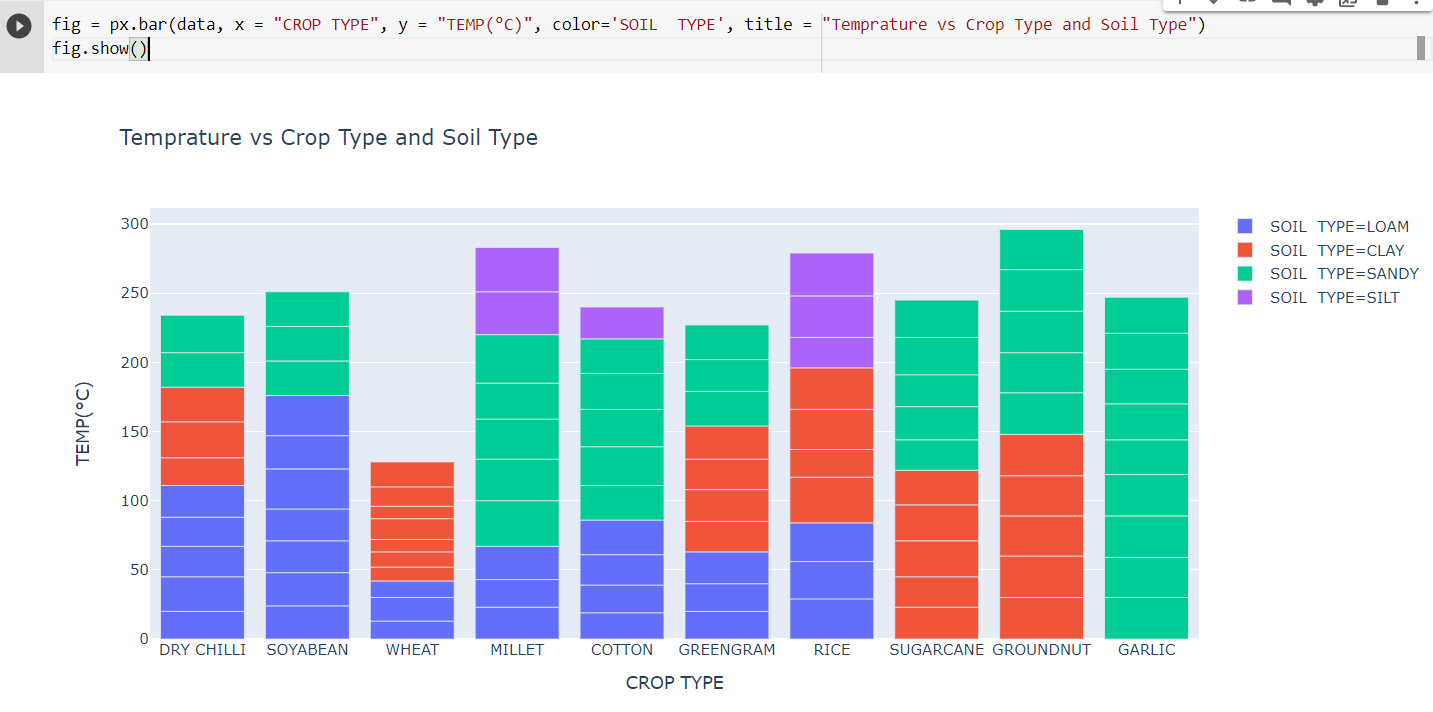
1. Scatter plot between temperature and crop’s water need for different sowing season.



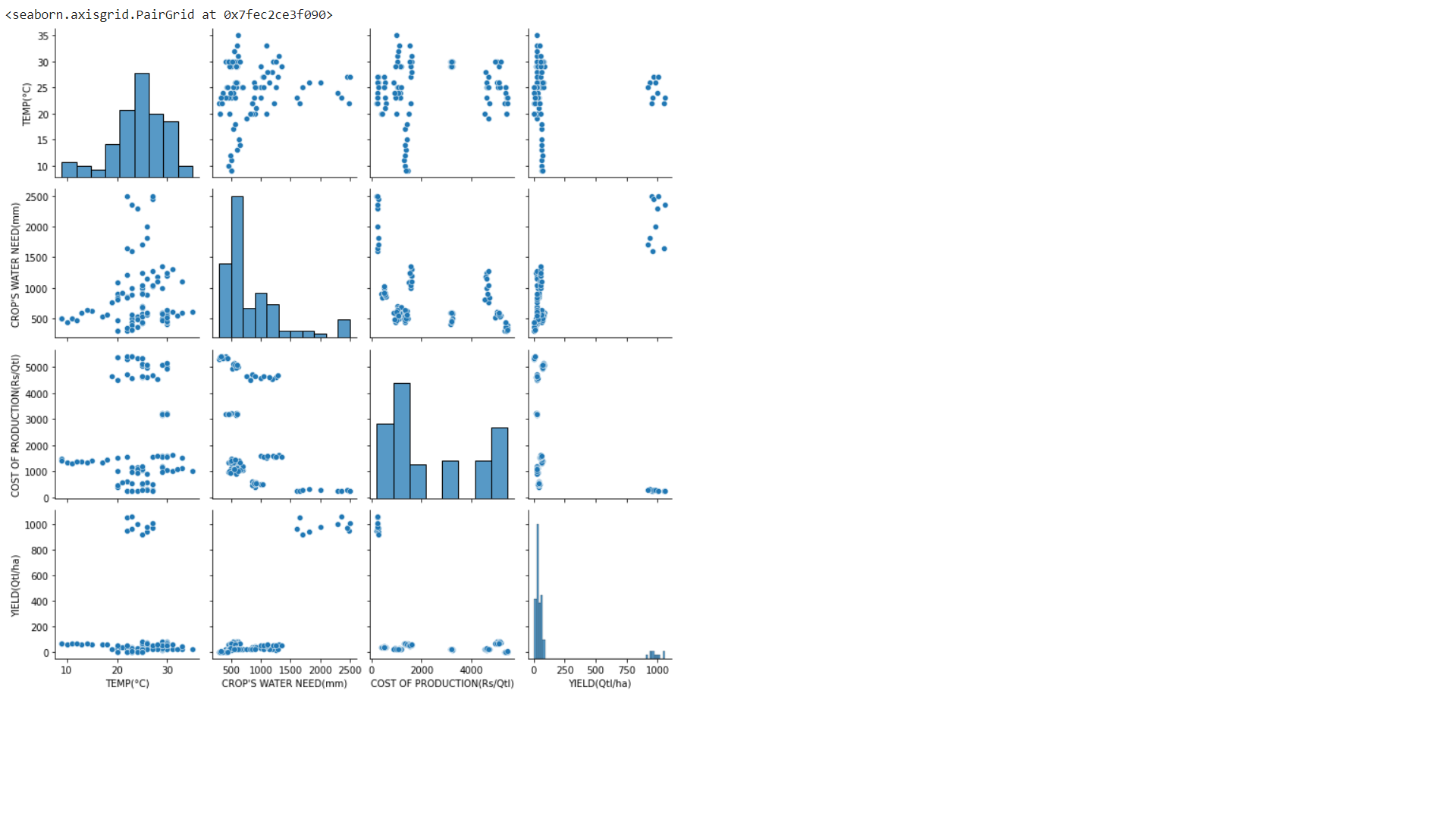
1. This graph is for state wise total seasonal yield.



1. Graph of temperature vs crop type and soil type using python.



1. Pair grid using seaborn in python.



**Modelling Approaches**

1. **Decision Tree**

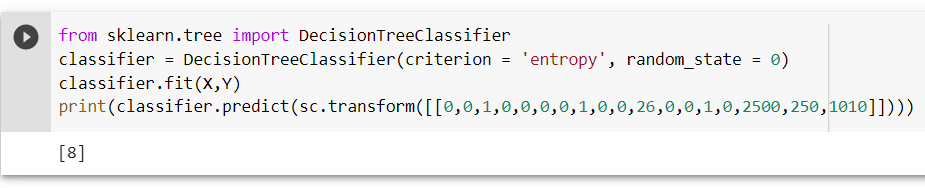
Decision tree algorithm is use for classification and prediction. This algorithm performs splitting on dataset and convert into small subset. This process calls recursive partitioning. Decision tree can handle high dimensional data.

Pros:

* A decision tree does not need data to be normalized or scaled.
* Decision trees require relatively less time and energy in preparing the data prior to process compared to other algorithms.
* In addition, missing values in the data have no significant impact on the decision tree-building process.
* A decision tree model is simple to understand and communicate to technical teams and stakeholders.

Cons:

* As compared to other algorithms, a decision tree's calculation can become very complicated at times.
* Even the slightest modification in data, the output results in a very large difference to decision tree’s basic structure directly resulting into its instability.
* The training time for a decision tree is usually longer.
* When it comes to applying regression and estimating continuous values, the Decision Tree algorithm falls short.
* Because of the difficulty and time required, decision tree training is relatively costly.



1. **Logistic Regression**

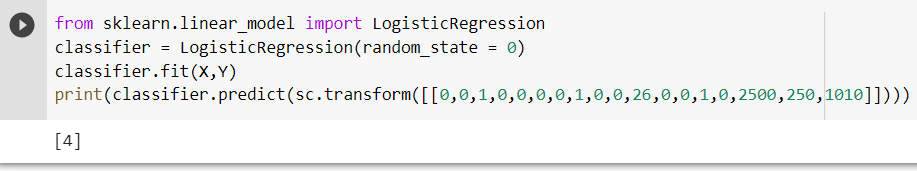
Logistic regression is one of the common classification algorithm with two possible outcomes. It is also known as binary classification.

Pros:

* The best perk of using LR is that it is more straightforward to apply, interpret, and train that does not require high computation power.
* Unlike decision trees or support vector machines, this algorithm allows models to be easily modified to represent new data. Stochastic gradient descent can be used to update the data.
* Logistic Regression doesn't make any claims about class distributions in feature space.
* Model coefficients can be interpreted as measures of function importance.
* The inferences about the value of each function are based on the expected parameters (trained weights). The association's course, positive or negative, is also defined. As a result, logistic regression can be used to determine the relationship between the features.

Cons:

* Because logistic regression has a linear decision surface, it is not efficient in solving nonlinear problems.
* The average or no multicollinearity between independent variables is needed for logistic regression.
* Logistic Regression should not be used if the number of observations is less than the number of features; otherwise, it should result in over fitting of the training dataset.
* Only essential and appropriate features should be used to build a model; otherwise, the model's probabilistic predictions will be inaccurate, and its predictive value will suffer.
* Complex relationships are difficult to capture using logistic regression. This algorithm is easily outperformed by more efficient and complex algorithms such as Neural Networks.



1. **Random forest**

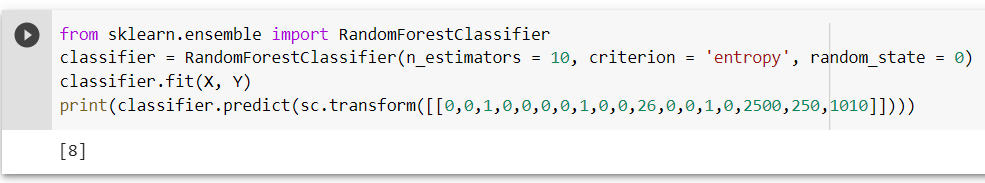
random forest algorithm is supervised learning algorithm used for classification as well as regression. As a name this classification algorithm works as a forest. It will select some random result from given dataset then decision tree is constructed. After that voting will be performed for every result and most voted result is final result.

Pros:

* Both categorical and numerical data function well with Random Forests. In most cases, no scaling or transformation of variables is needed.
* Random Forests are capable of handling both linear and non-linear relationships.
* It has a good system for estimating missing data and keeps its precision even though a lot of data is missing.
* Random forest is capable of solving both classification and regression problems, as well as providing reasonable estimation in both cases.
* The above capabilities can be applied to unlabelled data, resulting in unsupervised clustering, data views, and outlier detection.

Cons:

* For large datasets like agriculture, Random Forests may be computationally intensive.
* Random Forests are difficult to understand. They give you a sense of how important a function is, but they don't give you as much insight into the coefficients as linear regression does.
* Random forest is similar to a black box algorithm in that you have no control over the model's action.



1. **Support Vector Machine**

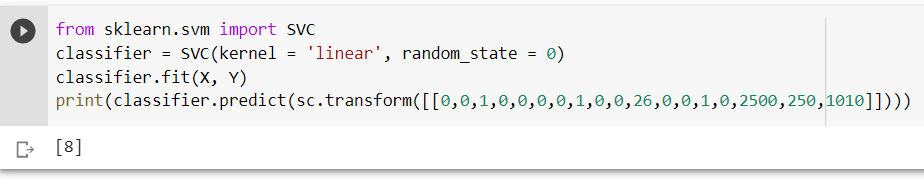
Support Vector Machine algorithm is supervised machine learning algorithm, which is also used solve classification and regression problems. Support Vector Machine takes data as input and generates the hyperplane as output that differentiate two classes accurately.

Pros:

* SVM is more effective when the dimension space is high and SVM is also memory efficient.
* It’s mostly used classification problems and works more accurately when classes are separated with clear margin.
* It is more effective when number of samples is less than number of dimensions.

Cons:

* SVM does not work efficiently when the data is large and noisy.
* SVM is also generate inaccurate results when number of features for data points are higher than number of training data samples.



1. **Stochastic Gradient Decent**

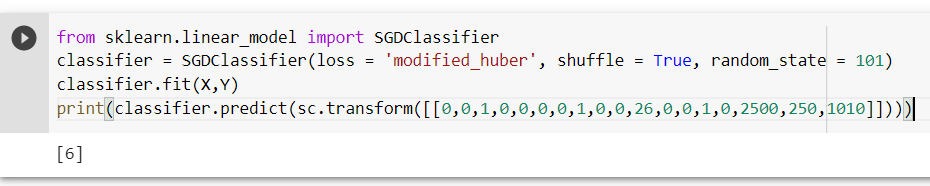
this algorithm is also known as random probability gradient descent .it will take some randomly selected sample instead of whole dataset. SGD is generally noisier than typical Gradient Descent.

Pros:

* It is easier to fit into memory as single observation is processed by network.
* It is faster than batch gradient descent as it will take less time to reach the minima.

Cons:

* It may produces wrong decision because of frequent updates.
* It does not support vectorization.



1. **Naive Bayes**

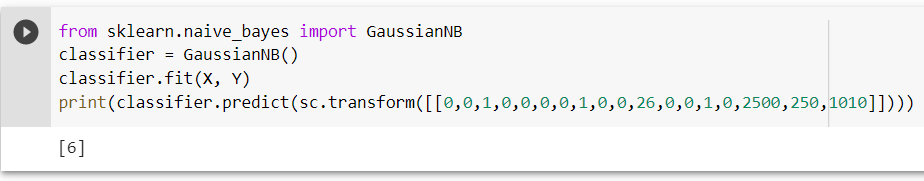
This algorithm based on bayes’s theorem. Naive Bayes model is easy to build and particularly useful for very large data sets. Along with simplicity, Naive Bayes is known to outperform even highly sophisticated classification methods.

Pros:

* It is easy and fast.
* Use for multiclass prediction also.
* Naive bayes also works well in less training data.
* It performs well in case of categorical input variables.

Cons:

* If categorical variable was not observed in training dataset this algorithm will unable to make a prediction.
* This algorithm known as bad estimator.



1. **K – Nearest Neighbours**

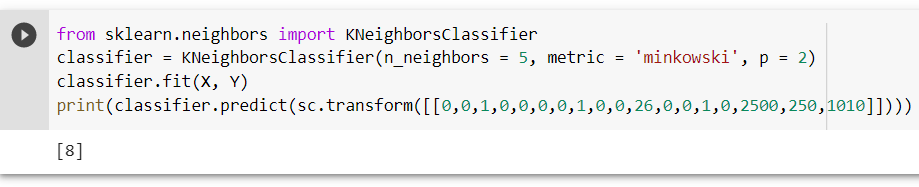
K-Nearest Neighbour algorithm is supervised machine learning algorithm. It is mostly used for classification problems. It takes new case and available cases and then put new case into most similar case from available cases.

Pros:

* It is very simple algorithm.
* It is very effective when we have large and noisy data.
* It can also be used to regression problem.
* During classification problems, classes do not have to be linearly separable.

Cons:

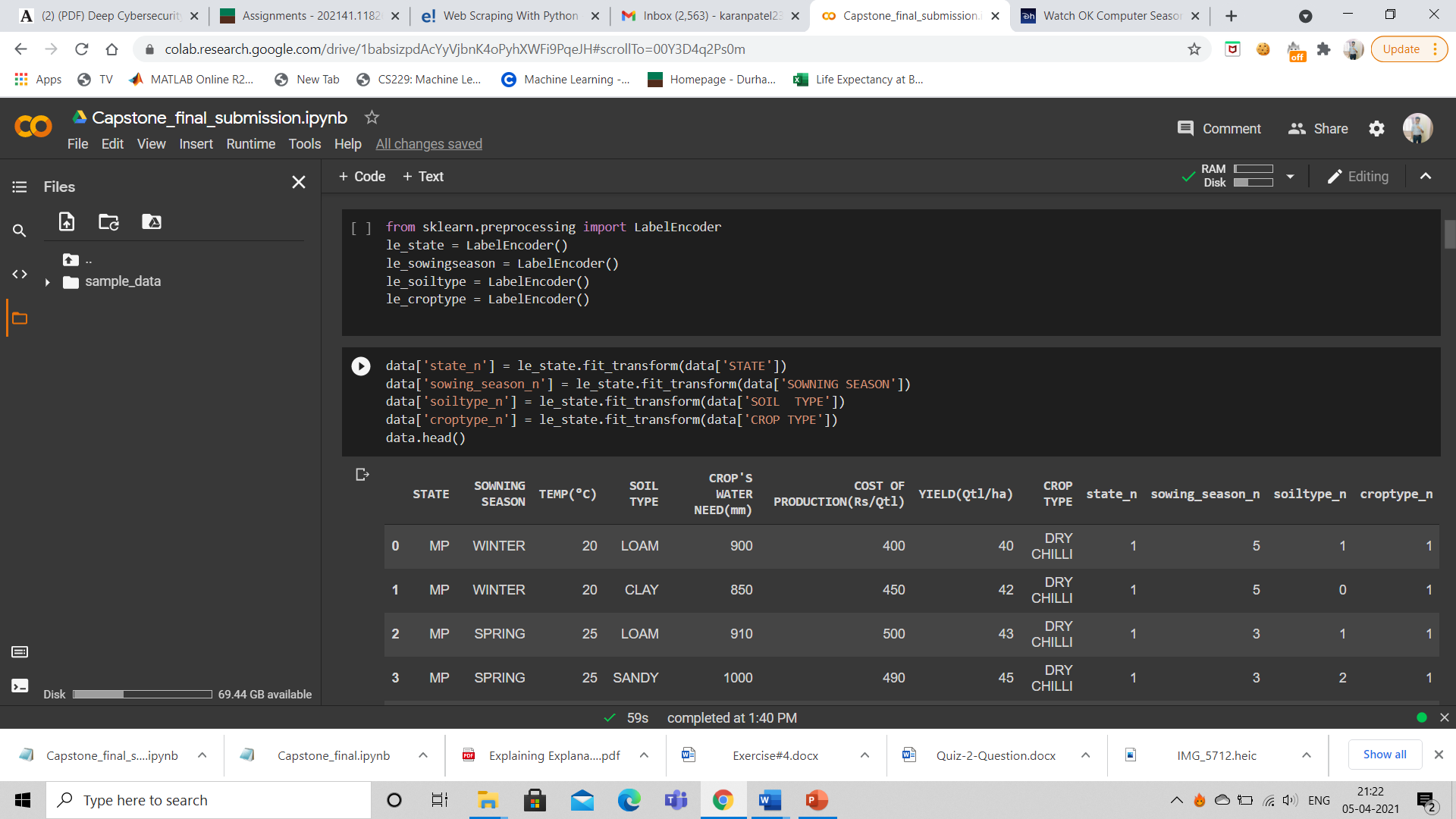
* It is also known as Lazy Learner Algorithm as it does not learn from training data immediately.
* K-NN is also known as Non-Parametric Algorithm because it does not make any assumption on underlying data.
* Its computation cost is high.

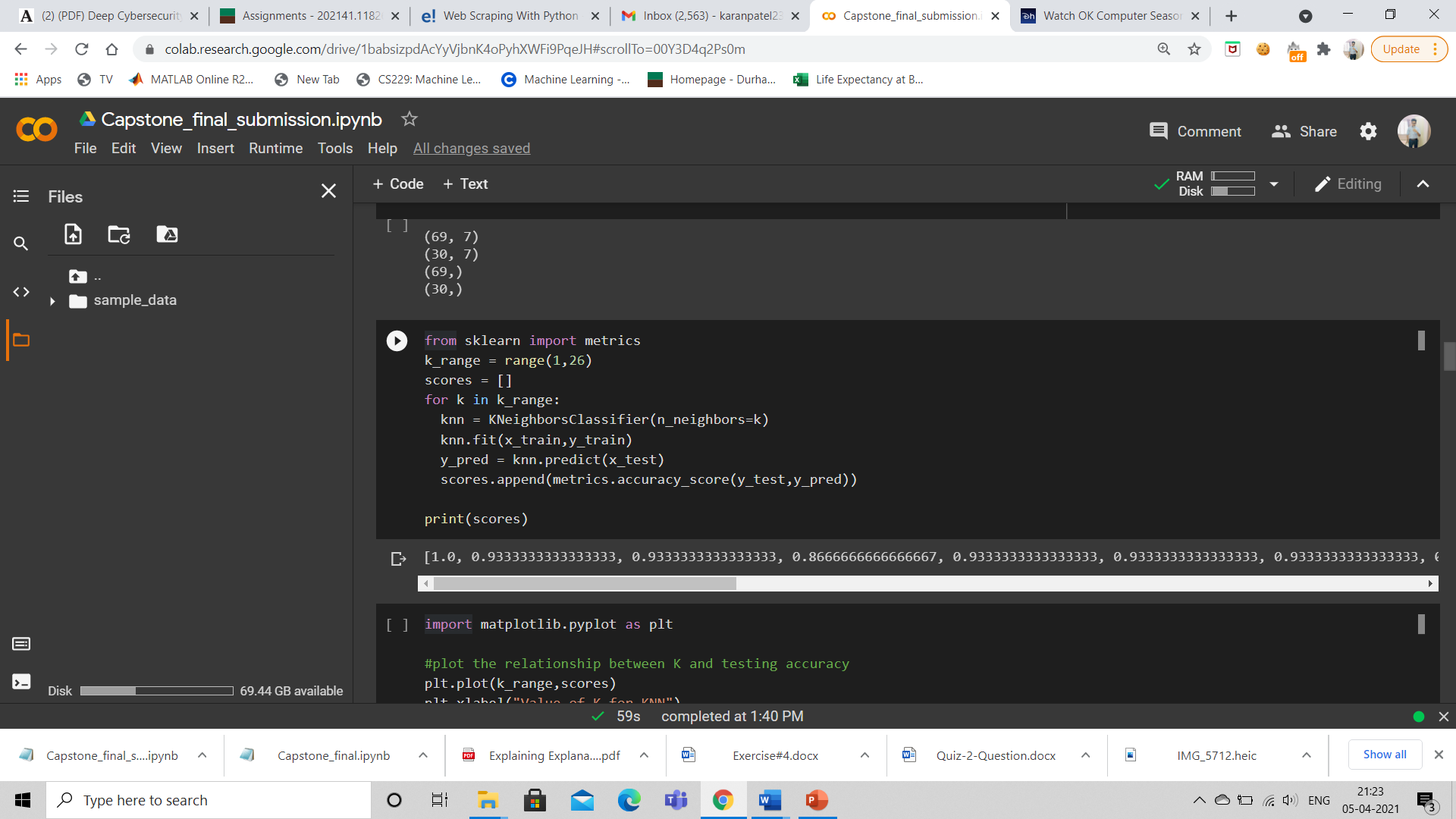


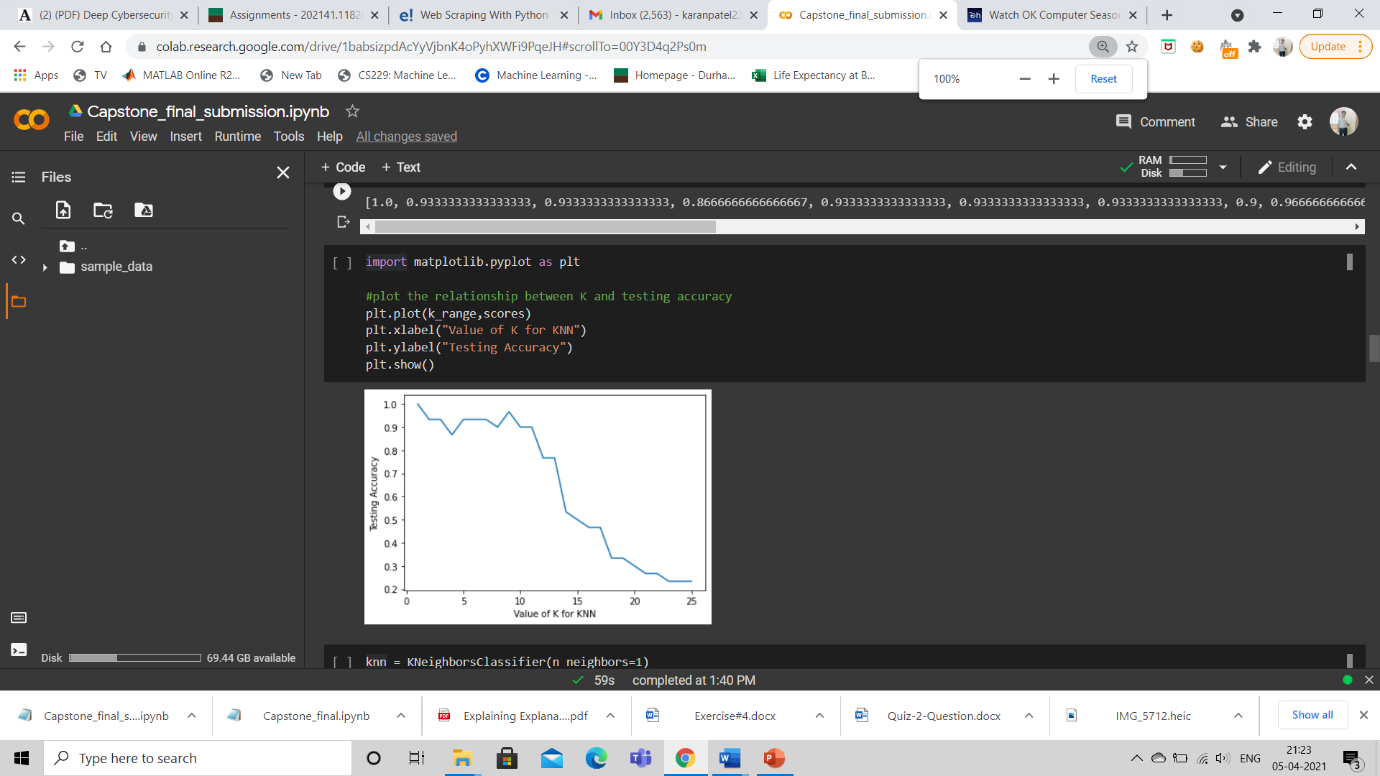
We have performed above seven algorithmsour dataset. For the test the algorithms should have given output: [8]. [8] means SUGARCANE. As you can see Decision tree, Random forest, KNN and Support vector machine are giving right prediction results. Hence these four algorithms are best suited for our datasets.

**INITIAL MODELING RESULTS:**

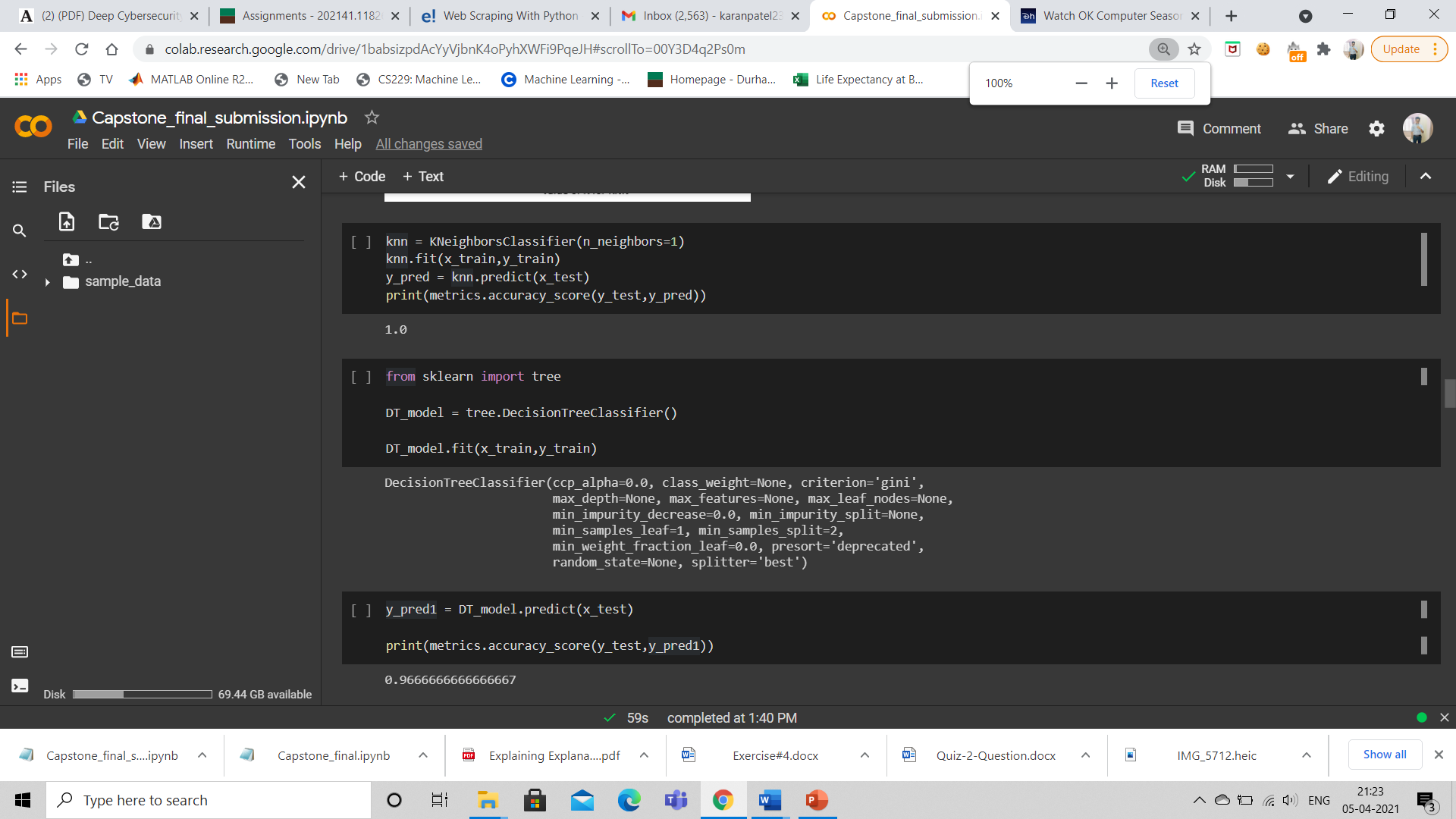
According to previous report we decided to perform KNN and decision tree algorithms for our modelling purpose. First we performed KNN classification algorithm. To perform this algorithm first we need to encode our string data to integer form. We used label Encoder class to encode our categorical data.

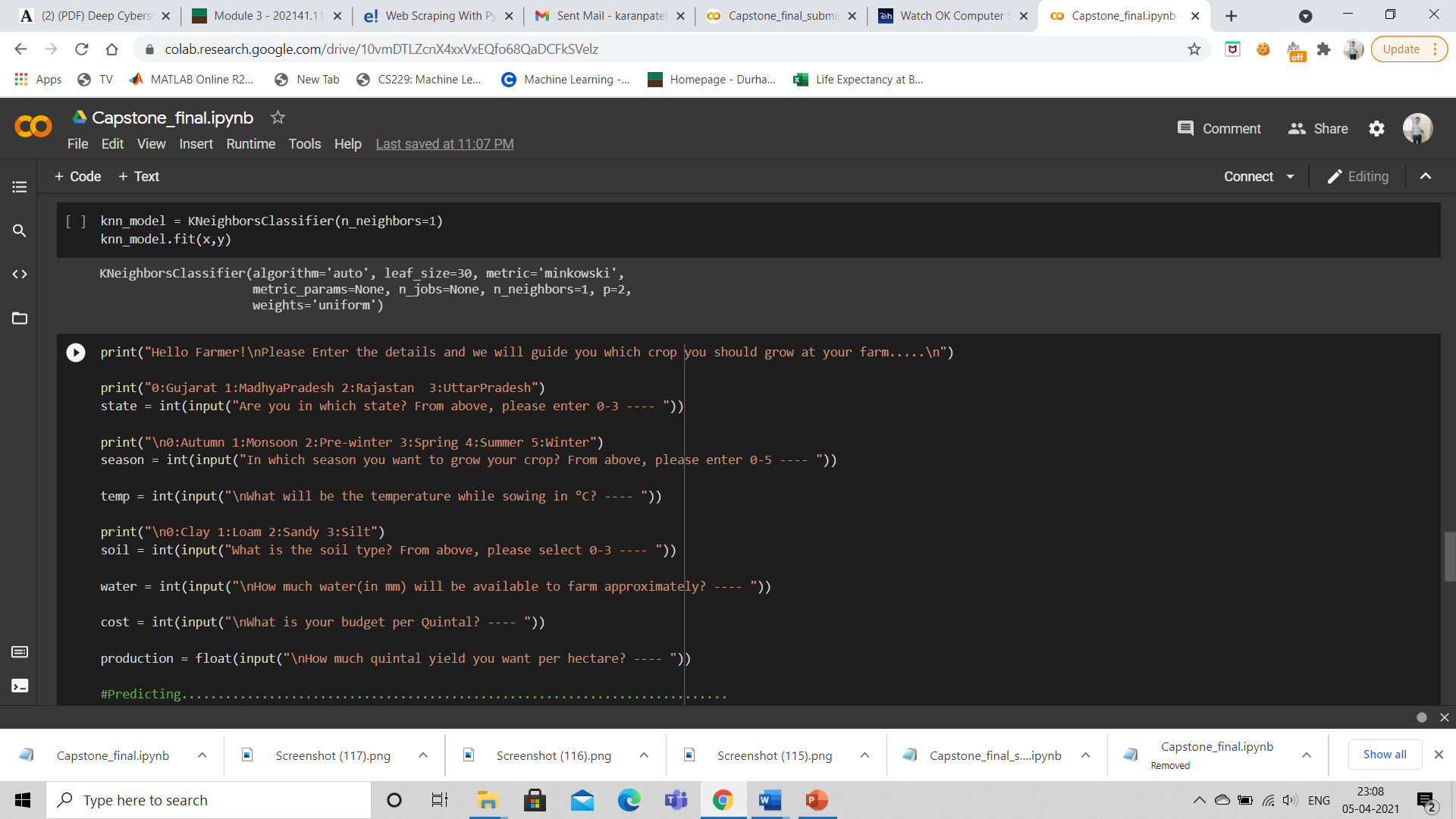


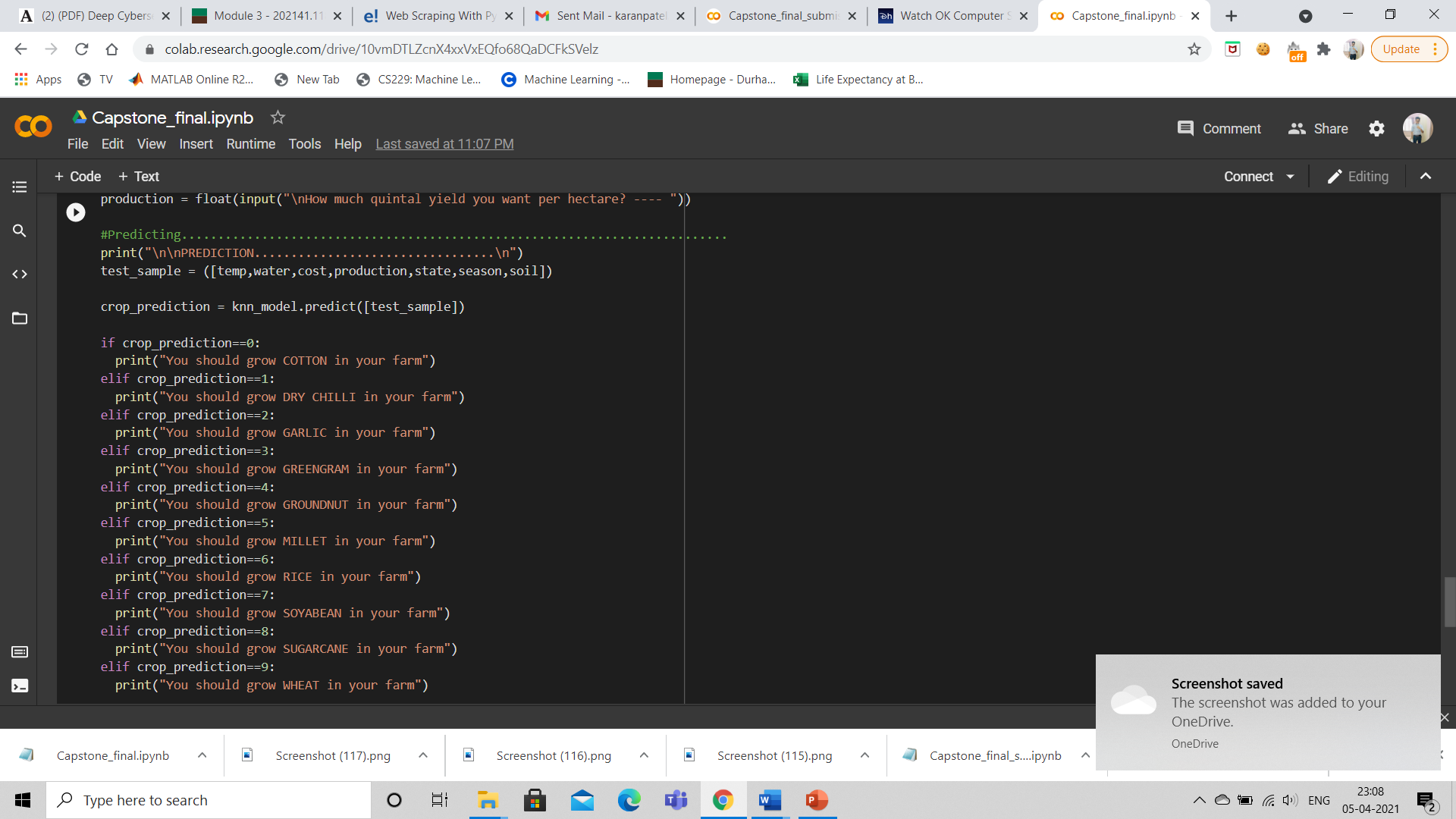




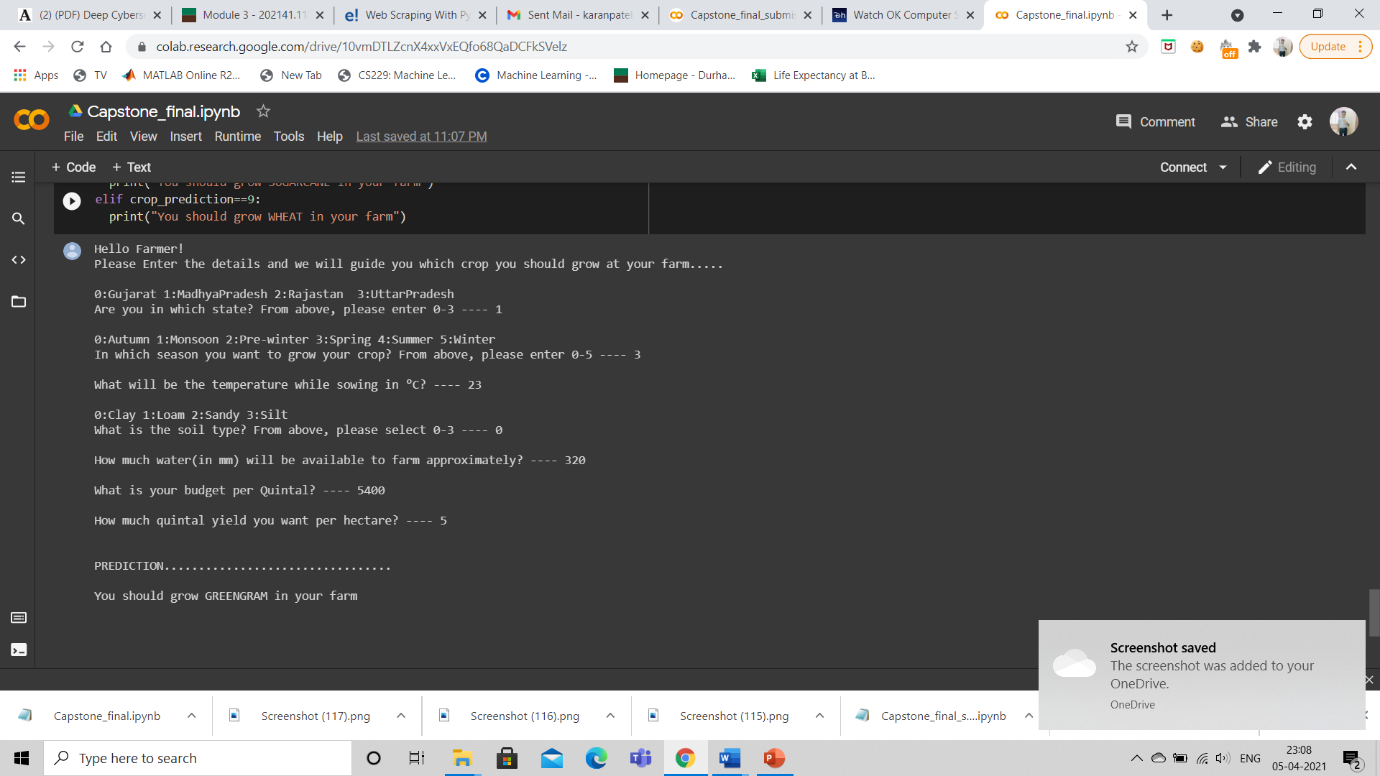
We performed accuracy metrix for KNN classifier and decision tree and we got 1 accuracy means 100% accuracy for KNN algorithm. We got 0.96 means 96% accuracy for decision tree algorithm. So we decided to go with KNN classifier algorithm.







Using KNN we developed our basic prediction system. We took crop type as our independent variable and remaining as a dependent variable. As we can see in screenshot, farmer entered all information and he got crop type as a final output.

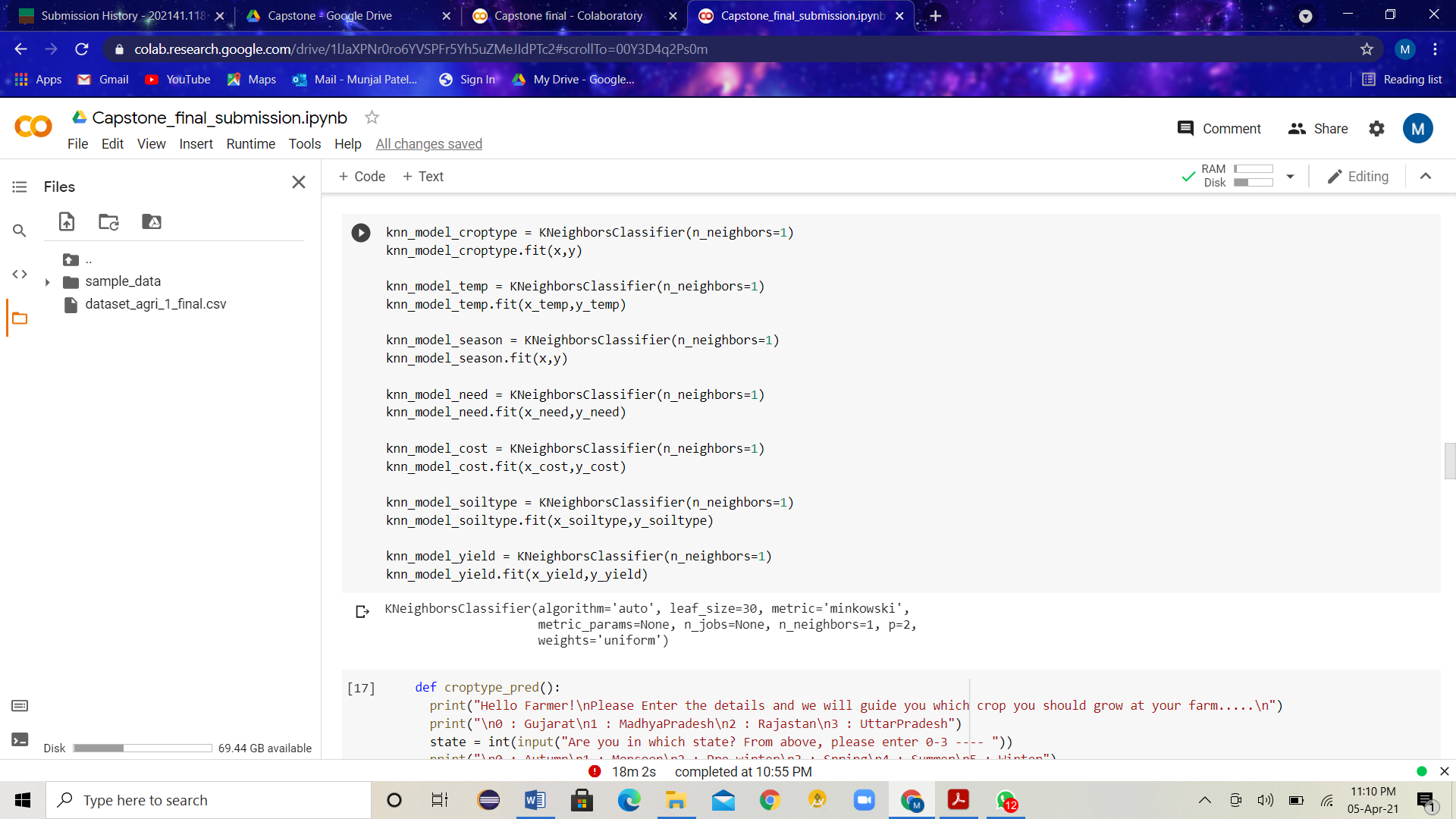


Till now our project was to predict best suited crop type according to different features. We wanted to take our project at further level and we wanted to make recommendation system for farmers.

In this recommendation system farmer will be able to choose what information they need and the system will make that feature as independent variable. After that, based on the dependent variables the system will give the prediction for required information.

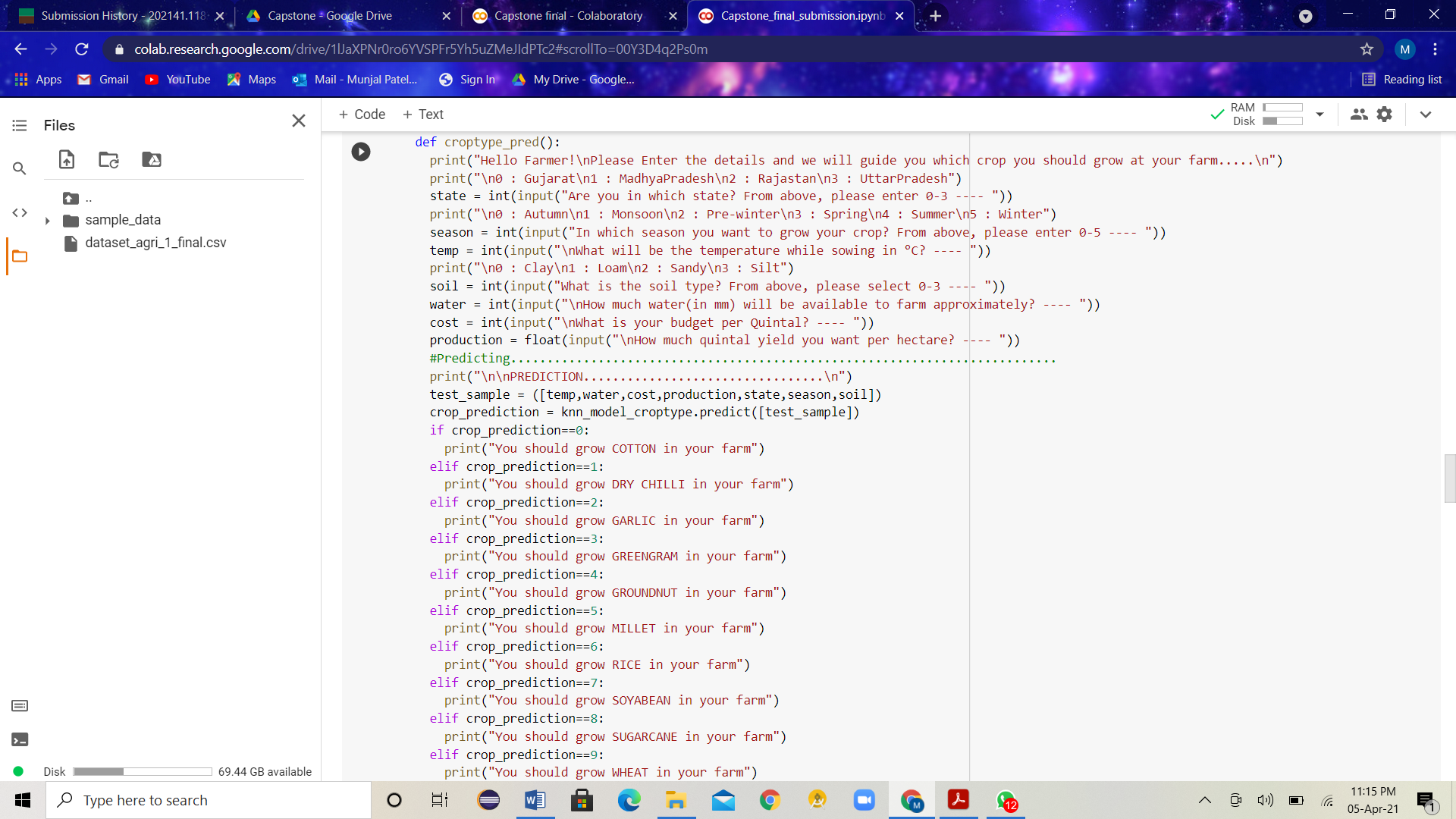
**Recommendation System:**

Using KNN we trained the models for every features which our recommendation system can predict.

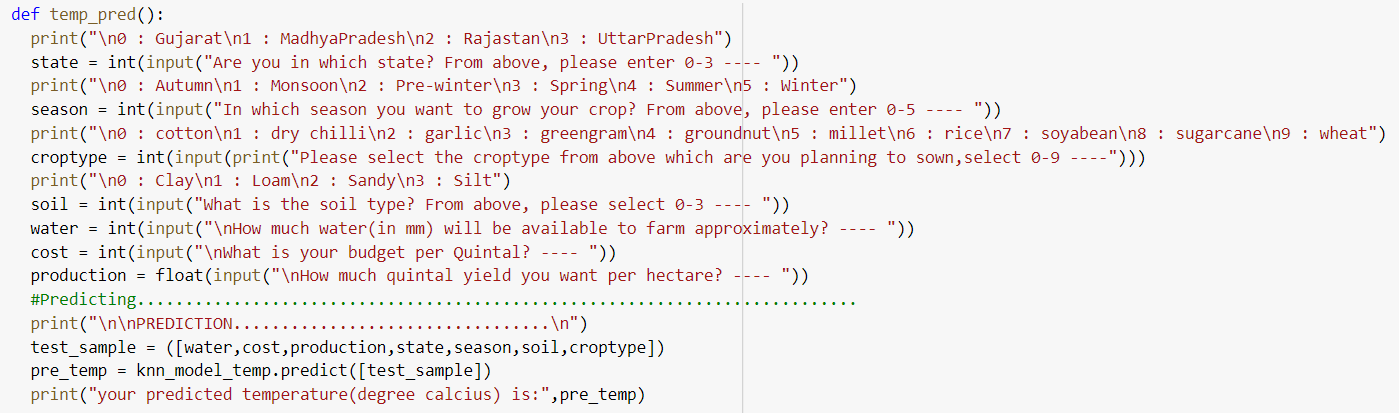


We created functions for every different feature which can be predicted by our System.

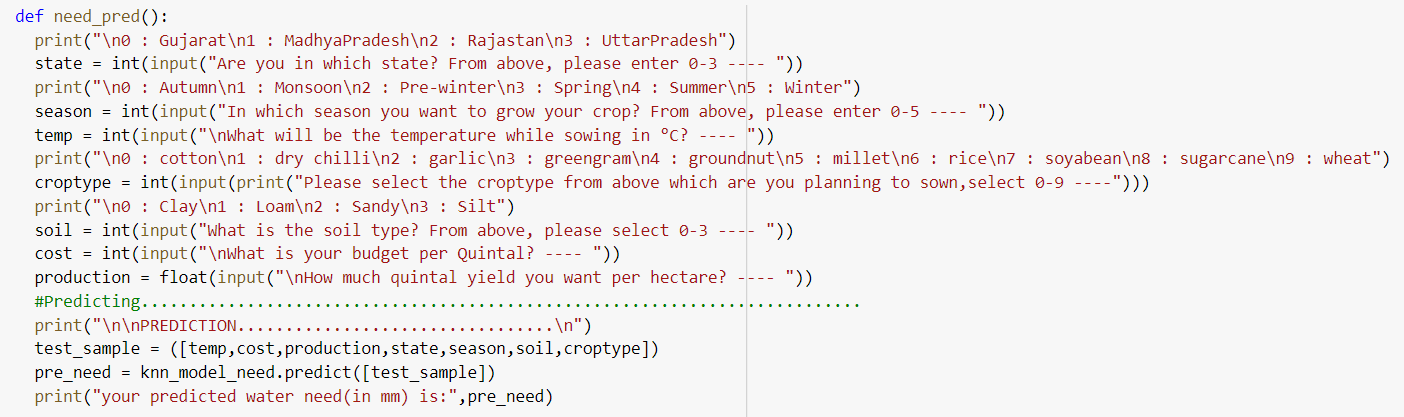
Function for ‘crop type’:



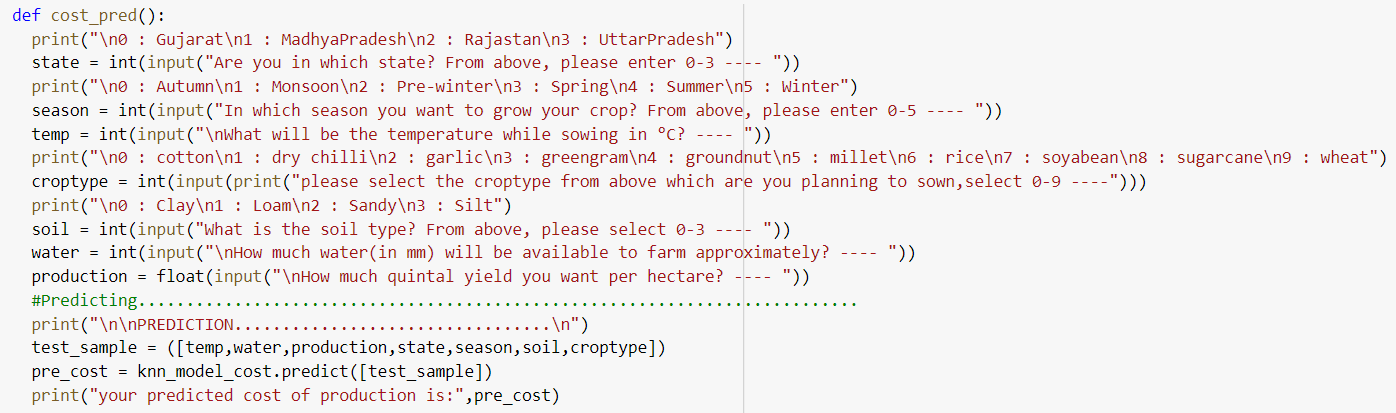
Function for ‘temperature’:



Function for ‘crop water need’:



Function for ‘cost of production’:



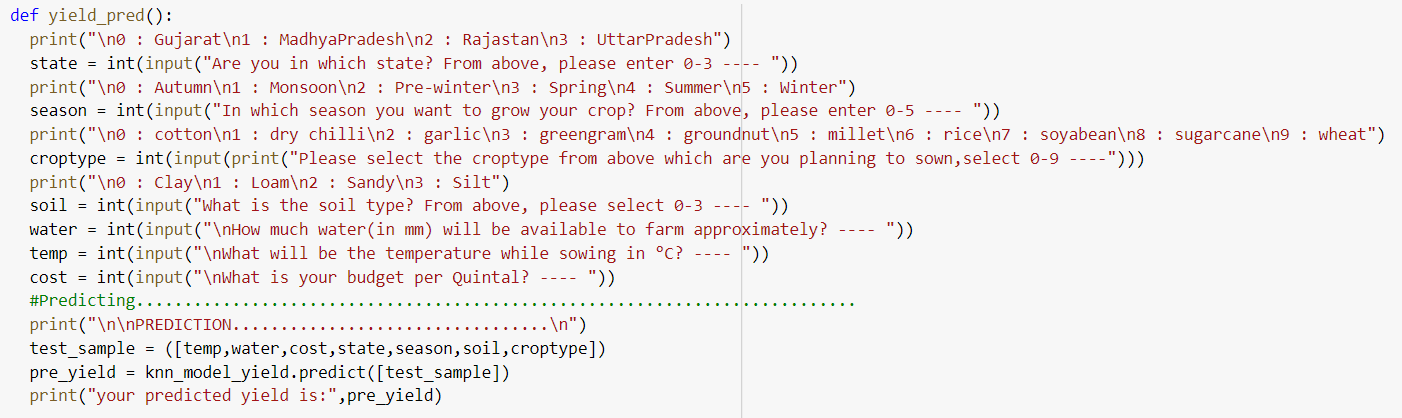
Function for ‘season’:



Function for ‘soil type’:



Function for ‘yield’:

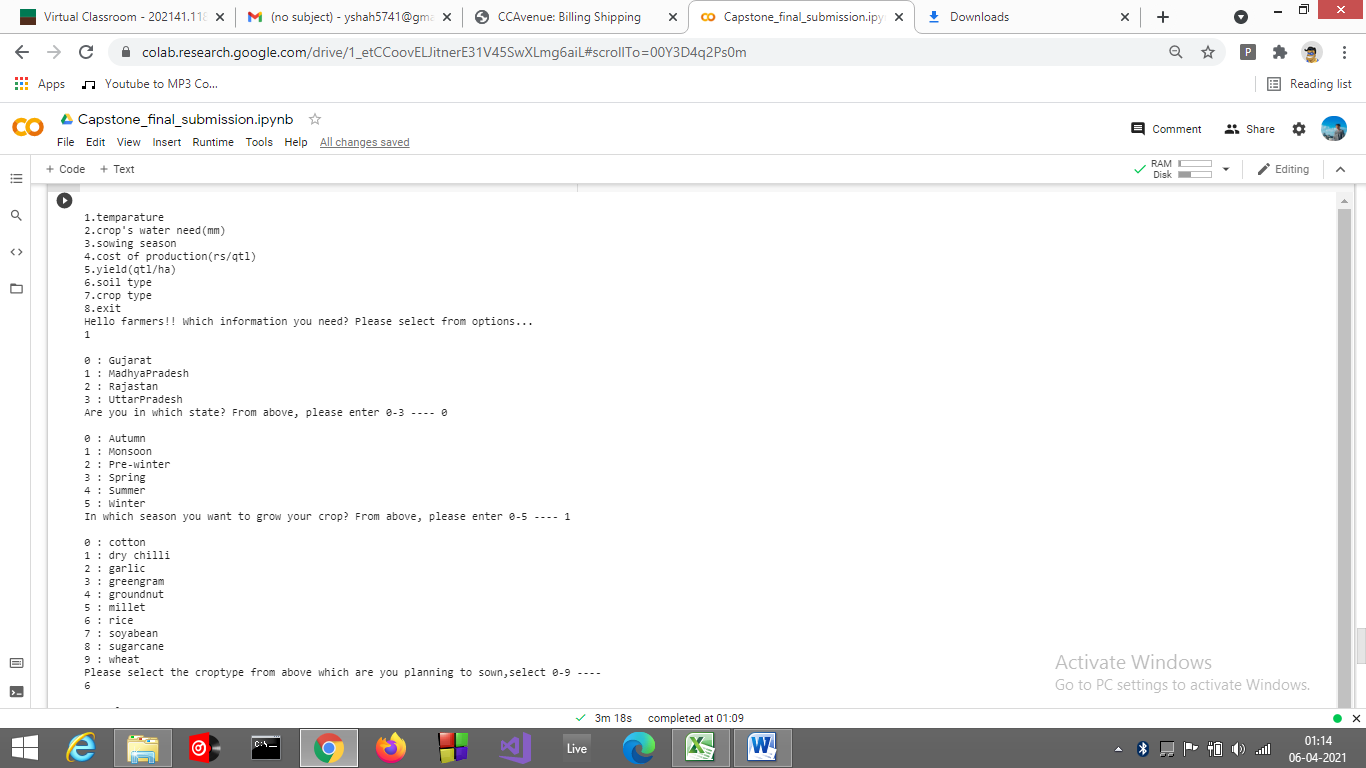


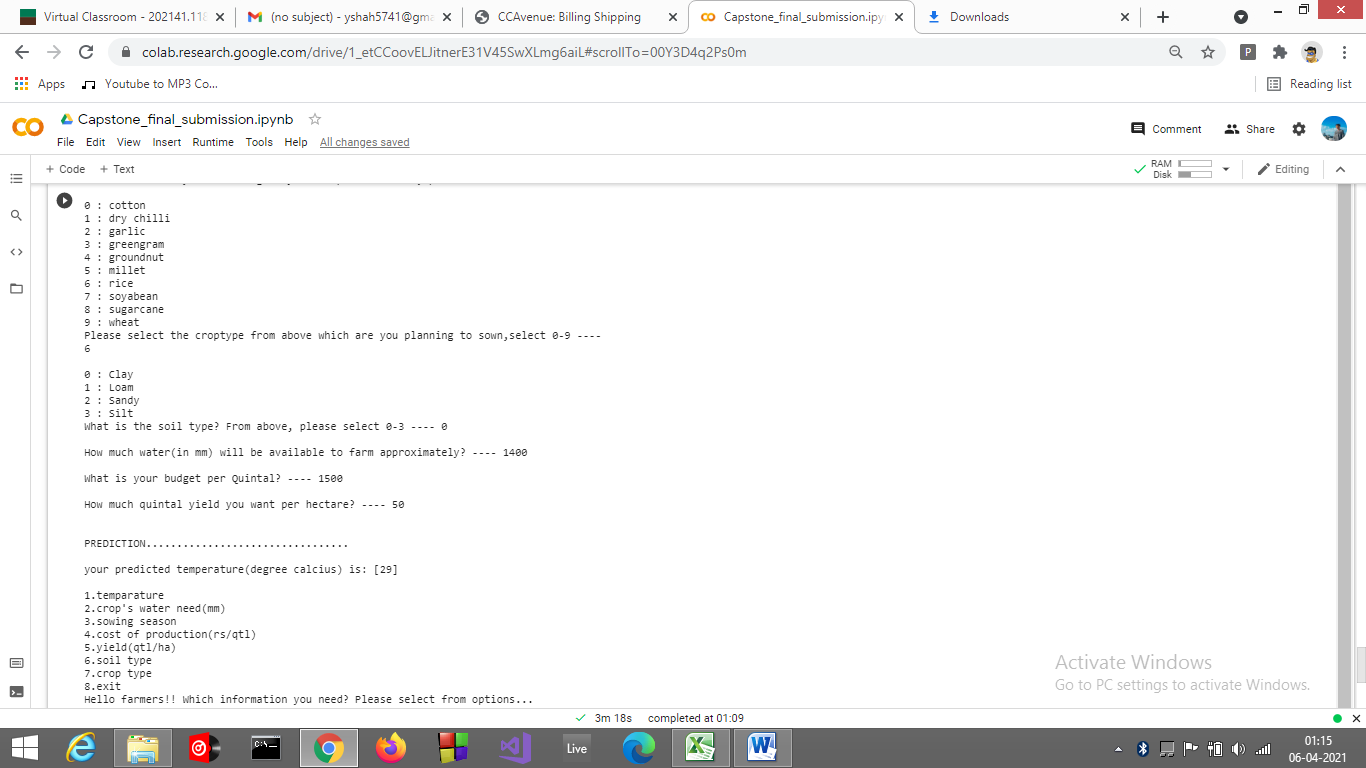
Using while loop we will ask the farmer to choose which information he needs.



**Test samples:**

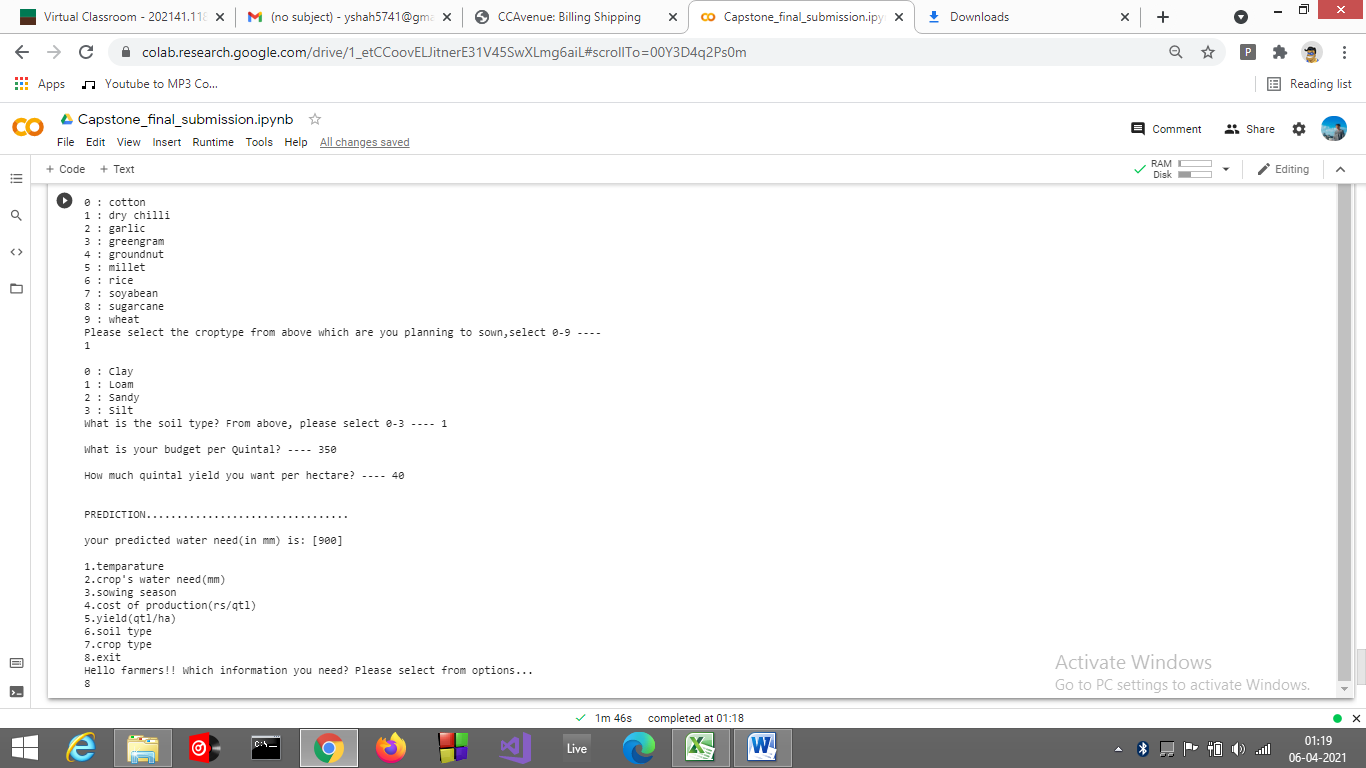
1. What is the best suited temperature for this farmer?



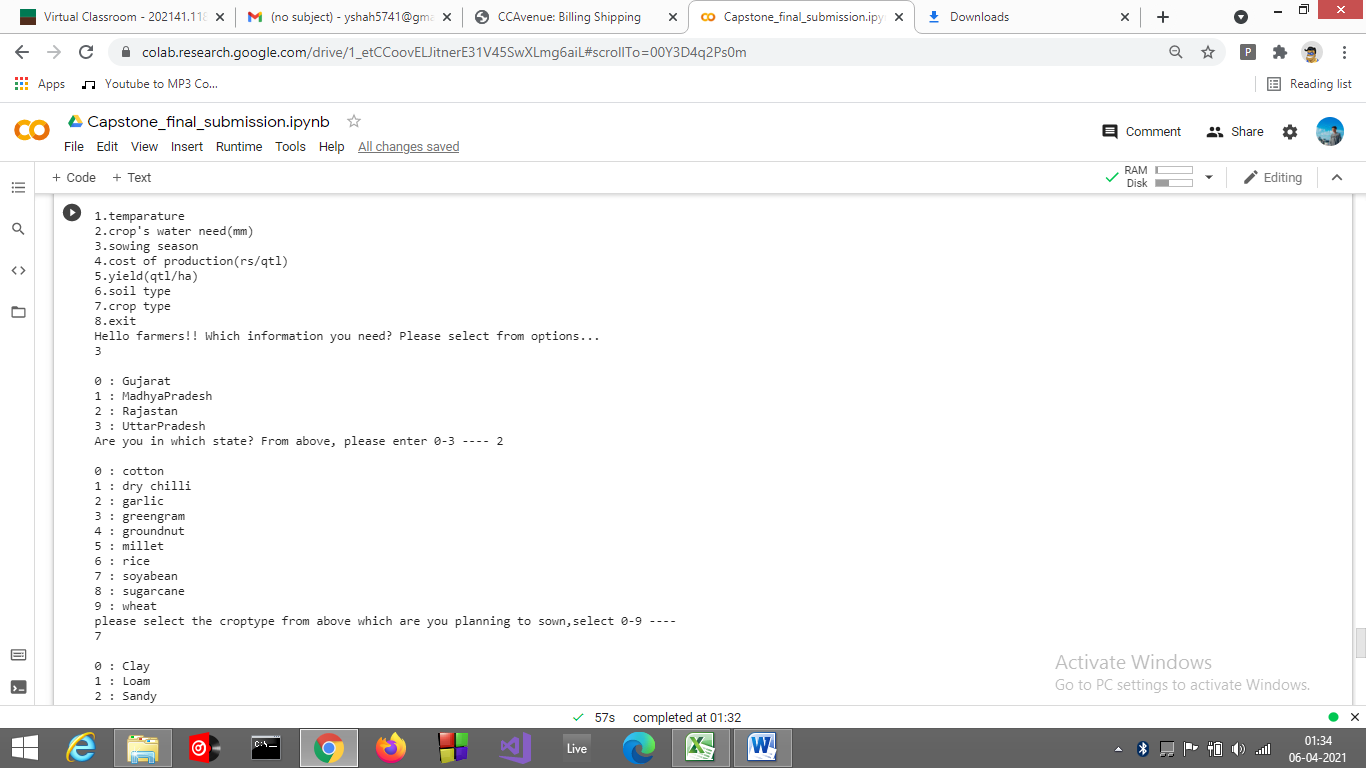


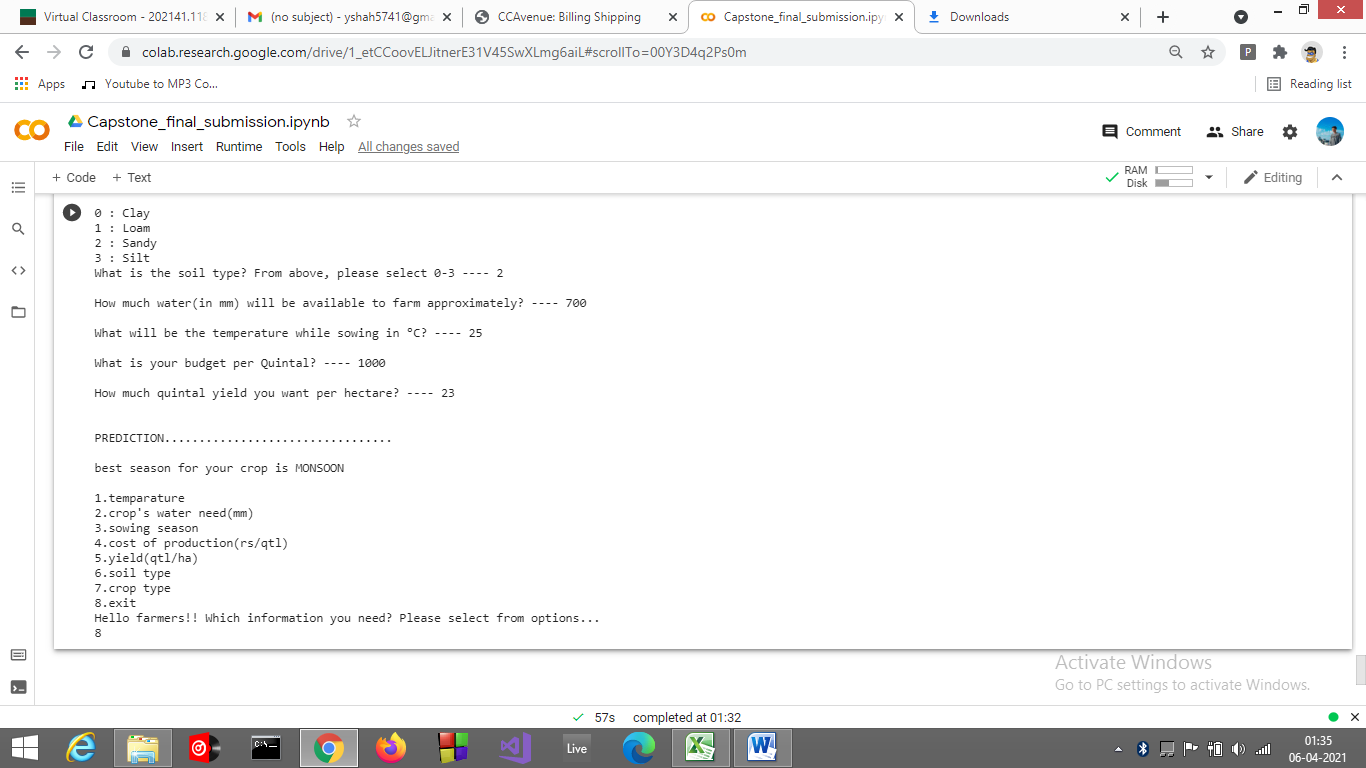
2. How much will be the need of water for this farmer?





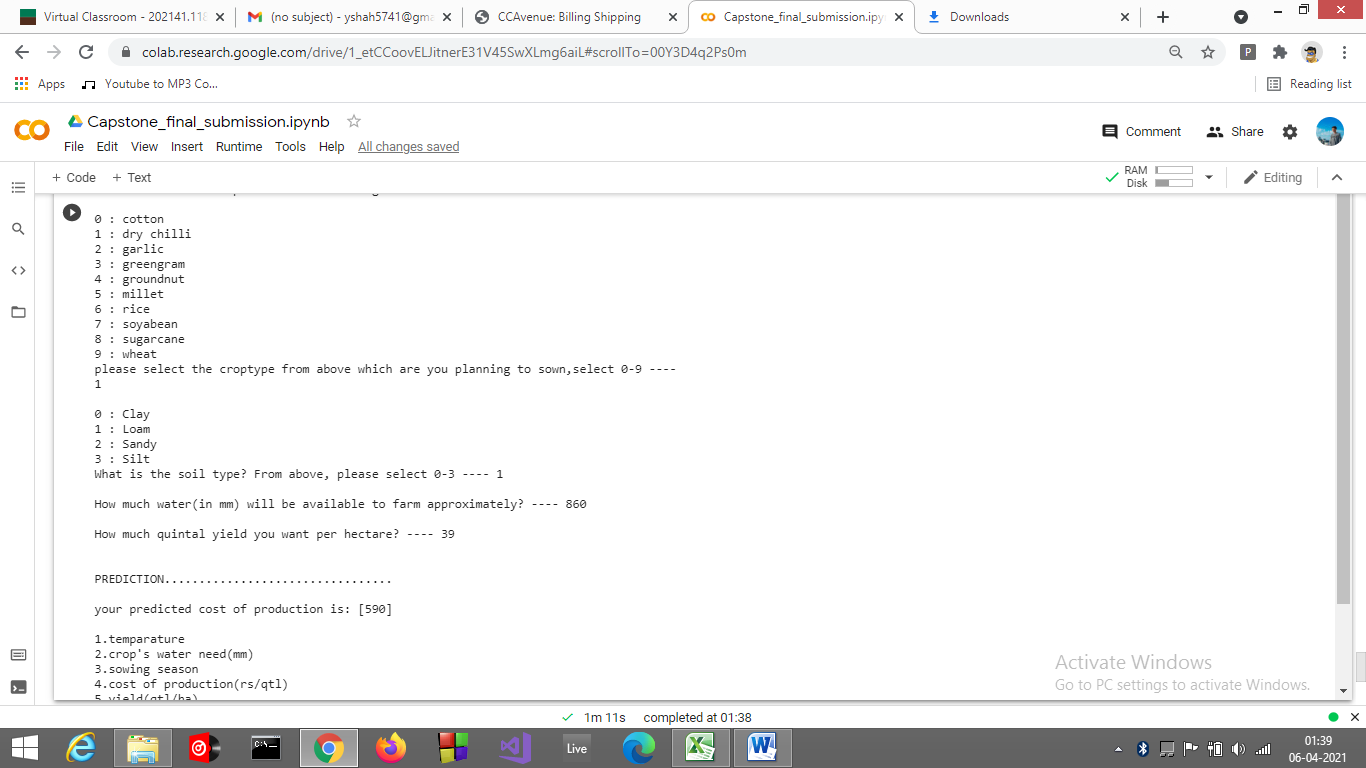
3. What will be best sowing season for this farmer?



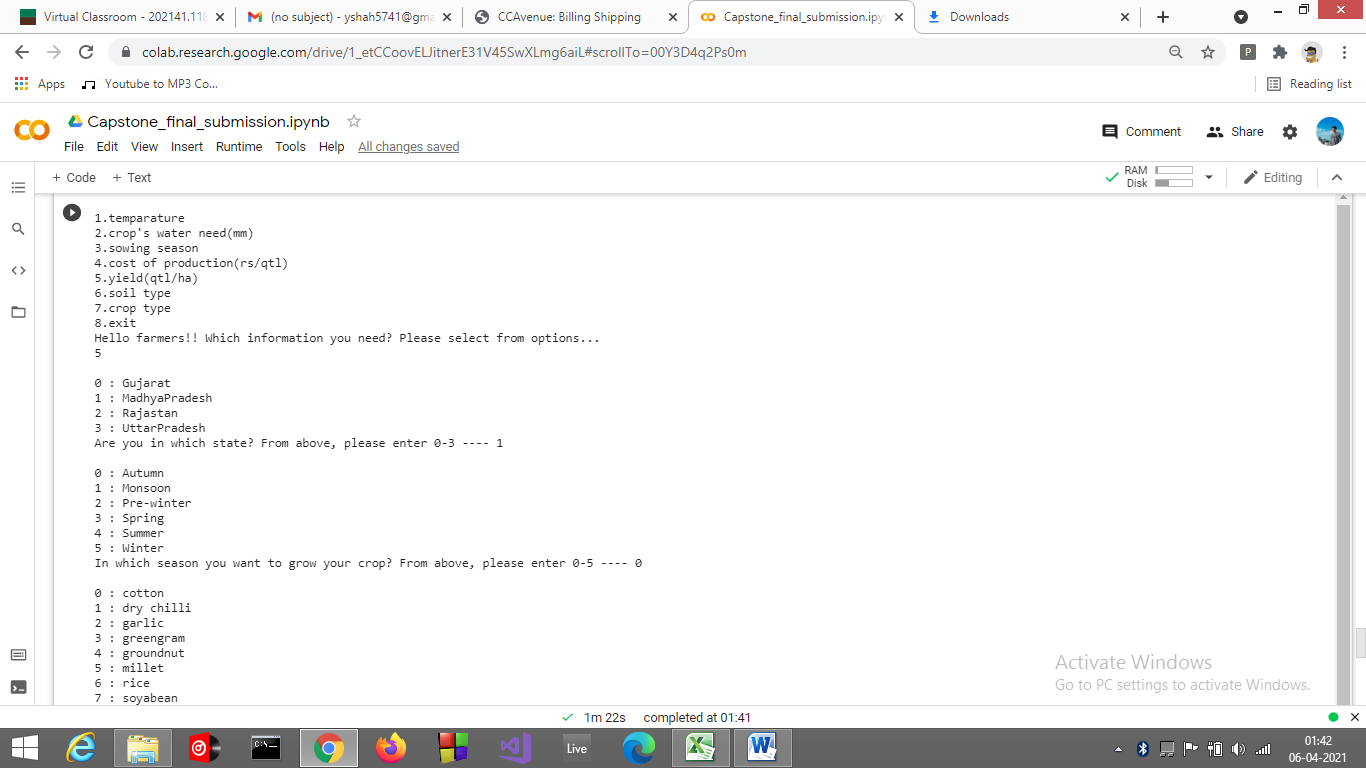


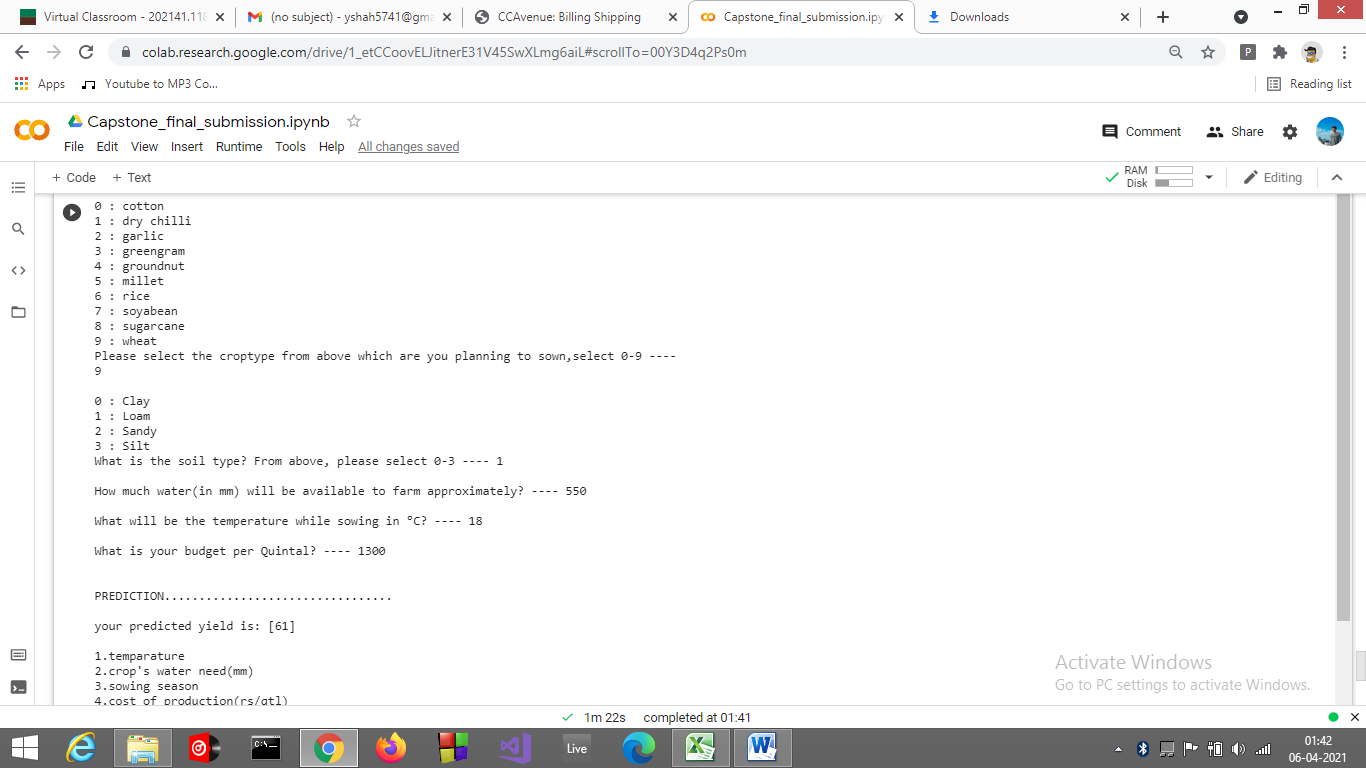
4. How much will be the cost of production for this farmer?



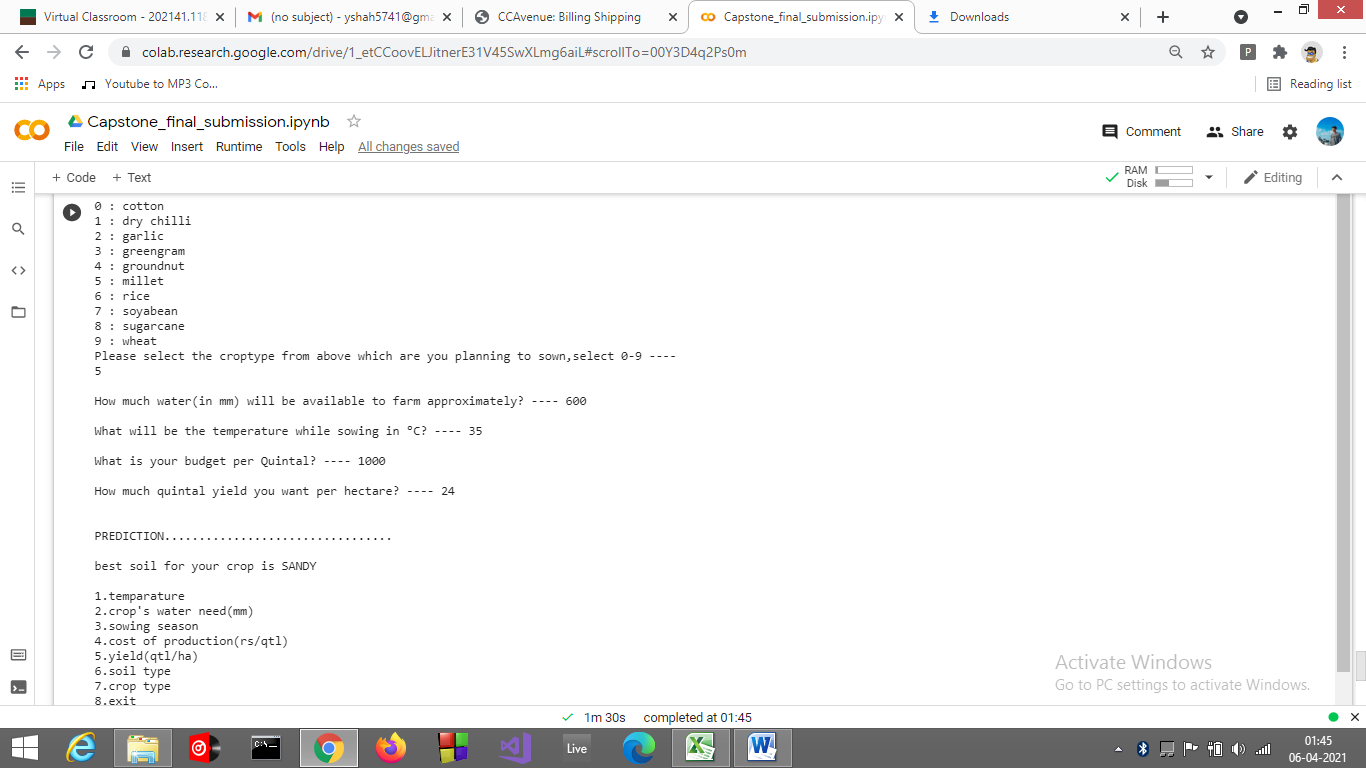
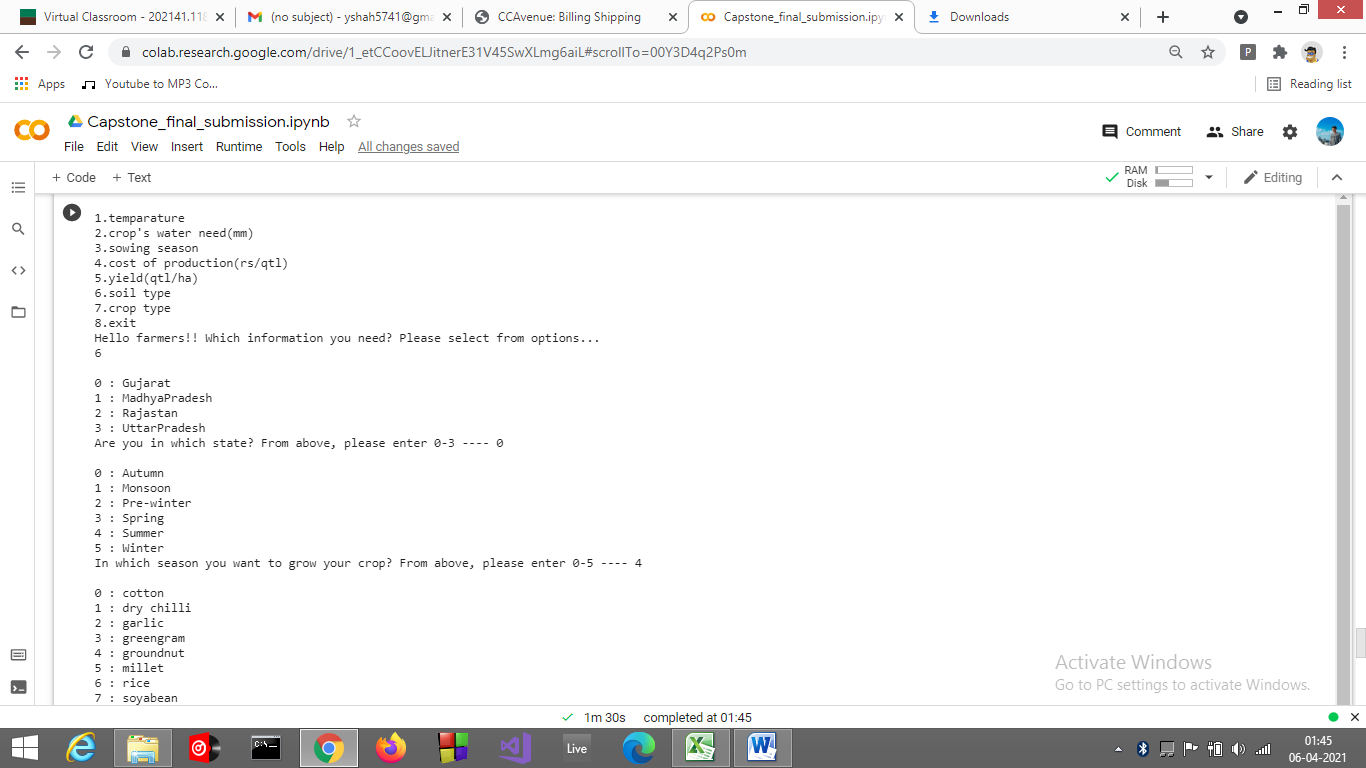


5. How much will be the yield for this farmer?





6. Which soil type will be best for this farmer?



7. Which crop should this farmer grow?

