Predictops - Crop Prediction System
Machine Learning (CSE523) Project

Aanshi Patwari
AU1841004
Ahmedabad University
Ahmedabad, India

Miracle Rindani
AU1841017
Ahmedabad University
Ahmedabad, India

Bhumiti Gohel
AU1841051
Ahmedabad University
Ahmedabad, India

Dipika Pawar
AU1841052
Ahmedabad University
Ahmedabad, India

Abstract—**Predicting the crop according to soil and weather is a major task for the farmers in the field of agriculture. It is based on the idea of classification problem from a Machine Learning point of view. To predict which crop can be grown under a particular set of conditions, we have implemented several classification algorithms.The crop prediction system based on the input parameters like soil type and weather conditions can predict the crops that can be grown which can be helpful to the farmers.**

Index Terms—**Classification, Crop Prediction, Support Vector Machine, Hyperparameter Tuning**

I. Introduction

Agriculture plays an important role in the development of our country. Improving the quality and the quantity of the crop production is a significant task in order to meet the food requirements of millions of people in our country. Doing farming or production of crops depends upon the properties of the soil as well as the conditions of the environment. Figuring out which crop can be grown on which land based on its properties can turn out to be useful to the farmers and the people who maintain the storage business of the crops harvested. So, it becomes important to build a system that can help in prediction of the suitable or appropriate crops that can be grown on an arbitrary land based on its soil parameters and the weather conditions.

II. Literature Survey

A. *Crop recommendation system for precision agriculture*

Precision Agriculture principle is applied on small, open farms[1] at individual farmer and crop level. Using data preprocessing techniques(Sammon's mapping method), high dimensional agricultural data is reduced to small size data to acquire the useful parameters. Crop Selection Method(CSM) helps to solve problem of crop selection as well as improve the net yield rate of the crop. The method provides output in the form of crop recommendations which can be grown over various seasons considering the factors of weather, soil type, water density, crop type. The predicted value of influential parameters determines the accuracy of CSM. K–Nearest Neighbor, random tree , CHAID(Chi-squared Automatic Interaction Detection), Naïve Bayes algorithm are used to improve the accuracy of the prediction to 80 to 88. The dataset used contains 5 environmental variables, 3 biotic variables and 2 area related variables to determine the crop yield in different districts.

B. *The Design of Hybrid Crop Recommendation System using Machine Learning Algorithms*

The dataset used is dependent on the soil quality[2] which is determined based on soil's NPK values, soil-PH value, crop disease and pesticides, seasonal parameters such as Kharif, rabbi, and summer crops. Data is collected from National Oceanic and Atmospheric Administration (NOAA) which contains climate and weather-related parameters and the Indian government agricultural portal www.data.gov.in. Algorithms like The neural network, SVM, Naive Bayes, KNN are implemented.

C. *Crop Prediction Using Machine Learning*

Dataset containing 1850 number of records and 5 number of features[3] is used. It contains soil parameters as the main attributes like Nitrogen, Phosphorus, Potassium and pH values of soil. KNN algorithm is used to train the model with a value of k=10 (best value) and accuracy of 85%. Also KNN with Cross Validation, Decision Tree, Naive Bayes and SVM are used for the comparison of performance in which kNN with Cross Validation performed the best with 88% accuracy.

D. *Crop Prediction System using Machine Learning Algorithms*

The dataset consisting of soil, weather and production parameters[4] is used to predict the best suitable crop. The preprocessing steps are well defined. Some methods to find most correlated features for the target column like Gini index, Entropy and Information Gain are discussed. KNN algorithm is used to predict the rainfall and weather report. Decision tree, KNN classifier and Naive Bayesian algorithms are used for crop prediction out of which the Decision tree gives poor results and the Naive Bayes gives the best results. The combination of both the classifiers also gives the good result.

E. *Evolutionary Tuning of SVM parameters*

The selection of the parameters in the SVM algorithm[5] is considered to be an important task. The choice of the kernel depends on the given type of the classification problem. The value of the hyperparameters determined by the grid search algorithm are much more reliable and optimum as compared to the values obtained through the gradient descent methods as they need the kernel and score function to be differentiable. Use of covariant matrix adaptation evolution strategy along with grid-search algorithm proved to provide the best results for the hyperparameter values as it was able to work for any type of kernel and handle all the other parameters as well.

III. Implementation

A. *Dataset*

The dataset used for the project mainly consists of features related to weather conditions and soil nutrient and pH levels at a particular place. The dataset contains 22 categories of crop labels on which we have trained the model.

B. *Preprocessing*

For the preprocessing part, we have calculated the correlation between the feature values with the crop label and with each other to make sure that the columns were i.i.d(independent and identically distributed) in nature. The major steps involved in preprocessing are:

1) Removing columns with high correlation values.
2) Deleting all the rows with one or more null values.
3) Equalising the number of data points for each crop label so that the final dataset we get is not biased towards any crop.
4) Extracting all the independent variables (soil and weather data) in a separate matrix and the labels (crop labels) in an array.
5) Dividing the dataset into train (70%) and test (30%) sets.

C. *Algorithms*

Based on the literature review, the performance of SVM algorithm was better compared to other algorithms.
The algorithms implemented on the dataset for the comparison of the results are:

- Support Vector Machine
- Naive Bayes
- K Nearest Neighbours

The explanation of the algorithms are as follows:

*1) **Support Vector Machine**:* Support Vector Machine or popularly abbreviated as SVM is highly preferred as it provides significant accurate results with less number of computations. The main objective is to determine a hyperplane in an N-dimensional space which can classify the datapoints into a specific label. This hyperplane is determined where in there is maximum distance between the datapoints i.e. maximum margin. The support vector points are considered helpful in maximising the margin.
Here in we have considered the bias term to be in association with the weight vector as follows:

$$f(x_i) = \boldsymbol{w}.x_i = \tilde{w}.\tilde{x}_i + w_0 \tag{1}$$

where $\boldsymbol{w}$=($\tilde{w}$,w_0) and x_i = ($\tilde{x_i}$,1)
The mathematical formulas included for the calculations of the algorithm are:

- Polynomial kernel function: The kernel function is also known as the score function which is used for better representation of the higher dimensional mapping of the datapoints.This can help in the separation and the formation of the hyperplane.

$$\phi(x) = k(x_i, x_j) = (1 + x_i * x_j)^d \tag{2}$$

where x_i = ith datapoint and x_j = jth datapoint in the dataset

- Loss computation formula: The formula mainly known as the hinge loss function, it helps in maximising the margin

$$J(\boldsymbol{w}) = \frac{1}{2}\|\boldsymbol{w}\|^2 + C\left[\frac{1}{N}\sum_{i}^{n} max(0, 1 - y_i * (\boldsymbol{w}.x_i + b))\right] \tag{3}$$

where J($\boldsymbol{w}$) = loss
$\boldsymbol{w}$ = weights
C = regularisation parameter
y_i = class label of ith example
x_i = ith datapoint

- Gradient equation:

$$\nabla_w J(\boldsymbol{w}) = \frac{1}{N}\sum_{i}^{n}\begin{cases}\boldsymbol{w}, & \text{if max(0, 1-}y_i * (\boldsymbol{w}.x_i) \text{ = 0} \\ \boldsymbol{w} - Cy_i x_i, & \text{otherwise}\end{cases} \tag{4}$$

where $\nabla_w J(\boldsymbol{w})$ = gradient value

- Weights updation:

$$\bar{\boldsymbol{w}} = \boldsymbol{w} - \nabla_w J(\boldsymbol{w}) \tag{5}$$

where $\bar{\boldsymbol{w}}$ = updated weights

The algorithm of Support Vector Machine can be defined as follows:

Algorithm 1 SVM algorithm
Input: X(features),y(labels),iterations
Output: W(weights)

```
Perform non-linear transformation($\phi(X)/k(x_i, x_j)$) kernel function
Initialise the weights(W) with the same dimension of $\phi(X)$
Loop i = 0 to number of iterations
    Loop i = 0 to number of samples(n)
        Computing the loss using the loss function
        Calculating the gradients using gradient equation
        Updating the weights
    end Loop
end Loop
return W
```

*2) **Naive Bayes**:* Naiv Bayes is one of the simplest and the easiest algorithm to implement on which can provide fast and accurate results. The fundamental assumptions made for the implementation of the algorithm are:

- Each feature is independent.
- Each feature has equal amount of contribution.
- There is no zero-frequency class label present.

The algorithm is based on the concept of Bayes' Theorem which is used for calculating the conditional probabilities.

$$P(A|B) = \frac{P(B|A).P(A)}{P(B)} \tag{6}$$

P(A) = (Prior probability)Represents the set of the class variable(consisting of all the classes for prediction)
P(B) = (Marginalisation)Represents the features of the dataset
P(B|A) = (Likelihood)Represents how much are the features related to a given crop
P(A|B) = (Posterior probability)Represents that given the features how likely the crop predicted will be correct
But the assumptions of the algorithm are not possible in real world situations due to which it is not generally prefer.

*3) **K-Nearest Neighbours:*** K-Nearest Neighbour or more formerly known as KNN is based on the concept of predict the class label by finding the nearest neighbour class. The steps included in the algorithm are:

- Calculate the euclidean distance for a given example with other n data points.
- Arrange the calculated n distance values in increasing order.
- Select value of 'k' for the algorithm.
- Choose k points from the n distance values
- Assign a label to the group of these k points

If the value of k is small then it leads to overfitting. If the value is too large then it leads to expensive computations. So, while applying the algorithm it is iterated over range of values which is not preferred as it leads to large number of computations.

IV. Results

A. *Algorithm Results*

The results of the indivdual algorithms implemented on the dataset are:

*1) **Support Vector Machine:*** After the implementation of the algorithm onto the dataset the performance results were like:

- Training accuracy: 94.74%
- Testing accuracy: 92.87%

This suggested that performing the step of hyperparameter tuning for the improvisation of the algorithm results.
Hyperparameters are the main charatertistics within any model which is external to the model and it varies according to the learning process performed by the model.
Grid-search is a mechanism which is used to find the optimal value of these hyperparameters of the SVM model which can help in predicting accurate results.
The hyperparameters involved in the SVM model are:

- Kernel function: Role of kernel is to take in the input data and tranform it into the required non-linear separable form. For this, different kinds of kernels such as linear, non-linear, polynomial, radial basis function(RBF) and sigmoid are available. These kernels perform the inner product between two different datapoints in a suitable feature space. This helps in computing the similarity among the points at a low cost in a higher dimensional vector spaces.
- Regularisation parameter(λ): This parameter is used to control the amount of error in the prediction which is mostly used for polynomial or linear type of kernel. As lambda becomes larger less wrongly classified examples are allowed. When lambda becomes smaller, more miss-classifications are allowed.
- Degree of polynomial: The degree of the polynomial is used in polynomial kernels only; it helps in controlling the flexibility of the decision boundary. More the degree of the polynomial more the flexibility. It helps the model to become more generalised.
- Tolerance value: It is the value which helps in limiting the number of iterations, keep a check on the convergence of the parameters and other metrics values such as gradient etc. The smaller the value, the faster the convergence which can degrade the performance, so larger value is preferred.

Conducting the grid-search technique on SVM algorithm suggested the values: Kernel function = Polynomial
Regularisation parameter(λ) = 10
Degree of polynomial = 2
Tolerance value = 0.1
The training loss of the algorithm tells that the loss decreases as the number of iterations increases is as follows: This suggests that other than optimisation of these values, the changes in the number of iterations can help in improvisation of the model.

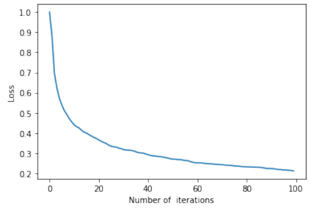


Fig. 1: Training loss SVM

The following figure illustrates the optimal number of iterations: The figure suggests that the value between 180-200 provides a greater amount of accurate results as compared to other results.

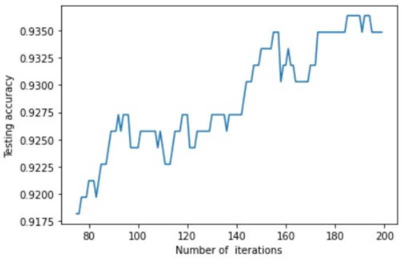


Fig. 2: Optimal number of iterations

After the implementation of hyperparameter tuning, the algorithm provided improved results like:

- Training accuracy: 96.71%
- Testing accuracy: 93.93%

*2) **Naive Bayes**:* As the algorithm is based on the concept of producing conditional probabilistic results by applying the Bayes Theorem. As from the dataset the problem of zero-frequency is removed, it gives the best accuracy results on the dataset.

*3) **K-Nearest Neighbour**:* As the algorithm tries to find k nearest data points to the test data given to the classifier. Here,the number of neighbours are checked by training with the values of k ranging from 1 to 50. For k = 8 or 10, n value(number of classes) = 22, and leaf size = 30; the algorithm gives best result.It is as follows:

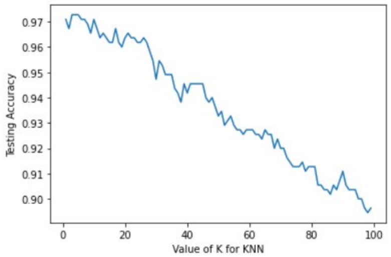


Fig. 3: Optimal value of k

B. *Comparison Results*

The following section shows the accuracy results of various algorithms employed in by us: Here, we can see that

Algorithm	Train Accuracy	Test Accuracy
SVM (scratch)	96.66%	94.18%
SVM (in-built)	99.71%	98.36%
Naive Bayes	99.61%	99.09%
KNN	95.15%	93.63%

KNN gives the lowest accuracy since it is a lazy algorithm. From the given algorithms, we chose to employ SVM from scratch mainly because it gave a comparable accuracy to Naive Bayes and it handles the interactions between the features in a better way as compared to Naive Bayes classifier.

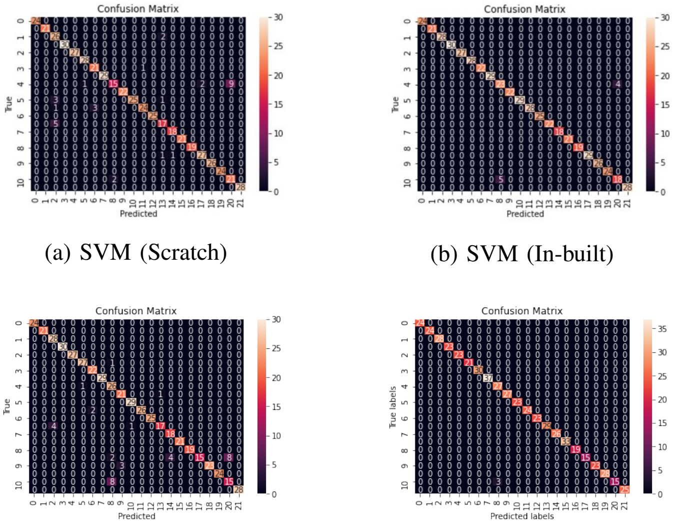
The following table shows the space and time complexities of these algorithms.

Algorithm	Space Complexity	Time Complexity
SVM	*O(n*d)*	*O(n*d)*
Naive Bayes	*O(d*m)*	*O(n*d*m)*
KNN	*O(n*d)*	*O(n*k*d)*

where,
n = number of training examples,
d = number of features,
m = number of classes, and
k = number of nearest neighbours.

The following are the confusion matrices generated by the algorithms employed.

The confusion matrices shows that:



(a) SVM (Scratch)

(b) SVM (In-built)

(c) KNN

(d) Naive Bayes

Fig. 4: Confusion matrices results

- The KNN matrix has numerous wrong predictions due to which it gives the lowest accuracy and so it is also known as lazy algorithm.
- There is almost no wrong prediction in the Naive Bayes results.
- The matrix results for inbuilt and from scratch algorithm for SVM are alomst similar.
- There are a few wrong predictions in the results of the SVM algorithm which suggests us to improve it a bit to achieve higher accuracy rates.
- For one particular class label all the algorithms predicted incorrectly which suggests that the example values for that particular label might be erroneous.

As an end product of the proposed system, a website is also deployed for the purpose of ease in access of the system and display of the results.
(Link:Predictops-Crop Prediction System)

V. Conclusion

The improved SVM algorithm gives almost equivalent results like that of Naive Bayes algorithm. So, the proposed project results suggests the use of any of the two algorithms for crop prediction based on the soil and the weather conditions which can help to determine which crop can be grown on an arbitrary land.

References

[1] R. H. R. C. K. T. K. S. Pudumalar, E. Ramanujam and J. Nisha, “Crop recommendation system for precision agriculture.”

[2] B. Viviliya and V. Vaidhehi, “The design of hybrid crop recommendation system using machine learning algorithms.”

[3] M. M. S. L. V. Kevin Tom Thomas, Varsha S and E. J. Thomas, “Crop prediction using machine learning.”

[4] V. Pavan and Shrikant, “Crop prediction system using machine learning algorithms.”

[5] F. Friedrichs and C. Igel, “Evolutionary tuning of multiple svm parameters.”