**R. C. Patel Educational Trust's**

**Institute of Management Research and Development, Shirpur**

**"Multi-Algorithm Comparative Framework for Crop Recommendation Using Classical Machine Learning Models"**

**Authors:-** Patil Prem, Patil Harshal

**Abstract:**

And machine learning is starting to become a valuable analysis tool for agriculture, particularly when trying to figure out what crops to plant in different fields based on the conditions of the soil and airspace. In this work, we compared six widely employed machine learning algorithms: Logistic Regression, Naive Bayes, Support Vector Machines (SVM), K-Nearest Neighbors (KNN), Decision Tree and Random Forest with a crop recommendation dataset. We tuned a combination of settings (e.g., the types of kernels, values of k in the k-nearest neighbor methods, and splitting rules) for each model to optimize performance. Our results demonstrate that Random Forest was the most accurate if everything else is assumed to be the same, but surprisingly it did not perform much better when compared with Naive Bayes (the default implementation in scikit-learn). Logistic Regression and Decision Trees were both straightforward to explain, SVM and KNN excelled at capturing intricate non-linear behaviours of the data. In conclusion, this research demonstrated the tradeoff regarding accuracy, speed, and explainability when machine learning is used in farming. It also paves the way for future studies with more advanced technologies such as hybrid models and deep learning to develop intelligent crop advisory systems.

**Introduction:**

Agriculture is the primary lifesaving activity, multiple challenges limit its continuation. They include climate change, soil degradation, the large population requiring food, and market unpredictability. Historically, crop selection was done selectively based on the farmer’s experience, historical patterns, and the trial-and-error approach. Although these methods work, there are inadequate to support the farmer’s crop selection in the prevailing dynamic and multifaceted environmental and soil factor scenarios. As such, data-based solutions are essential in assisting farmers to make appropriately informed and reliable crop selections.

Machine learning, or ML, has emerged in recent years as a powerful technology to develop a more intelligent and adaptive agricultural sector. ML algorithms can analyze large datasets encompassing crop performance, soil characteristics, and climatic data to expose patterns that are often beyond the experience of human observation. Based on these results, machine learning models can advise the best crops for a location which can help improve yield, minimize waste of resources, and enhance food security.

In this paper, six classical machine learning algorithms are used on a crop recommendation dataset to compare performances: Logistic Regression, Naive Bayes, Support Vector Machines (SVM), K-Nearest Neighbors (KNN), Decision Tree, and Random Forest. Each algorithm has its own trade offs: Logistic Regression and Decision Trees contain interpretability, Naive Bayes will be the simplest, SVM and KNN can process complex data, while Random Forest is often more accurate. This work will assess the performance of these algorithms under different parameterization, providing analysis of the trade offs involved between accuracy vs efficiency vs interpretation. In addition, these results can help develop decision support systems in support of furthering precision agriculture and sustainable farming.

**Objectives:**

The key aim of this study is to analyze and evaluate six classical machine learning algorithms (Logistic Regression, Naive Bayes, Support Vector Machines (SVM), K-Nearest Neighbors (KNN), Decision Tree, and Random Forest), and compare the algorithms for crop recommendation. Specifically, the study aims to:

1. Assess the accuracy, efficiency, and interpretability of each algorithm.
2. Optimize the model's predictive ability by tuning the hyperparameters.
3. Evaluate the trade-offs between lightweight models and ensemble methods.
4. Provide insights into the development of feasible tech-based agricultural crop advisory systems.
5. Easy to identify which crop is suitable for the land.

**Challenges:**

1. Data Quality and Availability – Agricultural datasets are frequently impacted by missing values, noise, and class imbalances that may decrease the reliability of models.
2. Hyperparameter Tuning – Every algorithm has its own set of parameters that need to be purposefully selected (for example, the value of C in Logistic Regression, the value of k in KNN, the form of the kernel in SVM). Not selecting these with precision may result in underfitting or overfitting
3. Model Generalization – Achieving the ability for models trained on one dataset to generalize to other mechanisms in various soil, climate, and regional conditions remains challenging.
4. Interpretability vs. Accuracy Trade-off – High-accuracy models such as Random Forests are less interpretable than Decision Trees or logistic regression methods and make task adoption with farmers and policymakers complex and challenging.
5. Computational Complexity – Certain algorithms (e.g., SVM with RBF kernel or large KNN searches) can be computationally heavy, preventing real-time use in a resource-limited environment.
6. Integration with Real-world Farming – For a model prediction to successfully transition into useful, actionable advice for farmers, it requires domain knowledge, localization, and a user-friendly interface.
7. Environmental Variability – The dynamic environment of factors like changes in weather, soil degradation, and pest infestation are not always model or captured in static datasets, which reduce prediction performance.

**Methodology:**

***Table 1: KNN PREDICTION TABLE***

**KNN:** (K-Nearest Neighbors) is a supervised learning algorithm that classifies data based on the majority class of its nearest K neighbors using distance measures.

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **No of features** | **K Value** | **Train%** | **Test%** | **Accuracy%** |
| 22 | 3 | 0.8 | 0.2 | 85.00 |
| 22 | 3 | 0.6 | 0.4 | 87.25 |
| 22 | 5 | 0.6 | 0.4 | 86.67 |
| 22 | 5 | 0.8 | 0.2 | 86.88 |
| 22 | 7 | 0.8 | 0.2 | 85.87 |
| 22 | 7 | 0.6 | 0.4 | 84.86 |

This table showing the model accuracy for different settings of K value, train-test split, and number of features,

Accuracy ranges 85.00% to 87.25%.

Best accuracy: 87.25% (K=3, Train/Test=0.6/0.4, 8 features).

***Table 2: SVC PREDICTION TABLE***

**SVM:** Support Vector Machine (SVM) is a supervised learning algorithm that finds the best boundary (hyperplane) to separate classes by maximizing the margin between data points.

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **No of features** | **Kernel Value** | **Train %** | **Test %** | **Accuracy%** |
| 8 | linear | 0.8 | 0.2 | 88.00 |
| 8 | linear | 0.7 | 0.3 | 89.00 |
| 8 | linear | 0.6 | 0.4 | 88.75 |
| 8 | rbf | 0.8 | 0.2 | 90.00 |
| 8 | rbf | 0.7 | 0.3 | 88.67 |
| 8 | rbf | 0.6 | 0.4 | 93.00 |
| 8 | poly | 0.8 | 0.2 | 89.00 |
| 8 | poly | 0.7 | 0.3 | 88.00 |
| 8 | poly | 0.6 | 0.4 | 92.00 |

This table shows the Support Vector Machine (SVM) model performance with different kernel types (linear, polynomial, RBF), train-test splits, and 8 selected features.

Accuracy ranges 88.00% to 93.00%

Best accuracy: 93.00% (RBF kernel, train/test=0.6/0.4, 8 features).

***Table 3: NAIVE BAYES PREDICTION TABLE***

**Naive Bayes:** Naive Bayes is a classification algorithm based on Bayes' theorem that assumes feature independence and predicts the class with the highest probability.

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **No of features** | **NB** | **Alpha** | **Binarize** | **Accuracy%** |
| 8 | Gaussian NB | - | - | 71.00 |
| 8 | Bernoulli NB | 1.0 | 0.5 | 69.00 |
| 8 | Complement Naive Bayes | 1.0 | - | 69.00 |

This table shows the Naïve Bayes model performance with three variants (Gaussian NB, Bernoulli NB, Complement NB) using 8 features.

Best accuracy: 71.00% (Gaussian NB)

Other model (Bernoulli NB, Complement) achieved about 69% accuracy.

***Table 4: DECISION TREE PREDICTION TABLE***

**Decision Tree:** Decision Tree is a machine learning algorithm that splits data into branches based on feature decisions to predict outcomes. It creates a tree where each node represents a condition and leaves represent final predictions.

|  |  |  |  |
| --- | --- | --- | --- |
| **No of features** | **Tune Parameters** | **max\_depth** | **Accuracy%** |
| 8 | criterion="gini" | None | 79.50 |
| 8 | criterion="entropy" | None | 78.50 |
| 8 | criterion="gini" | 5 | 84.00 |
| 8 | criterion="entropy" | 5 | 81.00 |
| 8 | min\_samples\_split=2 | None | 79.50 |
| 8 | min\_samples\_split=2 | 5 | 84.00 |

This table shows the Decision Tree model performance with different splitting criteria (gini, entropy) and max\_depth settings.

Accuracy range: 78.50% to 84.00%

Best accuracy 84.00% (criterion=gini, max\_depth=5,8 features)

**Result:**

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| ***Sr no*** | ***MOdel*** | ***Parameters*** | ***Train%*** | ***Test%*** | ***Accuracy%*** |
|  |  |  |  |  |  |
| 1 | KNN | K=3, Features=8 | 0.6 | 0.4 | 87.25% |
| 2 | SVM | RBF Kernel, Features=8 | 0.9 | 0.1 | 93.00% |
|  | Naïve Bayes | Gaussian NB | - | - | 71.00% |
| 4 | Decision tree | Criterion=gini, max\_depth=5 | - | - | 84% |

**Conclusion:**