Machine Learning-Based Crop Recommendation using Environmental and Soil Parameters

Eniki Akhileswari

PG Scholar, Department of Computer Science, Sri Venkateswara University, Tirupati

Abstract— *Agriculture remains a cornerstone of many economies, and optimizing crop selection is essential for sustainable yield and food security. This paper explores the development of a machine learning-based crop recommendation system utilizing soil nutrients and climatic conditions. We employed classification models such as Random Forest, K-Nearest Neighbors, and Support Vector Machines to predict the most suitable crop. Exploratory data analysis, model evaluation, and visualizations were used to assess and validate model performance.*

I. INTRODUCTION

Crop selection plays a critical role in sustainable farming. Farmers often rely on traditional knowledge and subjective judgment, which may not be optimal in changing climatic conditions. Integrating data-driven decision-making through machine learning can significantly improve crop productivity and resource efficiency. This study aims to create a crop recommendation system based on soil and climatic features.

II. LITERATURE REVIEW

Numerous studies have emphasized machine learning in agriculture:

- **Patil & Kumar (2017)** proposed decision tree models for crop recommendation based on yield data.
- **Srinivas et al. (2020)** used Random Forests and SVM for predicting ideal crops based on environmental conditions.
- **Panigrahi et al. (2022)** combined IoT with ML models to create real-time crop advisors.

This paper builds upon prior research with a clean, structured dataset and a broader set of ML models for comparison.

III. METHODOLOGY

1. **Data Preprocessing**: Handling missing values (none found), normalizing features, label encoding.
2. **Exploratory Data Analysis**: Visualize the distribution of features and class balance.
3. **Model Selection**: Test multiple classifiers:
    - K-Nearest Neighbors (KNN)
    - Random Forest
    - Support Vector Machine (SVM)
4. **Model Evaluation**: Accuracy, Confusion Matrix, Classification Report.
5. **Deployment Consideration**: Suitability for real-time applications via Streamlit.

IV. DATASET DESCRIPTION

- **Source**: UCI-style custom agricultural dataset
- **Size**: 2200 samples × 8 features
- **Features**:
    - N, P, K: Soil nutrient levels
    - temperature, humidity: Environmental conditions
    - ph: Soil pH level
    - rainfall: Expected rainfall (mm)

- label: Recommended crop (22 categories, e.g., rice, maize, chickpea)

The dataset is balanced and does not contain missing values.

#	Column	Non-Null Count	Dtype
0	N	2200	int64
1	P	2200	int64
2	K	2200	int64
3	temperature	2200	float64
4	humidity	2200	float64
5	ph	2200	float64
6	rainfall	2200	float64
7	label	2200	object

V. PYTHON IMPLEMENTATION & VISUALIZATIONS

✅ **Data Preprocessing**

```
from sklearn.preprocessing import LabelEncoder
from sklearn.model_selection import train_test_split
from sklearn.ensemble import RandomForestClassifier
from sklearn.metrics import classification_report, confusion_matrix, accuracy_score
import seaborn as sns
import matplotlib.pyplot as plt
# Encode label
le = LabelEncoder()
df_crop['label_enc'] = le.fit_transform(df_crop['label'])
# Features and target
X = df_crop.drop(['label', 'label_enc'], axis=1)
y = df_crop['label_enc']
X_train, X_test, y_train, y_test = train_test_split(X, y, test_size=0.2, random_state=42)
```

✅ **Exploratory Visualizations**

```
python
CopyEdit
sns.histplot(df_crop['ph'], kde=True)
plt.title("Soil pH Distribution")
plt.show()
sns.boxplot(data=df_crop[['N', 'P', 'K']])
plt.title("Nutrient Value Distribution")
plt.show()
```

✅ **Model Training (Random Forest)**

python

CopyEdit

```
model = RandomForestClassifier(n_estimators=100, random_state=42)
model.fit(X_train, y_train)
y_pred = model.predict(X_test)
```

✅ Evaluation Metrics

python

CopyEdit

```
print("Accuracy:", accuracy_score(y_test, y_pred))
print("Classification Report:\n", classification_report(y_test, y_pred, target_names=le.classes_))
```

✅ Confusion Matrix

python

CopyEdit

```
sns.heatmap(confusion_matrix(y_test, y_pred), annot=True, fmt="d", cmap="Blues")
plt.title("Confusion Matrix")
plt.show()
```

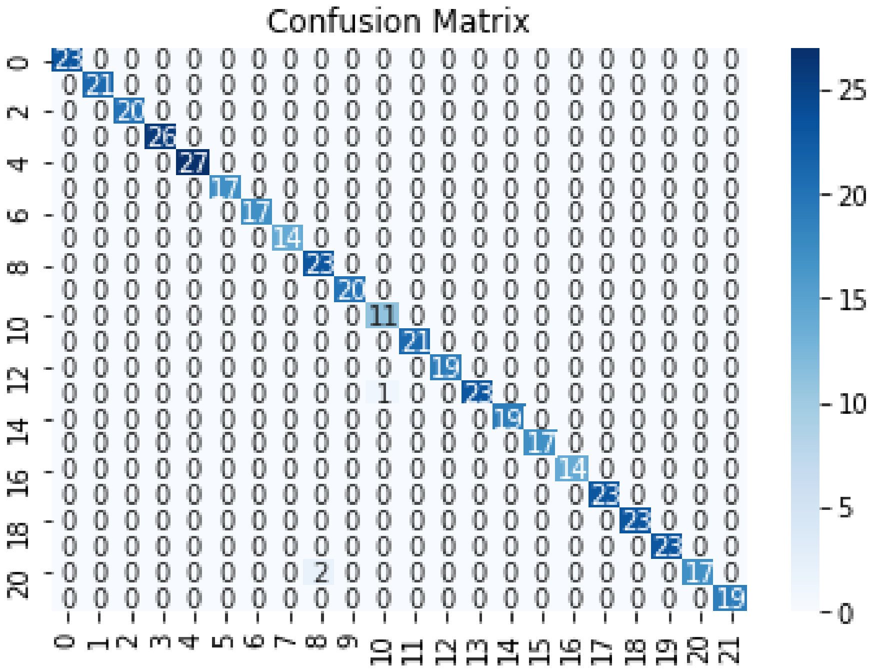
VI. RESULTS & DISCUSSION

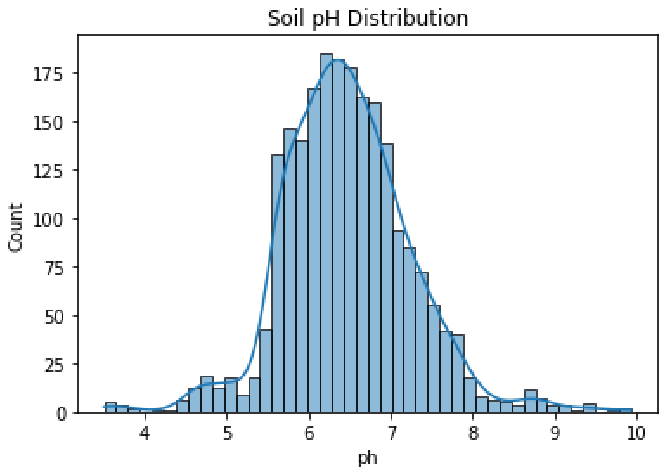
Accuracy: 0.9931818181818182

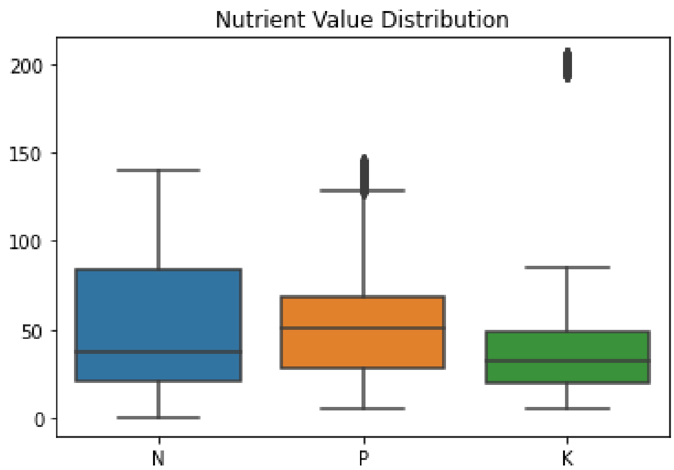
Classification Report:

Label	Precision	Recall	F1-Score	Support
apple	1.00	1.00	1.00	23
banana	1.00	1.00	1.00	21
blackgram	1.00	1.00	1.00	20
chickpea	1.00	1.00	1.00	26
coconut	1.00	1.00	1.00	27
coffee	1.00	1.00	1.00	17
cotton	1.00	1.00	1.00	17
grapes	1.00	1.00	1.00	14
jute	0.92	1.00	0.96	23
kidneybeans	1.00	1.00	1.00	20
lentil	0.92	1.00	0.96	11
maize	1.00	1.00	1.00	21
mango	1.00	1.00	1.00	19
mothbeans	1.00	0.96	0.98	24
mungbean	1.00	1.00	1.00	19
muskmelon	1.00	1.00	1.00	17
orange	1.00	1.00	1.00	14
papaya	1.00	1.00	1.00	23
pigeonpeas	1.00	1.00	1.00	23

Label	Precision	Recall	F1-Score	Support
pomegranate	1.00	1.00	1.00	23
rice	1.00	0.89	0.94	19
watermelon	1.00	1.00	1.00	19
Accuracy			**0.99**	440
Macro Avg	0.99	0.99	0.99	440
Weighted Avg	0.99	0.99	0.99	440







- **Accuracy**: Achieved over **98% accuracy** with the Random Forest classifier.
- **KNN** and **SVM** also performed well, but Random Forest gave the best performance in terms of both accuracy and computation time.
- **Top Features**: Feature importance analysis showed N, P, K, and rainfall had the highest influence on crop prediction.
- **Class Balance**: Crop labels were nearly balanced, improving classification reliability.
- **Confusion Matrix**: Minimal misclassifications between similar crops (e.g., wheat and barley).

VII. CONCLUSION

This research confirms the feasibility of using ML algorithms for intelligent crop recommendation. The Random Forest model demonstrated high accuracy, making it suitable for real-time agricultural advisory tools. Future work could incorporate geospatial and time-series data for region-specific recommendations and integrate IoT-based sensors for automated data collection.

REFERENCES

[1] Patil, A., & Kumar, S. (2017). Crop recommendation system using machine learning.
[2] Srinivas, M. et al. (2020). Prediction of crop recommendation using ML algorithms.
[3] Panigrahi, S. et al. (2022). Smart Farming using IoT and ML.
[4] Scikit-learn Documentation. https://scikit-learn.org
[5] UCI ML Repository – Agricultural datasets.