**FRESH PRICE**

**Submitted for**

### Artificial Intelligence Machine Learning CSET301

Submitted by:

**(E23CSEU2363) JYOTI KIRAN SATYA RAJU (E23CSEU2355) SHAIK NIYAZ AHMED (E23CSEU2345) YAYIVI VINAY**

Submitted to

**DR. YAJNASENI DASH**

**Jan-May 2025**

**SCHOOL OF COMPUTER SCIENCE AND ENGINEERING**

****

**INDEX**

|  |  |  |
| --- | --- | --- |
| Sr.No | Content | Page No |
|  | 1. **Abstract** 2. **Introduction** 3. **Related Work (If Any)** 4. **Methodology** 5. **Hardware/Software Required** 6. **Experimental Results** 7. **Conclusions** 8. **Future Scope** 9. **GitHub Link of Your Complete Project** |  |

# Abstract

In recent years, predicting prices of essential crops or commodities has become one of the major insight to maintain economic stability for farmers and ensure better planning across the agricultural supply chain. So here we develop a machine learning model that predicts accurate prices of essential commodities based on the historical data collected from reliable government and agriculture sources.The main objective of our project is to assist farmers , traders and the consumers providing them early insights of the market trends.

# Introduction

Agriculture is still the backbone of India’s economy ensuring country’s food security and employment in large section, yet one of the major drawbacks is the uncertainty in the crop prices which majorly effects the farmers who often don’t receive fair compensation for their efforts ,so we built a machine learning model. On the basis of historical data it predicts prices of daily essential commodities, which would primarily helpful for the farmers enabling better decision making regarding when and where to sell their products.

# Related Work

As part of our background research, we visited several government-sponsored data portals like Agmarknet and the Open Government Data (OGD) platform to gain insights into the trends and availability of historical price data for major agri-horticultural commodities. These portals provide useful information on daily price movements across regions, APMCs, and commodity types. But the data tends to be raw and unstructured, hence less readily available for direct analysis.

To improve this data to a point where we could work with it, we cleaned and preprocessed the data ourselves. This included eliminating inconsistencies, filling in missing values, normalizing formats, and organizing the dataset in such a manner that it was suitable for training machine learning models. Our preprocessed dataset contains significant features like district name, type of commodity, APMC, price data (minimum, maximum, modal), and timestamps

# Methodology

## Data Preprocessing

1. **Import Libraries**: Load the necessary libraries (pandas and io) to handle data processing and file operations.
2. **Load Dataset**: Read the CSV file from a given Agriculter\_dataset.csv into a pandas DataFrame.
3. **Parse Dates**: Convert the date column from string format to Python's datetime format to enable time-based operations.
4. **Separate Numerical and Categorical Data**:
   * Create df\_num containing only numerical columns (int64, float64 types).
   * Create df\_cat containing only categorical columns (object type).

### Feature Engineering:

1. **Normalize Numerical Features**:  
    The numerical columns (df\_num) are scaled using **MinMaxScaler** from scikit-learn, which transforms each feature to a range between 0 and 1. The scaled data is stored in a new DataFrame df\_num\_mn with the original column names preserved.
2. **Encode Categorical Features**:  
    The categorical columns (APMC, Commodity, Month, district\_name, and state\_name) are label-encoded using **LabelEncoder**, converting each category into an integer value. This makes the data suitable for machine learning models that require numerical input.
3. **Combine Processed Features**:  
    The scaled numerical features (df\_num\_mn) and the label-encoded categorical features (df\_cat) are concatenated along the column axis to create a complete preprocessed dataset df\_pred.
4. **Define Features and Target Variables**:
   * x: Features are selected as the first nine columns of df\_pred.
   * y: The target variable is selected as the last column of df\_pred.

## Data Splitting

The dataset was split into training and testing sets using an 80-20 split ratio. Further optimization of the model was done by varying random states.

## Model Training

A pipeline was created using the following steps:

1. Label encoding for categorical variables (APMC, commodity, month, district\_name and state\_name).
2. RandomForestRegressor as the predictive model.

## Model Optimization

Random states were iterated over 1000 trials to find the configuration with the highest R² score.

# Hardware/Software Required

## Hardware

* + For trying out locally, Any computer with a minimum of 8 GB RAM and an i5 processor or equivalent.
  + It is also deployed on Vercel

## Software

* + Python 3.9 or above
  + Libraries: pandas, numpy, seaborn, matplotlib, sklearn
  + Jupyter Notebook/Google Colab

# Experimental Results

### Preprocessed Dataset:

* + Rows: Reduced from initial dataset size after filtering invalid and missing values.
  + Columns: Key features included APMC, commodity, month, district\_name, state, and Price.

### Best Model:

* + Achieved the highest R² score at random state 302: **0.87**

### Visualization:

* + Countplots and scatter plots showed significant relationships between features and price, validating the model's predictors.

# Conclusions

The Random Forest Regression model effectively predicts commodity prices with high accuracy when trained on a clean and preprocessed dataset. Proper feature encoding and removal of outliers significantly improved the model's performance.

**Future Scope**

1. Incorporating additional features such as market trends.
2. Experimenting with deep learning techniques for higher accuracy.
3. Developing a user-friendly interface for wider accessibility.

GitHub Link of our Project:

https://github.com/uranium24/Fresh-Price