**Agriculture Data analysis:**

Problem statement:

The dataset includes the following columns:

**Domain Code** Represents a code identifying the domain of the data, such as a specific sector, category.

**Domain**: A descriptive name for the domain associated with the Domain Code.

**Area Code (M49)** An identifier for geographic areas, likely based on the M49 standard (used by the United Nations).

**Area** The name of the geographic area corresponding to the Area Code (M49).

**Element Code** A numeric code representing specific data elements

**Element** A descriptive label for the data element associated with the Element Code.

**Item Code (CPC)** A code identifying specific items or products.

**Item** The name of the product or item

**Year Code** A code for the year the data pertains to

**Year** The calendar year of the data.

**Unit** The unit of measurement for the data in the Value column.

**Value** The numeric value or observation corresponding to the specified domain, area, element, item, and year.

**Flag** A code indicating the quality, status, or source of the data.

**Flag Description** A descriptive explanation of the Flag.

**Note** Additional comments or explanations for certain rows

The project aims to leverage historical data to provide predictive insights into agricultural values.

**Here's a step-by-step approach to achieve our analysis:**

1. **Data Preprocessing**:
   * Handle missing values appropriately.
   * Convert categorical variables using encoding techniques.
   * Normalize or scale the "Value" depending on the chosen model.
2. **Exploratory Data Analysis (EDA)**:
   * Analyze relationships between features and the target variable.
   * Visualize trends over time or across regions.
3. **Model Selection**:
   * Start with simpler models (Linear Regression, lasso , ridge ….) for baseline performance.
   * Graduate to more complex models (e.g., Random Forest, XGBoost)
4. **Model Evaluation**:
   * Split the dataset into training and test sets to make predictions using the test data breaking dataset to train and test to 75% train data and 25% test data
   * Use metrics such as R-squared, MSE to evaluate model performance.
5. **Hyperparameter Tuning**:
   * Implement techniques like grid search or random search to enhance model performance.
6. **Prediction**:
   * Once the best model is identified and tuned, make predictions based on the input features.
7. **Post-Analysis**:
   * Interpret results and identify practical implications or applications of the predictions.

**Exploratory Data Analysis Project (EDA)**

**Plot** : using seaborn's countplot() function to create a countplot for categorical variables

using seaborn's boxtplot() function to create a boxtplot for continuous variables

|  |  |  |  |
| --- | --- | --- | --- |
| **S. No** | **KPI** | **Visualization** | **Outcome** |
| 1 | Features vs Value | Boxplot | multiple boxplots for different variables: "Domain," "Element Code," "Element," and "Year." Boxplots to visualize the distribution of data and identify outliers. |
| 2 | Distribution of categorical column | Count plot | The bar plot illustrates the distribution of the Domain , Element Code , Element , Year , Unit , flag , flag description. |
| 3 | Plotting histogram for the continuous column | Histogram |  |
| 4 | plot the histograms and count distribution for the attributes | Histogram |  |
| 5 | Plotting the distribution of the fields | Countplot |  |
| 6 | Sunburst Chart of Agricultural Data | Sunburst | hierarchical data, which is particularly useful for displaying parts of a whole and understanding relationships between categories |
| 7 | Area vs Element by Value | Subplot | agricultural data related to different countries, with specific metrics like area harvested, production, and yield |
| 8 | Year vs Element by Value | Subplot |  |
| 9 | Production Value by Area | Box plot | Boxplot represent the value by area, india is the first producer follows by China and Indonesia |
| 10 | Total Production Value by Element Code | Bar plot | the element code 5510 is the biggest value in term of production |
| 11 | Distribution of Production Value by Element Code | Box plot | the element code 5510 is the biggest value in term of production |
| 12 | Distribution of Production Value by Element | Box plot | the production is the biggest value |
| 13 | Element vs Value | Bar plot | the production is the biggest value |
| 14 | Unit vs Value | Bar Plot | t unit is the biggest value |
| 15 | average price trends across regions | Bar Plot | represent the value by area, India is the first producer follows by China and Indonesia |
| 16 | Average Production Value by Element Over Time | Bar plot | 2022 has the biggest production so far comparing to the other years |
| 17 | Total Production Value by Year | Bar plot | 2021 has the biggest value in term of production over the years |
| 18 | Stacked Bar Chart of Production Value by Crop and Year | Stacked bar | 2021 is the highest in term of production |
| 19 | Pairplot to see relationships between numeric features | Pair plot | Pairplot to see relationships between numeric features |
| 20 | Heatmap of the correlation matrix | Heatmap | "Area Code" and "Element Code" have a very weak positive correlation.  Year" appears to have a slight negative correlation with "Value," indicating that as the year increases, the value may slightly decrease (or vice versa). |
| 21 | Value vs Element Code, Colored by Flag | Scatter Plot | The scatter points appear to cluster at specific values for each year, suggesting that the relationship between "Value" and "Year" might not be continuous but is influenced by the corresponding area. |
| 22 | Ploting the line chart to check the deviation of the value over the year | Line plot | the line illustrates a trend in data over the span of several years, from 2016 to 2022. the average value shows a slight increase over time, with some fluctuations along the way |
| 23 | Treemap of Values by Domain, Area, and Item | Tree map | The treemap is divided into several rectangular sections, each representing different domains, areas, and items. |
| 24 | Sunburst Chart of Domain and Area and item in Agriculture | Sunburst | The chart is a circular, multi-level diagram that visually represents hierarchical data. |
| 25 | Trend Analysis of Production Value Over Years | Trend plot | the production value has shown both growth and decline over the years.  The highest production value was in 2019.  After a dip in 2020, the production value has been steadily increasing. |

**Applying Machine Learning Models**

* Applied Logistic Regression model, Ridge & lasso , decision tree, random forest, Gradient boost regressor, Extreme gradient boost regressor (Xgboost)

**Comparing Model Accuracy and Tuning**

**Interpretation:**

1. **Default Parameters**

* **Decision Tree Regressor: High accuracy (0.9327) and R2 score with minimal overfitting.**
* **Random Forest Regressor: Highest accuracy (0.9910) and low overfitting.**
* **XGBoost Regressor: Good balance between train and test scores with an R2 score of 0.8019.**
* **Gradient Boost Regressor: Lower performance with accuracy around 0.3305.**
* **Linear, Lasso, and Ridge Regressions: Very low accuracy scores and significant MSE scores.**

1. **Hyperparameter Tuning**

* **Decision Tree Regressor: Maintained high accuracy.**
* **Random Forest Regressor: Optimal parameters improved overfitting issues.**
* **XGBoost Regressor: Reduced performance with parameter tuning.**
* **Gradient Boost Regressor: Improved performance with optimal parameters.**
* **Linear Regression: Not included.**
* **Lasso and Ridge Regressions: Minor improvements with tuning.**

**Best Model:**

the Random Forest Regressor appears to be the best model, especially when using default parameters. It shows the highest accuracy (0.9910) with low overfitting.

When hyperparameters are tuned, the Random Forest Regressor still performs very well, indicating its robustness and stability.

From MSE & R2 metrics, the Random Forest Regressor stands out for its low MSE and high R2 score, indicating both its accuracy and robustness in predictions. The Decision Tree Regressor also performs quite well, while models like Gradient Boost Regressor show significant improvement with hyperparameter tuning.