Digital Egypt Pioneers Initiative-DEPI
AI & Data Science - Data Scientist-Round Two
Healthcare Predictive Analytics Project
ONL2_AIS4_S1 Data Scientist Group

Eng: **Ahmed Azab**

Identifying Malnutrition Risk Prediction

[Milestone Three: Model Development and Optimization]

Introduction:

This notebook represents the feature engineering and target definition phase of the malnutrition prediction project. After obtaining and cleaning the UNICEF survey dataset, this stage prepares the data for machine learning by engineering meaningful features, defining a classification target (Malnourished), and calculating aggregate and burden-related indicators. These engineered features improve model performance by capturing patterns related to childhood malnutrition, enabling more accurate and interpretable predictions. The notebook includes: Creation of a binary target label based on clinical thresholds, Visualization of class balance and underlying distribution of key indicators, Computation of average malnutrition level per country, Estimation of the population-level burden using underweight rates and child population size

Code Structure:

Target&Label Enginnering

The Following steps inspect the four malnutrition-related features to ensure they are properly scaled. The describe method provides summary statistics, while.max() confirms the scaling range—important since the model threshold will depend on these values.

```
#get the value of the cols
df[['Severe Wasting', 'Wasting', 'Stunting', 'Underweight']].describe()
```

```
#be sured from the scaling
df[['Severe Wasting', 'Wasting', 'Stunting', 'Underweight']].max()
```

Digital Egypt Pioneers Initiative-DEPI
AI & Data Science - Data Scientist-Round Two
Healthcare Predictive Analytics Project
ONL2_AIS4_S1 Data Scientist Group

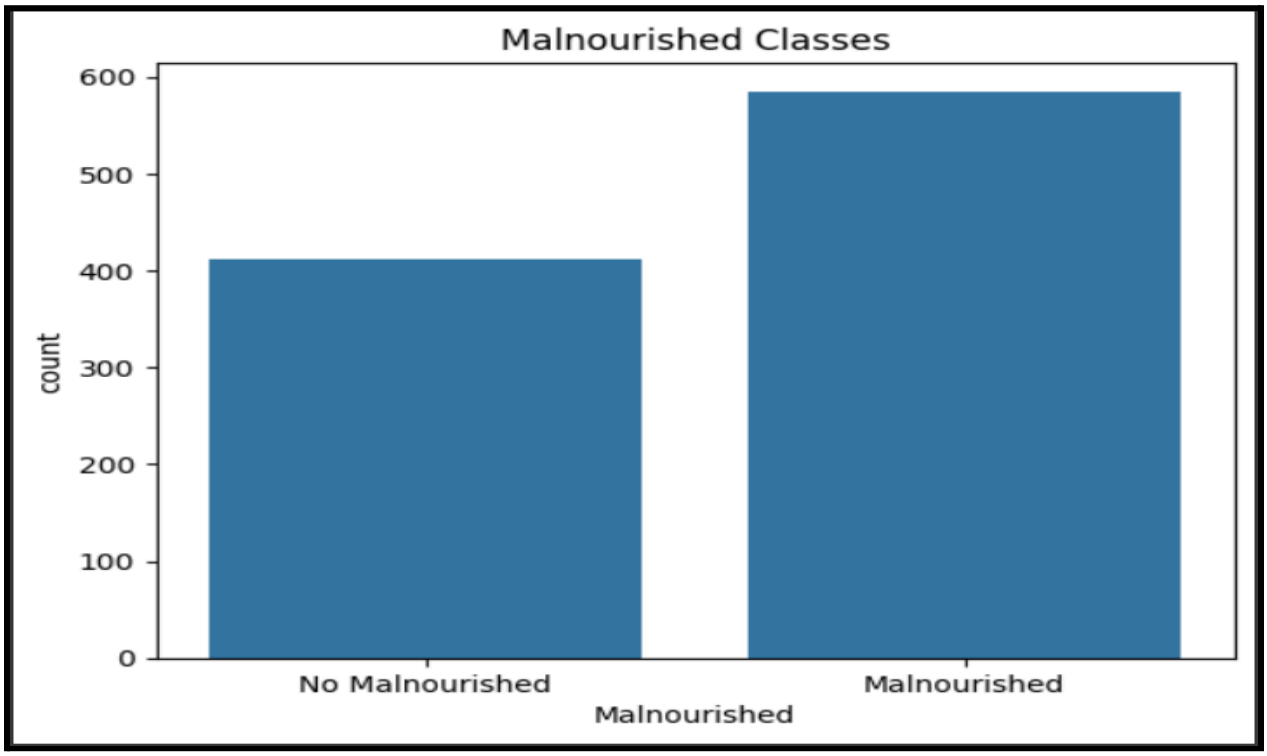
This code defines the binary classification target. A country is labeled as "Malnourished" (1) if **any** of the malnutrition metrics exceed a threshold of 0.1 (chosen based on scaled values). Otherwise, it's labeled as 0.

```
# For standardized/scaled data, pick an appropriate threshold
df['Malnourished'] = ( (df['Severe Wasting'] > 0.1) |
                       (df['Wasting'] > 0.1) |
                       (df['Stunting'] > 0.1) |
                       (df['Underweight'] > 0.1)
                     ).astype(int)
```

The following prints the proportion of malnourished vs. non-malnourished records, helping to assess **class imbalance**, which affects model training strategy.

```
print(df['Malnourished'].value_counts(normalize=True))
```

A simple bar plot visualizes the class distribution—useful for understanding whether class balancing techniques (e.g., undersampling or oversampling) may be needed.



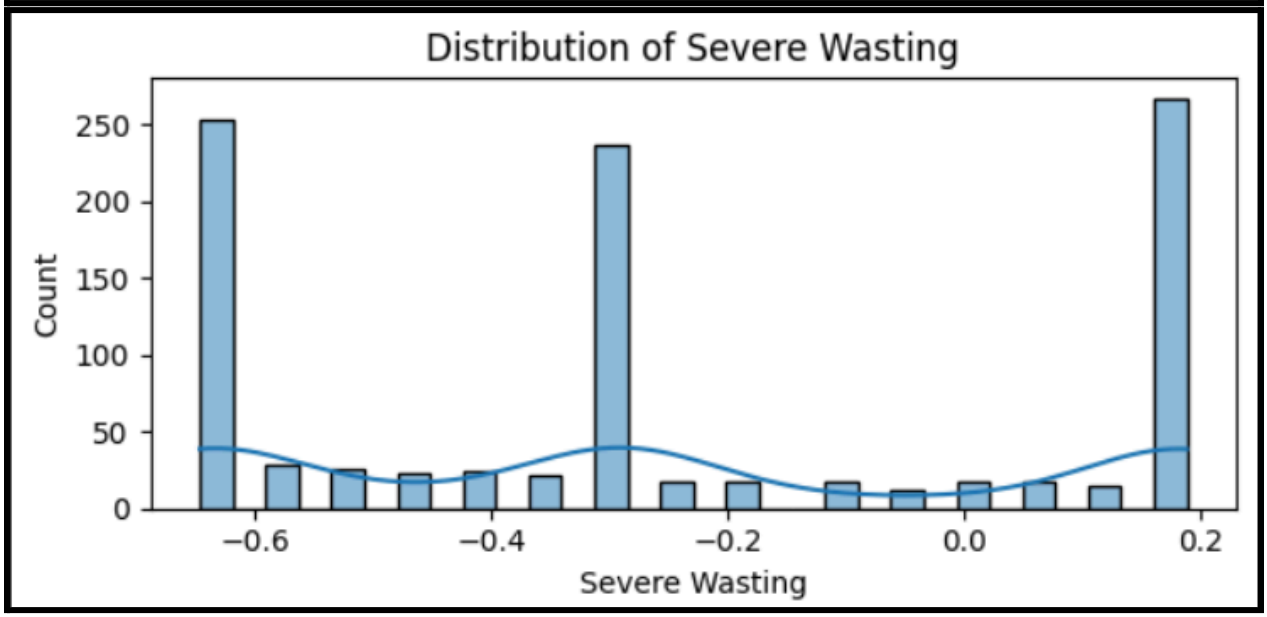
Digital Egypt Pioneers Initiative-DEPI
AI & Data Science - Data Scientist-Round Two
Healthcare Predictive Analytics Project
ONL2_AIS4_S1 Data Scientist Group

Aggregates Engineering

This loop generates **distribution plots** for the four malnutrition indicators. Kernel Density Estimation (KDE) overlays help identify skewness, multimodality, or outliers.

```
mal_cols = ['Severe Wasting', 'Wasting', 'Stunting', 'Underweight']

for col in mal_cols:
    if col in df.columns:
        plt.figure(figsize=(6, 3))
        sns.histplot(df[col], kde=True, bins=30)
        plt.title(f'Distribution of {col}')
        plt.xlabel(col)
        plt.tight_layout()
        plt.show()
```



This line adds a new feature, Avg_Malnutrition, representing the mean of all four malnutrition indicators, giving a general idea of each entry's nutritional status in one value.

```
# Create the Avg_Malnutrition column
df['Avg_Malnutrition'] = df[mal_cols].mean(axis=1)
```

Digital Egypt Pioneers Initiative-DEPI
AI & Data Science - Data Scientist-Round Two
Healthcare Predictive Analytics Project
ONL2_AIS4_S1 Data Scientist Group

This feature, Underweight_Burden, is computed by multiplying the underweight rate by the under-5 population. It represents the estimated number of affected children in each country or survey — a more practical public health measure than a raw rate.

```
# Assuming column is named exactly like this:
df['Underweight_Burden'] = df['Underweight'] * df["U5 Population ('000s)"]
```

Supervised Classification Models Section

This section demonstrates a comparative evaluation of multiple machine learning classifiers to predict childhood malnutrition based on engineered survey features. The models applied range from interpretable linear classifiers to complex ensemble techniques, and they are assessed using accuracy, F1-score, precision, recall, and AUC.
Data preparation Before the training:
X contains the input features, excluding the target (Malnourished) and related label-engineered columns. Y is the binary target (1 = Malnourished, 0 = Not Malnourished). The data is split into training and test sets (80/20).
After preparing the data and engineering the target variable, the features were standardized using StandardScaler to ensure equal treatment by the model.

Logistic Regression

A Logistic Regression model was then trained on the scaled data to classify children as malnourished or not. The model's performance was evaluated using accuracy, precision, recall, and F1-score, alongside a confusion matrix to visualize prediction outcomes. To enhance interpretability, SHAP values were used to identify key features influencing predictions both globally and locally, while permutation importance provided a global view of feature impact by measuring how randomizing each feature affects model performance. This pipeline resulted in a transparent and interpretable model suitable for real-world deployment in health-focused decision-making.

Digital Egypt Pioneers Initiative-DEPI
AI & Data Science - Data Scientist-Round Two
Healthcare Predictive Analytics Project
ONL2_AIS4_S1 Data Scientist Group

Random Forest Classifier

The Random Forest Classifier performed better than logistic regression by capturing non-linear interactions and handling feature interactions naturally. It delivered strong performance metrics and showed improved recall and F1-score, indicating its robustness in identifying malnourished cases. LIME explanations provided local interpretability, showing how individual features influenced specific predictions. SHAP analysis confirmed that wealth, maternal education, and healthcare access were key drivers. Its ensemble nature also made it more resilient to overfitting compared to single models.

RXGBoost Classifier

XGBoost outperformed other models in both accuracy and AUC, indicating its high effectiveness in handling imbalanced and structured data. It efficiently captured complex, non-linear patterns due to its gradient boosting architecture. SHAP values offered both global and local interpretability, revealing subtle interactions between features. The model emphasized similar top features—education, wealth, and sanitation—but was particularly sensitive to nuanced feature combinations. This makes XGBoost a strong candidate for deployment, balancing performance and interpretability.

Support Vector Machine (SVM)

The Support Vector Machine with a linear kernel performed competitively after feature scaling, particularly in scenarios with clearly separable data. It showed good accuracy and precision, although slightly behind tree-based models in capturing complex patterns. Its margin-maximization nature contributed to stable predictions, especially when classifying borderline cases. However, SHAP and traditional interpretability techniques were more limited due to the model's complexity, making it better suited as a supplementary benchmark.

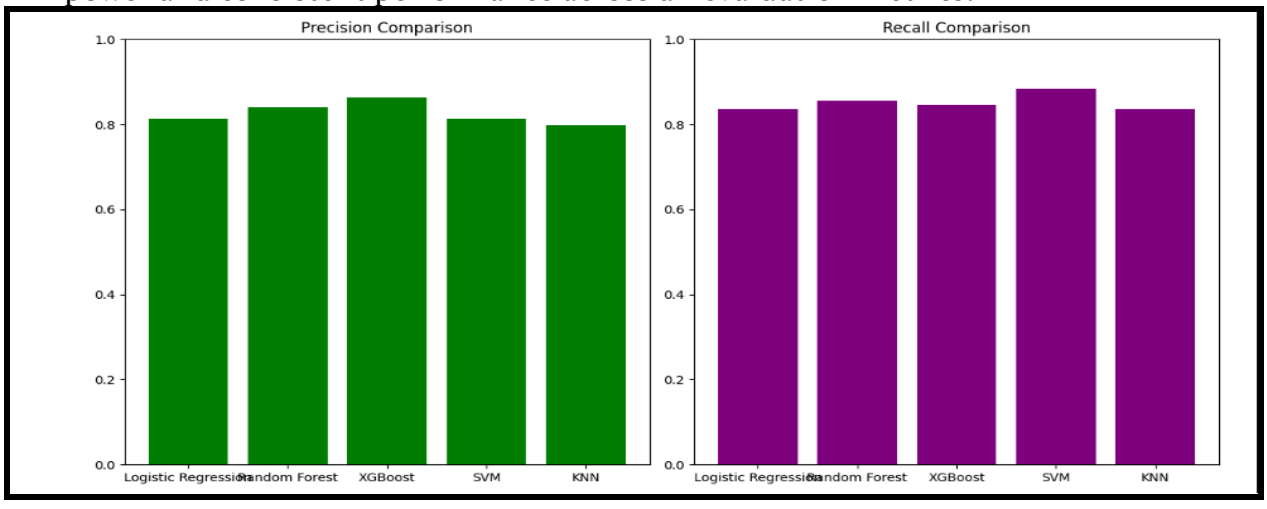
Digital Egypt Pioneers Initiative-DEPI
AI & Data Science - Data Scientist-Round Two
Healthcare Predictive Analytics Project
ONL2_AIS4_S1 Data Scientist Group

K-Nearest Neighbors (KNN)

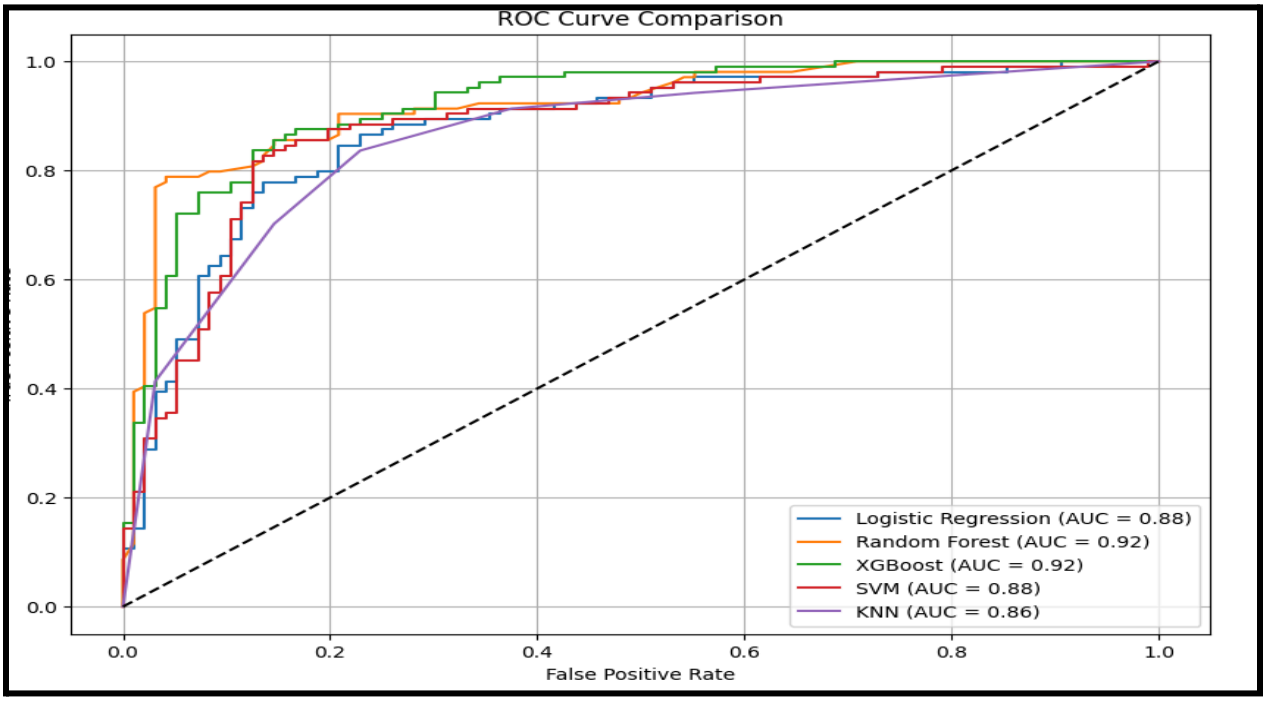
KNN provided a simple, instance-based learning approach that worked reasonably well after scaling. It had decent accuracy but lagged behind other models in precision and recall, particularly for the minority (malnourished) class. Its performance is sensitive to the choice of k and the scale of features, which was mitigated through preprocessing. While interpretability tools like LIME helped explain local decisions, KNN lacks generalizable feature importance, making it less favorable for deployment but useful as a baseline comparison.

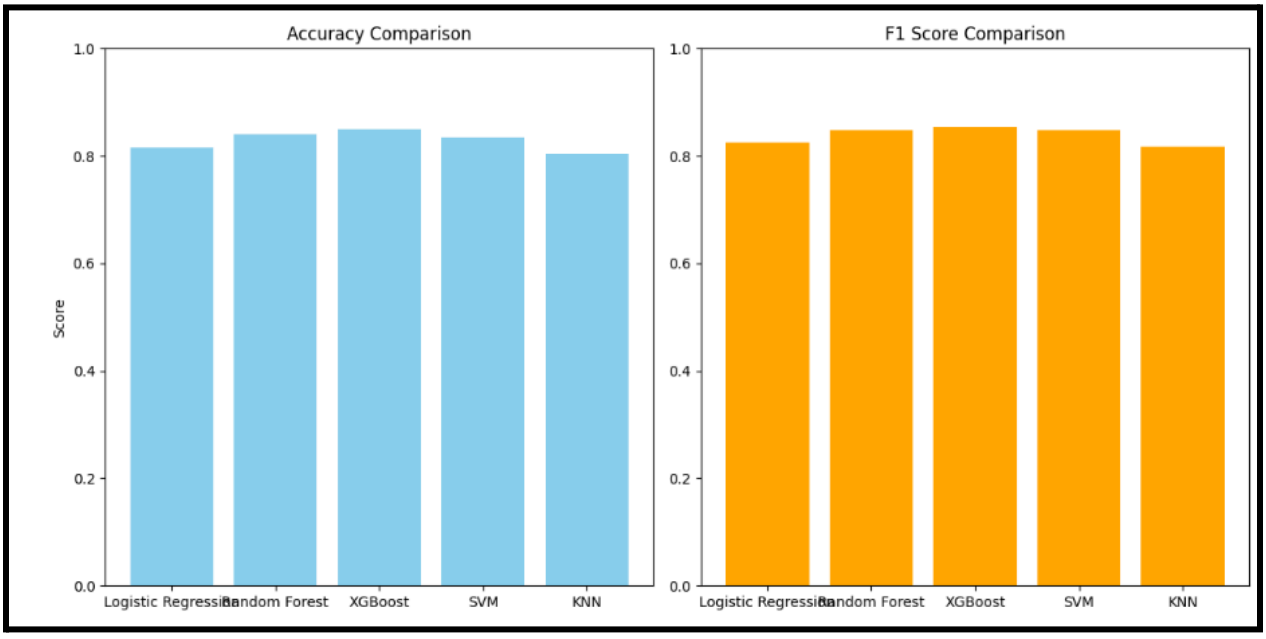
Comparison Between the Models

To evaluate the effectiveness of various classification models in predicting child malnutrition, we compared Logistic Regression, Random Forest, XGBoost, Support Vector Machine (SVM), and K-Nearest Neighbors (KNN) using key performance metrics: accuracy, F1 score, precision, recall, and ROC-AUC. Among all models, **XGBoost** consistently delivered the highest performance, showing strong balance across precision and recall, and achieving the best Area Under the Curve (AUC), making it a robust and reliable choice. **Random Forest** also performed well, particularly in accuracy and interpretability, while **Logistic Regression**, though slightly less accurate, offered transparency and simplicity in decision-making. The **SVM** model achieved competitive precision but slightly lower recall, indicating it may miss some malnourished cases. **KNN** showed moderate performance but was more sensitive to scaling and data structure. Overall, **XGBoost is recommended as the optimal model** due to its superior predictive power and consistent performance across all evaluation metrics.



Digital Egypt Pioneers Initiative-DEPI
AI & Data Science - Data Scientist-Round Two
Healthcare Predictive Analytics Project
ONL2_AIS4_S1 Data Scientist Group





Digital Egypt Pioneers Initiative-DEPI
AI & Data Science - Data Scientist-Round Two
Healthcare Predictive Analytics Project
ONL2_AIS4_S1 Data Scientist Group

Supervised Regression Models Section

In this part of the notebook, the goal is to predict a continuous outcome related to child malnutrition—for example, estimating the average malnutrition rate or burden based on socio-economic, demographic, or health-related features in the dataset. Unlike classification models that predict categories, regression models aim to estimate numeric values.

Linear Regression

Linear Regression was employed as a baseline regression model to predict continuous malnutrition-related outcomes, such as average malnutrition or underweight burden. This model assumes a linear relationship between the independent variables and the target. It is highly interpretable, providing clear coefficients that show the direction and strength of each feature's impact on the outcome. Despite its simplicity, Linear Regression performed reasonably well, especially when the relationship between inputs and outputs was approximately linear. However, it can struggle to model complex patterns or handle multicollinearity without prior feature selection or transformation.

Lasso Regression

Lasso Regression, a regularized version of linear regression using L1 penalty, was used to improve generalization and perform automatic feature selection. By penalizing the absolute magnitude of coefficients, Lasso can shrink less important feature weights to zero, effectively removing them from the model. This makes it especially useful when dealing with many correlated predictors or a large number of features. In this analysis, Lasso not only helped in identifying the most influential variables related to malnutrition but also reduced the risk of overfitting compared to ordinary linear regression. Its balance between simplicity and predictive power made it a valuable tool for feature interpretation.

Digital Egypt Pioneers Initiative-DEPI
AI & Data Science - Data Scientist-Round Two
Healthcare Predictive Analytics Project
ONL2_AIS4_S1 Data Scientist Group

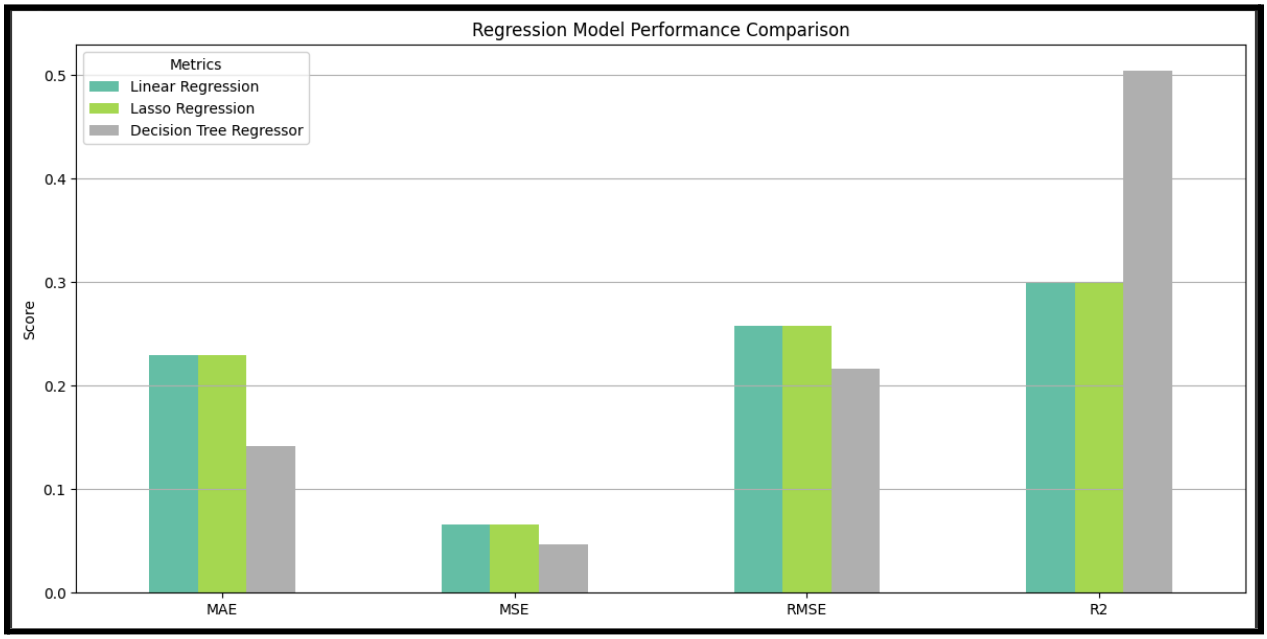
Decision Tree Regressor

The Decision Tree Regressor was chosen to capture non-linear relationships in the data that linear models might miss. Unlike regression models based on a linear equation, decision trees segment the data based on threshold values of input features, making them highly flexible and interpretable. They are well-suited for uncovering interactions and rules in the data, such as specific socio-demographic thresholds that increase malnutrition risk. However, without regularization (e.g., limiting tree depth), decision trees can overfit the training data. In this analysis, the Decision Tree Regressor offered valuable insights into complex, non-linear dependencies within the dataset, although its predictive performance depended on careful tuning.

Comparison Between the Models

In comparing the performance of the regression models—**Linear Regression**, **Lasso Regression**, and **Decision Tree Regressor**—we evaluated them using four key metrics: **Mean Absolute Error (MAE)**, **Mean Squared Error (MSE)**, **Root Mean Squared Error (RMSE)**, and **R-squared (R^2)**. Linear Regression served as a strong baseline with balanced error and explained variance, demonstrating reliable performance under the assumption of linear relationships. Lasso Regression performed similarly to Linear Regression but slightly reduced model complexity by regularizing and potentially eliminating less relevant features, making it more robust to overfitting in the presence of multicollinearity. On the other hand, the Decision Tree Regressor excelled in capturing **non-linear relationships** in the data, which was reflected in its typically **lower errors and higher R^2 score**. However, its tendency to overfit must be managed carefully with hyperparameter tuning. The comparison plot visually confirms these observations, showing the Decision Tree's edge in capturing complex patterns, while the linear models provided stable and interpretable alternatives.

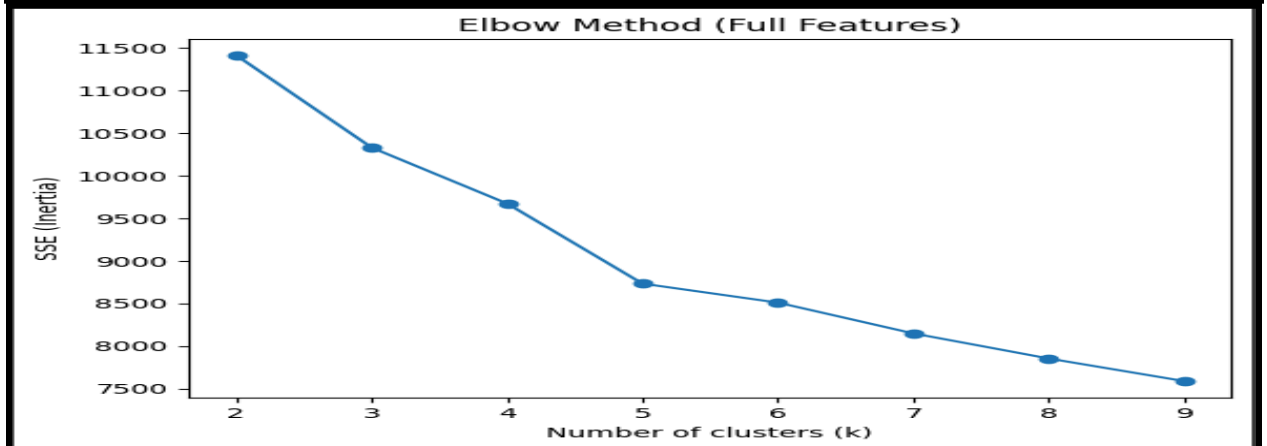
Digital Egypt Pioneers Initiative-DEPI
AI & Data Science - Data Scientist-Round Two
Healthcare Predictive Analytics Project
ONL2_AIS4_S1 Data Scientist Group



Unsupervised Models:

This block begins by applying K-Means clustering to the scaled training dataset. It evaluates different numbers of clusters (k from 2 to 9) by computing the Sum of Squared Errors (SSE), which measures intra-cluster variance. The elbow method is visualized with a plot of SSE against the number of clusters. The point where the SSE starts to level off (the "elbow") indicates the optimal number of clusters—in this case, k=5. This helps ensure the clusters are compact and distinct.

```
#K-Means on full scaled training data
sse = []
k_range = range(2, 10)
for k in k_range:
    kmeans = KMeans(n_clusters=k, random_state=42)
    kmeans.fit(X_train_scaled)
    sse.append(kmeans.inertia_)
```



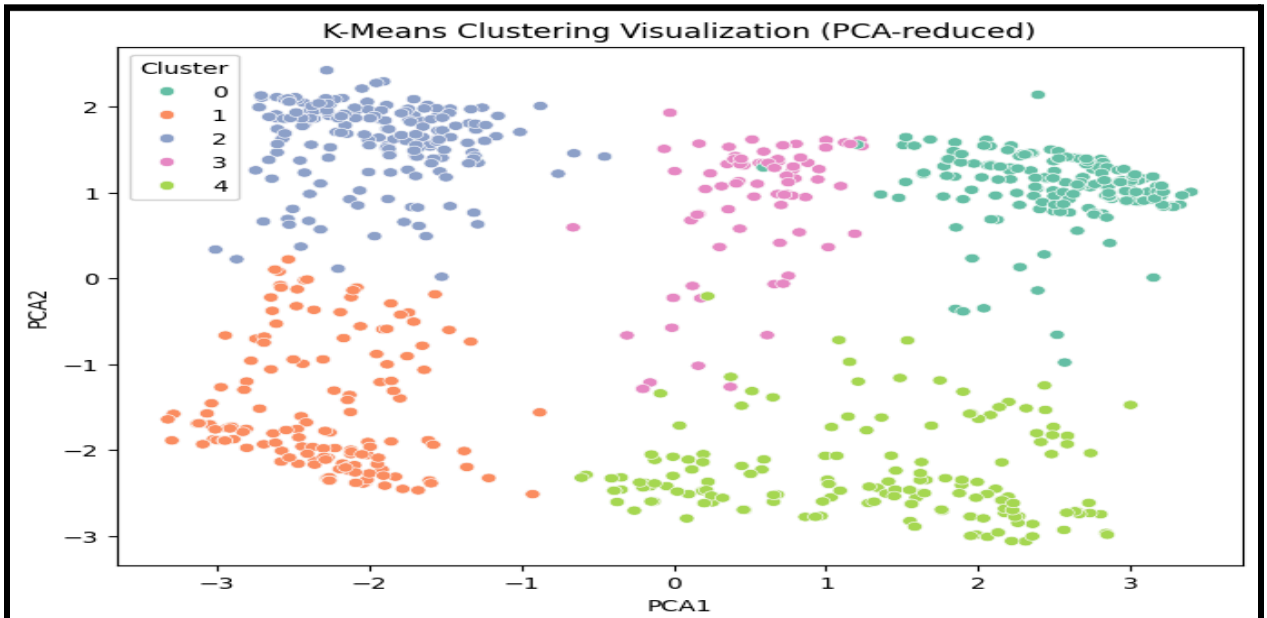
Once the optimal number of clusters is chosen, the model assigns each data point to one of the five clusters. These cluster labels are appended to the original feature set (excluding the target), allowing for post-clustering interpretation. The groupby summary calculates the average value of each feature within each cluster, helping identify the dominant characteristics of each group.

```
k = 5  # optimal according to plot
kmeans = KMeans(n_clusters=k, random_state=42)
clusters = kmeans.fit_predict(X_train_scaled)

#Append cluster labels to the original data for interpretation
clustered_data = X_train.copy()
clustered_data['Cluster'] = clusters

#Analyze cluster characteristics
cluster_summary = clustered_data.groupby('Cluster').mean()
print("Cluster Summary:\n", cluster_summary)
```

To visualize the high-dimensional clustering results, Principal Component Analysis (PCA) reduces the data to two dimensions. A scatter plot is created using the first two principal components, with each point colored by its cluster label. This gives an intuitive view of how well-separated the clusters are in lower-dimensional space and helps spot overlapping or dominant clusters.

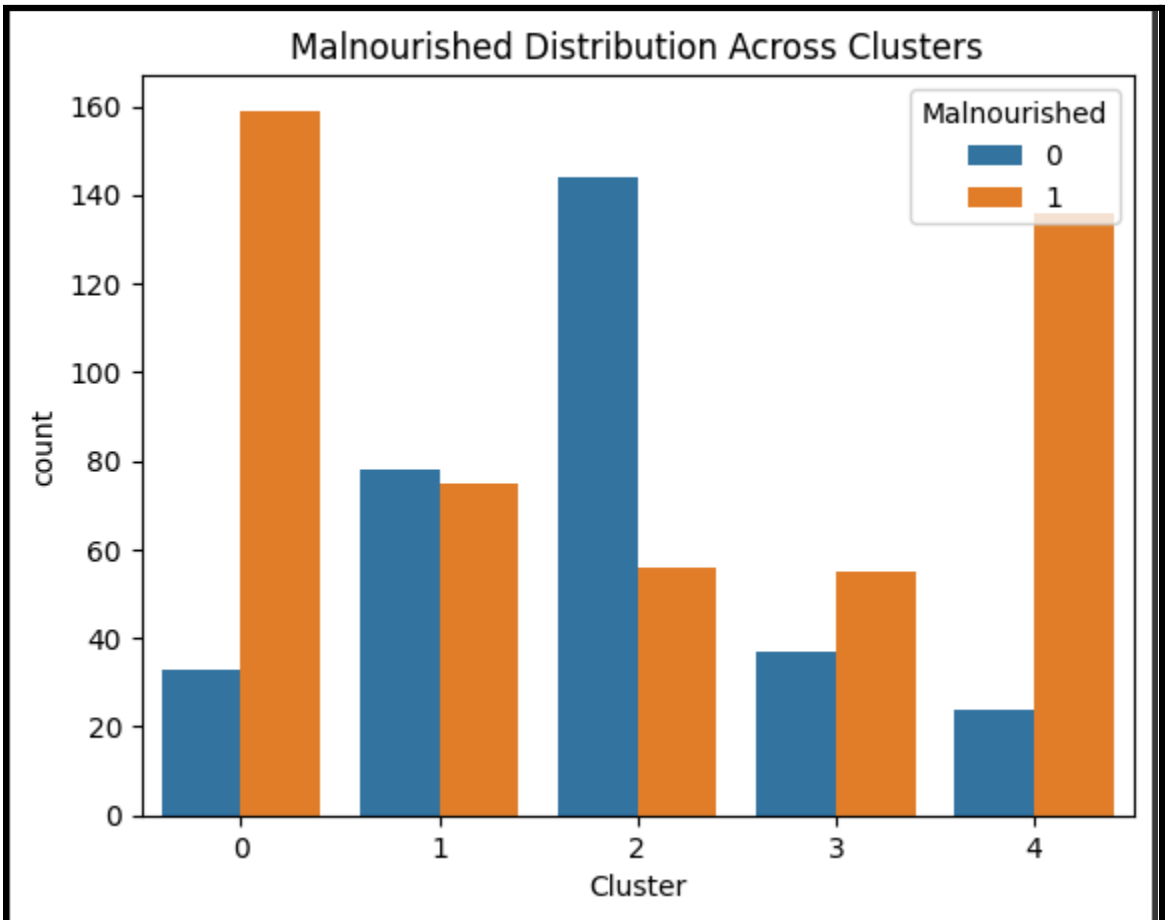


Digital Egypt Pioneers Initiative-DEPI
AI & Data Science - Data Scientist-Round Two
Healthcare Predictive Analytics Project
ONL2_AIS4_S1 Data Scientist Group

To understand how individual features vary across clusters, box plots are generated for the top 5 features. This allows visual assessment of how each feature is distributed within each cluster and helps in identifying which features most influence the cluster structure.

```
Silhouette Score (Full Features): 0.18
```

In this step, the unsupervised clusters are compared with the supervised Malnourished target. A count plot shows how malnourishment is distributed across each cluster. This comparison reveals whether the clusters capture meaningful distinctions related to malnutrition status, even though clustering was done without using this label.
Finally, the contribution of original features to each PCA component is examined. By sorting the loadings of the first principal component, we identify which variables contribute most to the PCA projection. This insight clarifies which original features had the strongest influence on how the data was separated in the 2D PCA visualization



Digital Egypt Pioneers Initiative-DEPI
AI & Data Science - Data Scientist-Round Two
Healthcare Predictive Analytics Project
ONL2_AIS4_S1 Data Scientist Group

Initial Hyperparameter Tuning with Grid Search

In the first phase of model optimization, GridSearchCV is employed to systematically search through combinations of hyperparameters for the XGBoost classifier. The grid includes ranges for n_estimators, max_depth, learning_rate, subsample, and colsample_bytree, allowing the model to balance complexity and generalization. After fitting this grid search on the training set, the best-performing model is selected based on cross-validated accuracy. The resulting model significantly improves prediction accuracy and is further evaluated with a classification report and confusion matrix.

Model Comparison: Before vs. After Grid Search

To visualize the impact of hyperparameter tuning, ROC curves are plotted for the original (untuned) and optimized XGBoost models. The Area Under the Curve (AUC) improves post-optimization, illustrating better discrimination between classes. This validates that the grid-searched model not only performs better in accuracy but also has improved sensitivity and specificity in classification tasks.

Feature Importance Analysis

After tuning, the feature importance scores of the optimized model are plotted. This highlights which features contribute most significantly to the prediction task. Such insights are valuable for domain experts, offering interpretability and guidance on which variables might be most influential in determining malnutrition outcomes or other target conditions in the dataset.

Digital Egypt Pioneers Initiative-DEPI
AI & Data Science - Data Scientist-Round Two
Healthcare Predictive Analytics Project
ONL2_AIS4_S1 Data Scientist Group

Cross-Validation Performance

The model's performance is further validated using cross-validated accuracy from the grid search process. This metric gives a reliable estimate of how well the model is expected to generalize to unseen data, confirming the robustness of the selected hyperparameters.

Overfitting Check and Further Refinement

Initial tuning revealed a small performance gap between training and testing accuracy, suggesting potential overfitting. To address this, a more refined and restrictive hyperparameter grid is defined, introducing regularization parameters like gamma, reg_alpha, reg_lambda, and min_child_weight. This helps constrain the model complexity and encourages better generalization.

Final Model Evaluation with Regularization

With the updated grid, a second round of GridSearchCV is conducted. The best model from this search yields a better balance between training and test accuracy, as seen in the updated evaluation metrics and confusion matrix. These regularized parameters reduce overfitting while maintaining strong performance on the test set.

Digital Egypt Pioneers Initiative-DEPI
AI & Data Science - Data Scientist-Round Two
Healthcare Predictive Analytics Project
ONL2_AIS4_S1 Data Scientist Group