Report : Nutrient Analysis on food dataset

1. Project Overview

The project focuses on analyzing a **Food Composition Dataset** to derive meaningful insights about nutrients, vitamins, and minerals using **Big Data analytics with PySpark**.
The goal is to:

- Understand the nutritional composition of various foods.
- Identify foods beneficial for specific health goals.
- Perform clustering to discover hidden patterns among food items.
- Visualize nutrient distributions and relationships.

2. Technologies Used

Category	Tool/Technology
Programming Language	Python
Big Data Framework	Apache Spark (PySpark)
Machine Learning Library	Spark MLlib
Visualization	Matplotlib
Data Format	CSV
Environment	Jupyter Notebook / PySpark Shell

3. Dataset Description

Dataset Name: Food Component (Nutritional Dataset)

File: `food.csv`

Total Records: ~1000+ food items (depending on dataset version)

Attributes:

`Data_Protein` – Protein content

`Data_Carbohydrate` – Carbohydrate content

`Data_Fat_*` – Various fat subtypes

`Data_Vitamins_*` – Vitamins (A, B, C, D, E, K, etc.)

`Data_Minerals_*` – Minerals (Calcium, Iron, etc.)

`Data_Energy_kcal` – Caloric value

Other descriptive columns (Food group, Source, etc.)

4. Methodology

Step 1: Data Loading

The dataset was read into a PySpark DataFrame:

```
df = spark.read.csv("food.csv", header=True, inferSchema=True)
```

Step 2: Data Cleaning

Removed special characters from column names (spaces, dots, hyphens, etc.).

Identified all **nutrient-related columns** (prefix `Data_`).

Used `try_cast()` to safely convert textual numeric columns into floats:

```
df = df.withColumn(c, expr(f"try_cast({c} as float)"))
```

Dropped rows with all `NULL` nutrient values.

Result:
A clean, numeric dataset suitable for analysis and ML operations.

5. Exploratory Data Analysis (EDA)

A. Summary Statistics: Computed `mean`, `min`, and `max` for all nutrient columns using PySpark aggregations.

Example: `df_clean.agg(F.mean("Data_Protein"), F.max("Data_Protein")).show()`

B. Visualization

Bar charts of average nutrient values.

Pie charts of mean vitamin/protein distribution.

Histograms and boxplots for nutrient distributions.

Insights:

Foods vary widely in carbohydrate and fat content.

Vitamin A and C-rich foods dominate certain clusters.

Protein distribution is skewed with outliers (e.g., meat, legumes).

6. Health-Based Food Recommendations

Health Focus	Criteria Used	Example Output
High Protein Foods	Highest Data_Protein	Chicken breast, Lentils
Diabetes-Friendly	Low Data_Carbohydrate + Low Data_Sugar_Total	Spinach, Eggs
Bone Health	High Data_Calcium, Data_Vitamins_Vitamin_K	Milk, Broccoli
Heart Health	Low Data_Fat_Saturated_Fat, High Data_Fiber	Oats, Almonds

Each group was retrieved using **PySpark's MapReduce-style sorting and filtering**:

```
df_clean.orderBy(df_clean['Data_Protein'].desc()).limit(10).show()
```

7. Machine Learning Analysis

A. Feature Engineering

Selected features: `Protein`, `Carbohydrate`, `Fat`, `Fiber`, `Vitamins`

Combined using **VectorAssembler**:

```
assembler = VectorAssembler(inputCols=features, outputCol="features")
```

```
df_features = assembler.transform(df)
```

B. KMeans Clustering

Performed **unsupervised clustering** with `k=4`:

```
kmeans = KMeans(k=4, seed=42)

model = kmeans.fit(df_features)
```

Each food item was assigned to a cluster (e.g., high-protein group, high-carb group).

C. Correlation Analysis

Computed correlation matrix among nutrients:

```
corr_matrix = Correlation.corr(df_features, "features").head()[0]
```

This identified **strong relationships** (e.g., Vitamin A ↔ Beta Carotene).

8. Dimensionality Reduction & Visualization (PCA)

Applied **Principal Component Analysis (PCA)** to reduce feature space to 2 dimensions.

Visualized clusters using Matplotlib scatter plot:

```
plt.scatter(pca_x, pca_y, c=cluster_labels)
```

Interpretation:

Cluster 0: High-carb, high-calorie foods.

Cluster 1: Low-fat, high-vitamin foods.

Cluster 2: Balanced nutrient foods.

Cluster 3: High-protein, high-fat foods.

9. Results & Insights

Observation	Description

Observation	Description
Data Quality Improved	Removed invalid numeric entries and renamed columns for consistency.
Nutrient Trends Found	Certain foods dominated in specific nutrients (e.g., Vitamin C in citrus fruits).
Health Grouping Effective	Identified foods suitable for heart, bone, and diabetic health.
Clusters Formed	Nutrient-based grouping gave meaningful results reflecting food types.
Correlations	Vitamin pairs and fat subtypes showed strong linear relationships.

10. Future Enhancements

Integrate **streaming food data** (e.g., from APIs or IoT devices).

Use **Spark GraphFrames** for relationship mapping between foods.

Deploy a **Food Recommendation Web App** using Streamlit + Spark backend.

Integrate with **AWS EMR** or **Databricks** for large-scale data handling.

11. Conclusion

This project demonstrates how **Big Data Analytics** using **PySpark** can process, clean, and analyze complex nutritional datasets efficiently.
The integration of **machine learning (KMeans, PCA)** and **visualizations** provides both analytical and predictive capabilities for food science and health-related decision-making.

12. References

Apache Spark Documentation – https://spark.apache.org/docs/latest/api/python/

U.S. Department of Agriculture (USDA) Food Composition Database

PySpark MLlib Guide – https://spark.apache.org/mllib/

Matplotlib Official Docs – https://matplotlib.org/stable/contents.html