# NutriGene Explorer™ - README

## Overview:

NutriGene Explorer™ enables researchers to explore the interaction between nutrition and genetic data. By combining datasets from nutritional databases and genetic repositories, it reveals key insights into how specific nutrients impact gene expression and health.

## Features:

● Nutrient-gene interaction analysis: Investigates how vitamins, minerals, and other nutrients affect genetic expression.

● Data import/export: Supports multiple formats, including CSV, JSON, and Excel.

● AI models: Uses regression models to predict nutrient impacts on gene expression.

● Visual analytics: Provides charts and graphs to represent the relationship between nutrition and genetics.

## Installation Instructions:

1. Prerequisites:

○ Python 3.7 or later

○ Libraries: Pandas, NumPy, Scikit-learn

Install libraries:

pip install pandas numpy scikit-learn

2. Setup:

○ Download the tool and load the nutritional and genetic data.

○ Run the main Python script to start analyzing the data.

## User Guide:

Step 1: Upload Nutritional and Genetic Data  
Upload nutritional data from sources like the USDA Nutrient Database and genetic data from repositories like Ensembl.

Step 2: Set Analysis Variables  
Define which nutrients and gene expressions you want to study. You can select multiple variables for multi-dimensional analysis.

Step 3: Run Analysis  
Click Run Analysis. The tool will apply AI models to analyze the interaction between nutrient intake and gene expression.

Step 4: View Results  
View graphs and charts representing nutrient-gene interactions. Export data for further research or presentation.

## FAQ:

1. What types of datasets can I use?  
NutriGene Explorer™ accepts CSV and Excel formats containing nutritional and genetic data.

2. Can I customize the AI models?  
Yes, the tool allows customization of AI model parameters, such as the type of regression model used.

## API Documentation:

This application provides an API to explore nutritional and genetic data. Below are the endpoints:

● GET / - Home route; returns a welcome message.

● GET /participants - Fetches a list of all participants.

● GET /participant/<participant\_id> - Retrieves detailed information for a specific participant, including NHANES and genetic data.

● POST /predict - Predicts long-term health impacts based on participant data.

## Code Example:

from flask import Flask, jsonify, request  
from flask\_cors import CORS  
import pandas as pd  
  
app = Flask(\_\_name\_\_)  
CORS(app)  
  
# Load data  
nhanes\_file = "nhanes\_data.csv"  
genetic\_file = "genetic\_data.csv"  
nhanes\_data = pd.read\_csv(nhanes\_file)  
genetic\_data = pd.read\_csv(genetic\_file)  
  
@app.route('/')  
def home():  
 return "Welcome to the Nutritional Impact Predictor API!"  
  
@app.route('/participants', methods=['GET'])  
def get\_participants():  
 participants = nhanes\_data['ParticipantID'].tolist()  
 return jsonify({"participants": participants})

## Requirements:

To run the application, install the following dependencies:

● flask

● flask-cors

● pandas

## Links:

GitHub Repository: https://github.com/YsD9954/jmedia\_07

API Link: <https://jmedia-07.onrender.com>  
  
  
attached below is the ouput screen shot on postman  
  
  
