# **COMPUTATIONAL GENOMICS PROJECT**

# **REPORT**

# 

# **by**

# Tacettin Batuhan BOSTANCI

# **Student ID Number**

# 110 510 179

# **Date of the Submission**

23.01.2023

INTORDUCTION:

The bioBakery 3 platform has been used to analyze 1262 metagenomes from colon cancer (CRC) patients, and has revealed novel connections between colon cancer and the microbiome. The platform includes tools for taxonomic, strain-level, functional, and phylogenetic profiling of metagenomes that have been newly developed to build on the largest set of reference sequences now available. These methods, such as MetaPhlAn 3 and HUMAnN 3, have been designed to increase the accuracy and resolution of microbial community studies, and have been used to uncover new insights about the relationship between colon cancer and the gut microbiome.

As previously mentioned, the bioBakery 3 platform was used to analyze 1262 metagenomes from colon cancer (CRC) patients from 10 different countries. The aim of this project is to analyze the results of the classification model by over-training this data in a cumulative way. This approach allows for a deeper understanding of the diversity of the microbiome across different populations and could potentially lead to the identification of new disease-microbiome associations specific to different geographic regions. Additionally, by over-training the data, the model can be fine-tuned to become more accurate in identifying specific microbes and their potential roles in colon cancer. Overall, this study aims to improve our understanding of the complex relationship between colon cancer and the gut microbiome.

DATASET:

As previously discussed, our data set includes 1262 metagenomic samples collected from individuals diagnosed with colon cancer across 10 different countries. One of the major challenges in microbial community research is the high degree of inter-individual variation, even among individuals with the same disease. This makes it difficult to identify patterns and associations that are consistent across populations. However, by analyzing a large and diverse data set such as the one used in this study, it is possible to gain a more comprehensive understanding of the variations and similarities in the gut microbiome across different populations. Additionally, by examining our data in a cumulative way, we can identify specific microbes and their functional potentials that are associated with colon cancer across different geographic regions, which can improve our understanding of the underlying mechanisms of colon cancer and help identify potential new diagnostic and therapeutic targets.

A screenshot of a computer screen

Description automatically generated with medium confidence

Figure1

"In Figure 1, we can see the basic information of our dataset."

Text

Description automatically generated

Figure2

Methodology:

"By utilizing the info method of the DataFrame, we are able to obtain a basic understanding of our dataset. As seen in the output, there are three columns of object data type. This may pose a problem for our classification task and will need to be addressed later."

Text

Description automatically generated

Figure3

"Our dataset appears to be devoid of missing values, as indicated by the absence of null values in the rows. This is favorable as it eliminates the need to address the issue of missing data, which can often have a significant impact on the analysis."

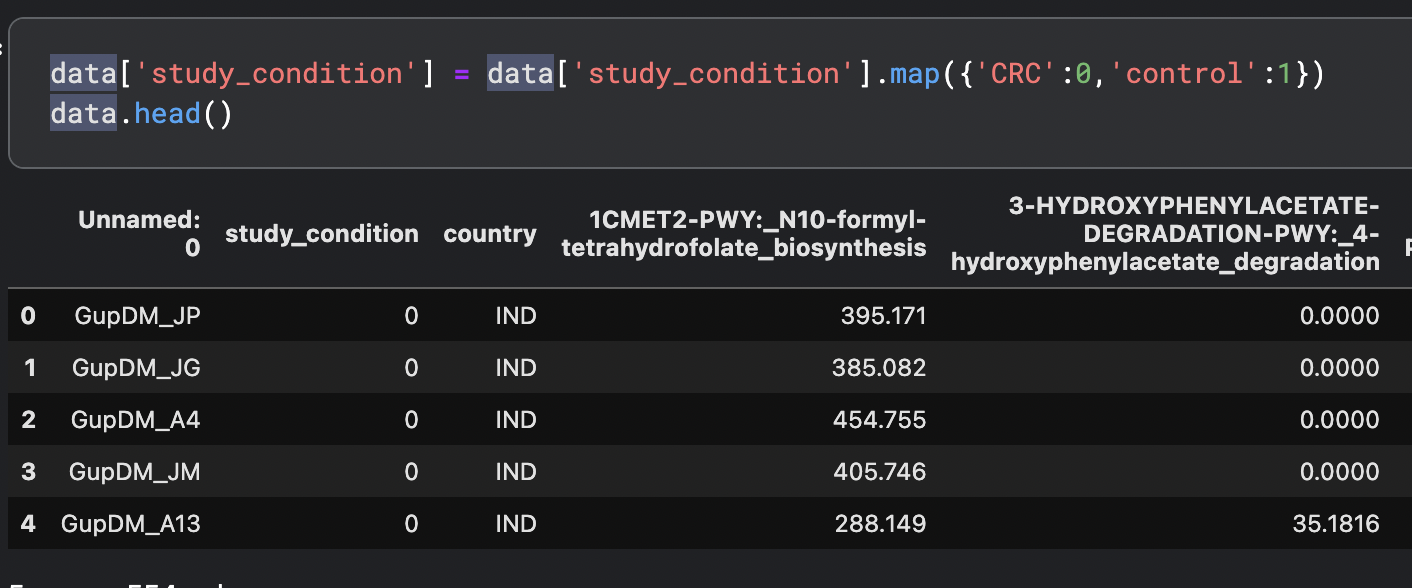


Figure4

"As previously noted, the presence of object data type columns requires attention. One approach to addressing this issue is through data conversion, such as converting the 'controller condition' and 'CRC' columns to numeric data types. Specifically, values of '1' and '0' respectively."

Text

Description automatically generated with low confidence

Figure5

Dropping the unnecessary column.

Now, we can pass the Principal Component Analysis part. Why we use it firstly: Principal Component Analysis (PCA) is a dimensionality reduction technique that is commonly used in data analysis and visualization. It is used to transform a high-dimensional dataset into a lower-dimensional one, while preserving as much of the variation in the data as possible.

Chart, scatter chart

Description automatically generated Chart, scatter chart

Description automatically generated

"Upon examination of the data, it is apparent that the range of values spans from -200000 to 200000. Additionally, it is observed that certain countries appear to encompass the data ranges of other countries. This may have implications on the model performance and results, however, further analysis and evaluation will be conducted to determine the impact."

"To further comprehend the underlying structure of the data, t-SNE can be applied on the PCA transformed data. This approach allows for a more comprehensive understanding of the relationships between the various data points."

Chart

Description automatically generated Chart

Description automatically generated

"The t-SNE visualization reveals a wide range of data points, indicating a high degree of variability among the countries. This suggests that the situation of each country is relatively independent of one another."

Now, we can divided the sub\_data.

Before progressing to the modeling stage, it is important to establish a clear strategy for dividing the data into sub-datasets. The goal is to examine the relationship between countries, thus, the following approach will be adopted:

* Initially, the entire dataset will be used to train the model.
* Next, the dataset will be subdivided based on continent, and each continent's sub-dataset will be used to train the model.
* Finally, the sub-datasets of individual countries will be cumulatively added to the model, one after the other, in order to analyze the performance of the model for each individual country.

We can see the divided countries below part:

alli = [AUT, CHN, DEU, FRA, IND, ITA, JP, JPN, USA]

mix1 = [FRA, IND, ITA, JP, AUT, CHN, DEU, JPN, USA]

mix2 = [USA, AUT, FRA, CHN, IND, JP, ITA, JPN, DEU]

us = [USA]

europe = [AUT, DEU, FRA, ITA]

asia = [CHN, IND, JP, JPN]

MLP Model:

The above code defines a neural network model using Keras library. The model has an input layer with 551 features, followed by several hidden layers (5 layers in total) with decreasing number of neurons, and an output layer with a single neuron. The activation function used in the hidden layers is 'relu' and the output layer uses 'sigmoid' activation function. The model is then compiled with the 'adam' optimizer, 'binary\_crossentropy' loss function and 'accuracy' as metrics. An early stopping callback is also defined with a patience of 25 epochs to stop training if the validation loss does not decrease. The model is then trained on a set of data (alli) and the performance is evaluated using accuracy and F1-score.

It's worth mentioning that the code also uses dropout and L1 and L2 regularization to reduce overfitting. Dropout regularization is used in the first hidden layer by randomly dropping out neurons during training. L1 and L2 regularization are used in all hidden layers to further reduce overfitting by adding penalties to the weights of the neurons. The L1 and L2 regularization values are set to 0.01, which helps to prevent the model from becoming too complex and overfitting the training data. These L1 and L2 regularization are also known as Lasso regularization, which is a powerful technique to shrink the importance of less important features in a model.

The code uses several hyperparameters to define and train the model. Some of the key hyperparameters used in this code are:

Input shape: The input layer of the model takes in 551 features

Hidden layers: The model has 5 hidden layers with decreasing number of neurons (128, 64, 32, 16, 4)

Activation function: The hidden layers use 'relu' activation function and the output layer uses 'sigmoid' activation function

Optimizer: The model is optimized using 'adam' optimizer

Loss function: The model is trained using 'binary\_crossentropy' loss function

Metrics: The model's performance is evaluated using 'accuracy' metric

Dropout: The first hidden layer uses a dropout rate of 0.2 to reduce overfitting

L1 and L2 regularization: All hidden layers use L1 and L2 regularization with a value of 0.01 to reduce overfitting

Batch size: The model is trained using a batch size of 16

Number of epochs: The model is trained for 128 epochs

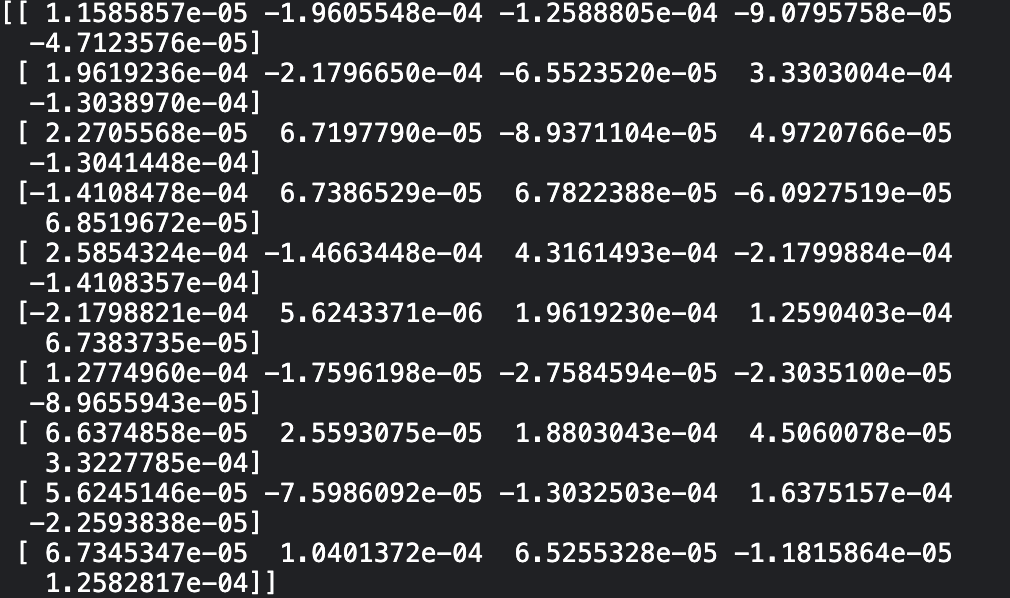
Early stopping: The training process is stopped if the validation loss does not decrease for 25 epochs.

These hyperparameters were chosen based on the size and characteristics of the dataset, and can be adjusted to improve the performance of the model. It's worth noting that the best hyperparameters for a model often need to be found through trial and error and by testing different combinations of them.

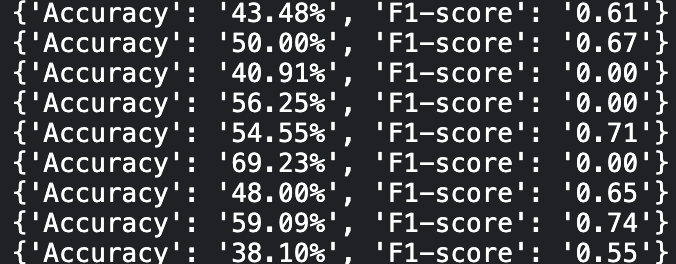
Model1(alli = [AUT, CHN, DEU, FRA, IND, ITA, JP, JPN, USA]):



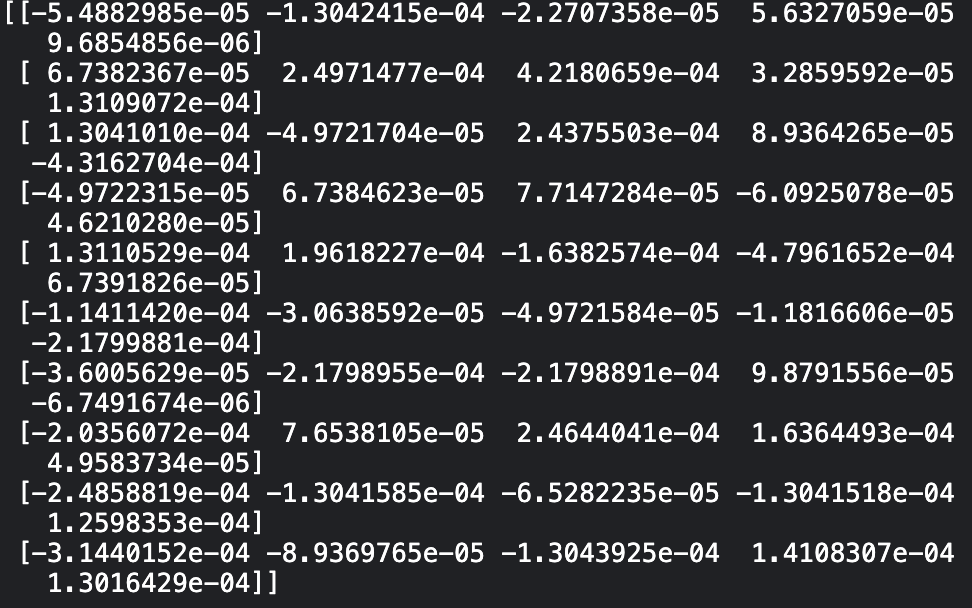
Based on the accuracy results of the model when applied to different countries, it appears that the model may not perform well when presented with new data from other countries. This suggests that the model may not generalize well to new data. Additionally, the feature importance scores of the model in below.



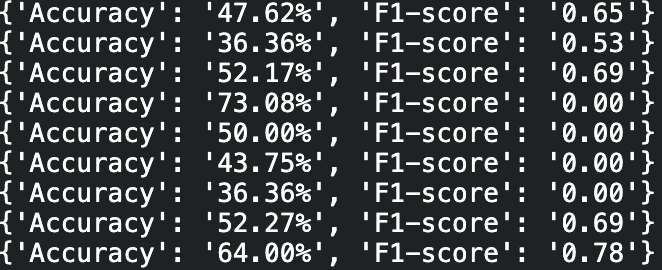
Model2(mix1 = [FRA, IND, ITA, JP, AUT, CHN, DEU, JPN, USA]):



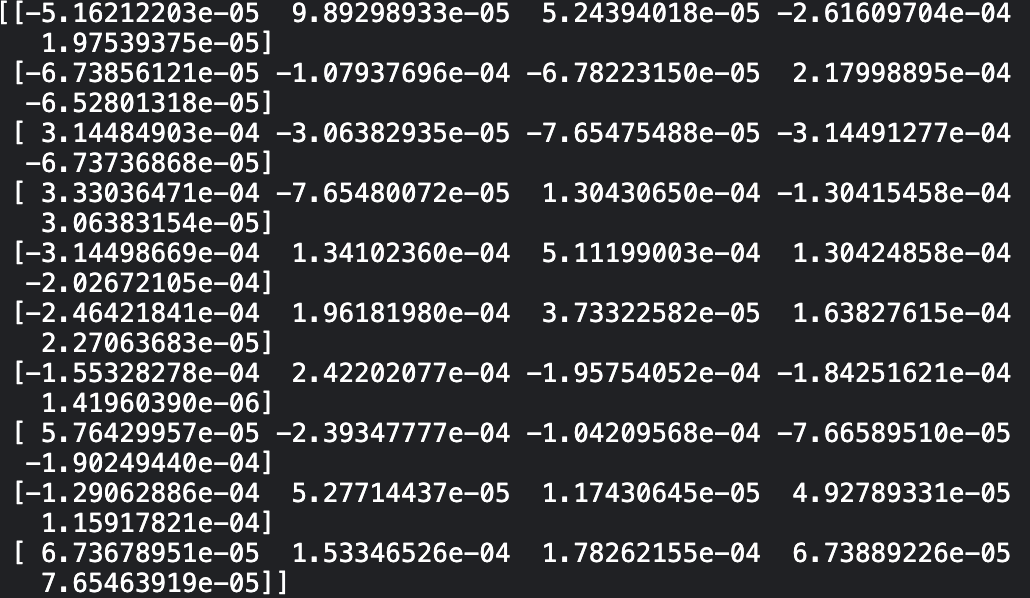
When using data from all countries for model training, we observe that the model's performance improves. This suggests that training the model on a diverse set of data from multiple countries can lead to better accuracy compared to training on data from specific countries alone. Additionally, the feature importance scores of the model in below. Additionally, the feature importance scores of the model in below.



Model3(mix2 = [USA, AUT, FRA, CHN, IND, JP, ITA, JPN, DEU]):



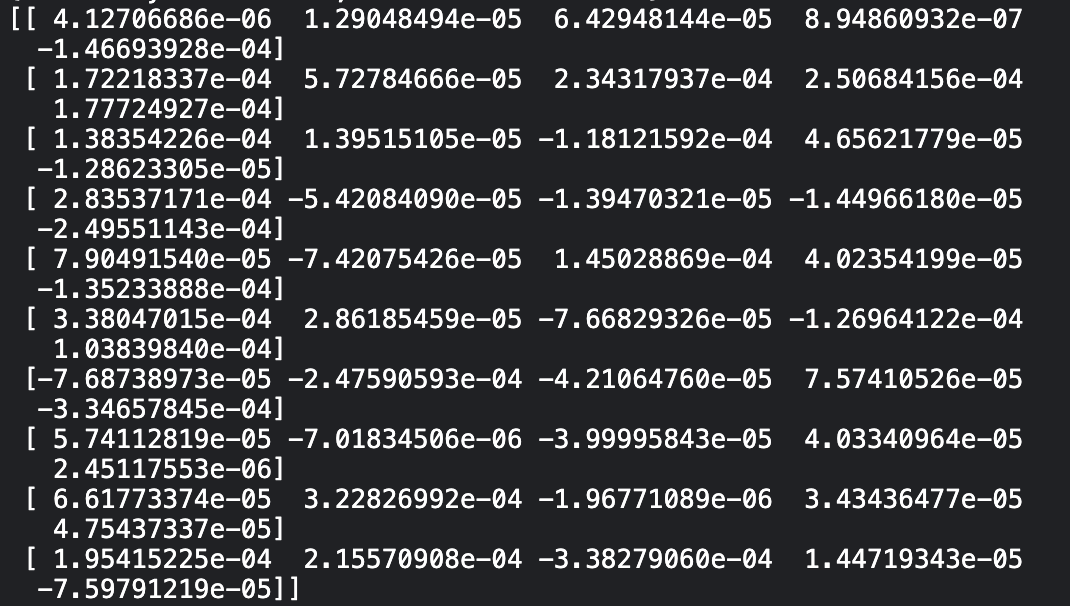
Finally, when we use data from all countries in the model, we observe that the model's performance is similar to that of the second model. It is worth noting that when data from China is included in the model, the model's accuracy increases. This suggests that Chinese data may be particularly useful for improving the model's performance. Additionally, the feature importance scores of the model in below.



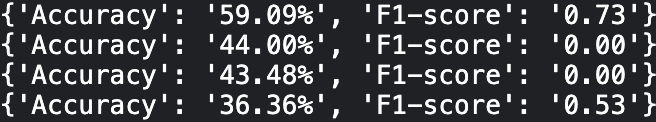
Model4(us = [USA]):



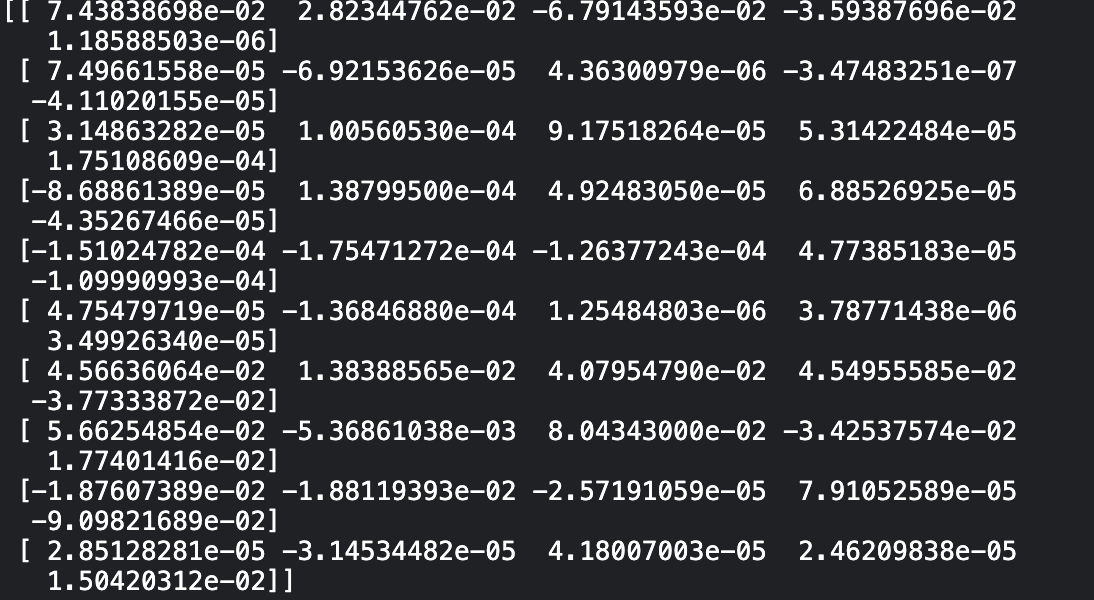
When training the model using data from the United States, the model's performance is not successful. One potential reason for this could be the large population distribution within the United States, which may affect the model's ability to generalize. Additionally, the feature importance scores of the model in below.



Model5(europe = [AUT, DEU, FRA, ITA]):



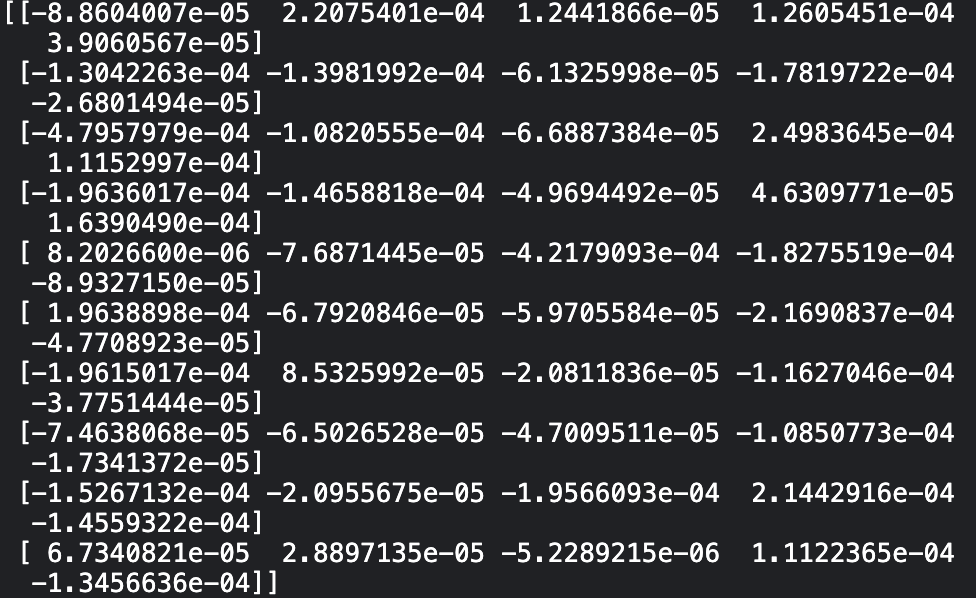
Similarly, when the model is applied to data from various European countries, we do not observe a significant decrease in accuracy as new countries are added cumulatively to the model. This suggests that the cultural factors from different European countries do not have a major impact on the model's performance. Additionally, the feature importance scores of the model in below.



Model6(asia = [CHN, IND, JP, JPN]):



When the model is applied to data from Asian countries, we observe that the inclusion of data from India has a negative effect on the model's performance. However, when data from Japan is added, the model's accuracy increases. This suggests that the cultural factors from China and Japan have a positive impact on the model's performance, and may be similar in nature. Additionally, the feature importance scores of the model in below.



Result:

In this study, we evaluated the performance of a model when applied to data from various countries. When the model was trained and tested using data from individual countries, we observed that the model's performance was not consistent and may have decreased when new countries were added to the dataset. This suggests that the cultural factors of countries may have a significant impact on the performance of the model.

Furthermore, when the model was applied to data from all countries, we noticed that the performance of the model did not improve consistently and may have even worsened in some cases. This highlights the importance of considering cultural factors such as food and beverage in the context of biological studies.

On the other hand, when we focused on Asian countries, we observed that China and Japan had similar effects and tended to improve the model's performance. This suggests that cultural characteristics shared by these countries may be beneficial for the model.

In conclusion, this study highlights the importance of considering cultural factors in the context of biological studies and the need to gather diverse data from multiple countries to train models that generalize well. Additionally, it's worth noting that Chinese and Japanese data may be particularly useful in improving the model's performance.