Research proposal – Anomaly Detection in Microbiome

1. Introduction & background

Human Microbiome:

The human Microbiome is a term that refers to 10-100 trillion symbiotic microbial cells in the human body, primarily bacteria in the gut. Some of these microbes are essential for maintaining health.

Recent years saw a surge in high-throughput microbiome datasets. One of the most prominent microbiome datasets is called HMP, The Human Microbiome Project. This research was conducted to improve understanding of the microbiota involved in human health and disease. In order to reach this goal, a repository of diverse human microbiome datasets was collected. This datasets include a total of 9285 samples of healthy individuals.

Human microbiome datasets have some unique characteristics such as: highly multivariate, as it composed of many features. It is a high dimensionality data and compositionality. In addition, an individual's microbiome can vary substantially in a short period of time, and different healthy individuals may have almost completely different microbiome composition. These unique characteristics of human microbiome require a special testing method for that task of identifying abnormal samples.

Anomaly Detection:

Anomalies are samples that have different characteristics from normal samples.

Anomaly Detection is the process of identifying abnormal samples in a dataset.

There are several methods for detecting anomalies, a well-known approach is Isolation Forest. This method creates random trees from the samples in the dataset. Each tree in the forest is constructed from sub-sampling of the dataset. Attributes and split point value are randomly selected. The method uses two anomalies' properties: they are few and different. Therefore, anomalies are more susceptible to isolation than normal samples and are isolated closer to the root of the tree.

Anomaly Detection in Microbiome:

An example for microbiome specific anomaly detection algorithm is CLOUD, published by Montassier et al. This is an anomaly detection test for microbiome samples designed to address the difficulties mentioned above. This method works as follows:

a. Running KNN algorithm on each reference subject.
b. Calculate the average ecological distance from that subject to its selected neighbours: di.
c. Calculate the average of all neighbourhoods: $\bar{d} = \frac{\sum_{i=1}^{n} d_i}{n}$
d. Running phase 'a' on test subject. Calculate : $\frac{d_j}{\bar{d}}$.
e. Calculate an empirical outlier percentile for the test subject as the fraction of reference outlier detection test greater than or equal to the test subject's outlier detection test.

A possible limitation of this algorithm is that it is not designed for identifying outliers from within the reference population.

2. Research design/methodology

Here, we will test a new microbiome specific unsupervised anomaly detection method called Omnipotent Microbiome Rigorous Isolation (OMRI). This method works as follows (for full pseudo code see appendix):
OMRI Isolation Forest is based on Isolation Forest method.
The dataset contains n microbiome samples and m features, taxa.
The new method uses the technique of building a forest of trees:
While building a tree, for each inner node in the tree we randomly choose k taxa out of m, Then Calculate distance matrix between the samples contained in the node based only on the k selected taxa. This distance matrix is the input for PCoA algorithm. We define P as the vector of each sample's projection on the PCoA's 1st component. . The samples are divided in this node by a random threshold T: $\min\{P\} \leq T \leq \max\{P\}$. This process is repeated for this nodes' descendants and stops when a node contains r samples at most or when the resulting tree is of depth d.
After this process of forest construction, anomaly score is given for each sample in the dataset by calculating the average depth of the sample across all trees in the forest.

The study will be conducted as follow: first we will use existing anomaly detection techniques, such as Isolation Forest, as benchmarks trying to detect outliers on several datasets, including HMP, as explained in the validation scenarios. Then we will implement Omnipotent Microbiome Rigorous Isolation-forest.

Afterwards we will test our method in the several validation scenarios: first we will combine multiple gut microbiome samples with several microbiome samples from other body sites, considering the latter as anomalies. Second, we will combine temporal dataset of an individual with few samples of different people. In this validation scenario we assume that longitudinal samples from the same individual are more similar than between different individuals. Third, we will combine "sick microbiomes" samples with "healthy microbiome" dataset.

We will test the performance of the algorithm by calculating the average depth of the anomalies samples we added over all trees in OmriForest, d_a. We will define d_b as the average depth of the normal samples. As we explained above, low values of d indicate that the sample is abnormal. If d_a is lower than d_b, it means that on average the algorithm detects anomalies correctly. Since this method is based on PCoA, the "testing" happens on the training set. Namely, we a sample cannot be tested for anomality if it wasn't in a part of the training set.
Hopefully, after proving that this method works, we will check whether anomaly detection improves downstream analysis.

Appendix:

Algorithm 1: $OmriForest(X, t = 100, \psi, l = ceiling(\log_2 \psi))$

```
Inputs: $X - input\ data, t - number\ of\ trees, \psi - sub - sampling\ size$
Output: a set of $OmriTrees$
Initialize $Forest$
For $i = 1$ to $t$ do
    $Forest \leftarrow Forest \cup OmriTree(X, 0, l)$
End for
return $Forest$
```

Algorithm 2: $OmriTree(X, e, l, k)$

```
Inputs: $X - input\ data, e - current\ tree\ height, l - height\ limit$
Output: an $OmriTree$
if $e \geq l\ or\ |X| \leq 1$ then
    return $exNode\{size \leftarrow |X|\}$
else
    $taxa \leftarrow sample(X, k)$
    $X' \leftarrow renormalize\_data(X, taxa)$
    $distance\_mat \leftarrow distance\_matrix(X')$
    $P \leftarrow PCoA\_1(X')$
    Randomly select a threshold between $T = (\min\{P\}, \max\{P\})$
    $X_l \leftarrow filter(X', X'[P] < T)$
    $X_r \leftarrow filter(X', X'[P] > T)$
    return $inNode\left\{\begin{array}{c} Left \leftarrow OmriTree(X_l, e+1, l), \\ right \leftarrow OmriTree(X_r, e+1, l), \\ SplitAtt \leftarrow p, \\ SplitValue \leftarrow T \end{array}\right\}$
end if
```

parameters tuning:

a. k:
    1) Try different k's.
    2) try to select the k features according to their relative abundance. i.e. if vector P represents the relative abundance of bacteria across all samples, select the k bacteria according to P.

b. Different distance methods.

Algorithm 3: $depth(x, OF)$

Inputs: $x - \ a\ sample\ , OF - \ \ OmriForest\ (a\ set\ of\ OmriTrees)$.

Output: $d - \ the\ average\ depth\ of\ x\ in\ all\ trees\ in\ OF\ forest$.

```
$d \leftarrow 0$
for tree in OF:
    $d += tree.nodes[x].d$ (according to implementation)
$d /= OF.TreesNum$
return d
```

Algorithm 4: $testASample(A, B, OF)$

Inputs: $A - \ a\ group\ of\ anomaly\ samples\ (A \subset X), B - a\ group\ of\ normal\ samples\ (B \subset X)$, $OF - OmriForest\ (a\ set\ of\ OmriTrees)$.

$* \ A\ \cup B = X, \qquad A \cap B = \emptyset$

Output: True or False

```
$d_a \leftarrow 0$
for anomaly in A:
    $d_a += depth(anomaly, OmriForest)$
$d_a /= len(A)$
$d_b \leftarrow 0$
for normalSample in B:
    $d_b += depth(normalSample, OmriForest)$
$d_b /= len(b)$
if $d_a < d_b$:
    return True
else:
    return False
```