RESEARCH ARTICLE

DCMD: Distance-based classification using mixture distributions on microbiome data

Konstantin Shestopaloff[1☯], Mei Dong[1☯], Fan Gao[1], Wei Xu[1,2]*

1 Dalla Lana School of Public Health, University of Toronto, Toronto, Ontario, CANADA, **2** Princess Margaret Cancer Centre, University Health Network, Toronto, Ontario, CANADA

☯ These authors contributed equally to this work.
* wxu@uhnres.utoronto.ca

Abstract

Current advances in next-generation sequencing techniques have allowed researchers to conduct comprehensive research on the microbiome and human diseases, with recent studies identifying associations between the human microbiome and health outcomes for a number of chronic conditions. However, microbiome data structure, characterized by sparsity and skewness, presents challenges to building effective classifiers. To address this, we present an innovative approach for distance-based classification using mixture distributions (DCMD). The method aims to improve classification performance using microbiome community data, where the predictors are composed of sparse and heterogeneous count data. This approach models the inherent uncertainty in sparse counts by estimating a mixture distribution for the sample data and representing each observation as a distribution, conditional on observed counts and the estimated mixture, which are then used as inputs for distance-based classification. The method is implemented into a *k*-means classification and *k*-nearest neighbours framework. We develop two distance metrics that produce optimal results. The performance of the model is assessed using simulated and human microbiome study data, with results compared against a number of existing machine learning and distance-based classification approaches. The proposed method is competitive when compared to the other machine learning approaches, and shows a clear improvement over commonly used distance-based classifiers, underscoring the importance of modelling sparsity for achieving optimal results. The range of applicability and robustness make the proposed method a viable alternative for classification using sparse microbiome count data. The source code is available at *https://github.com/kshestop/DCMD* for academic use.

Author summary

The uneven performance of conventional distanced-based classifiers when using microbiome profiles to predict disease status has motivated us to develop a novel distance-based method that accounts for uncertainty when modeling sparse counts. We propose a classification algorithm that uses mixture distributions to measure normed distances between microbiome distributions, which better models the underlying structure by handling

Citation: Shestopaloff K, Dong M, Gao F, Xu W (2021) DCMD: Distance-based classification using mixture distributions on microbiome data. PLoS Comput Biol 17(3): e1008799. https://doi.org/10.1371/journal.pcbi.1008799

Editor: Elhanan Borenstein, University of Washington, UNITED STATES

Received: December 10, 2020

Accepted: February 15, 2021

Published: March 12, 2021

Data Availability Statement: The dataset analysed in this study is publicly available in the MLRepo repository. The colorectal cancer: https://knights-lab.github.io/MLRepo/docs/kostic_healthy_tumor.html; and the Crohn's disease: https://knights-lab.github.io/MLRepo/docs/gevers_control_cd_ileum.html.

Funding: W.X. was funded by Natural Sciences and Engineering Research Council of Canada (NSERC Grant RGPIN-2017-06672), Crohn's and Colitis Canada (CCC Grant CCC-GEMIII), and Helmsley Charitable Trust. K.S. was supported by CCC Grant

CCC-GEMIII. M.D. was supported by NSERC Grant RGPIN-2017-06672 and CCC Grant CCC-GEMIII. The funders had no role in study design, data collection and analysis, decision to publish, or preparation of the manuscript.

Competing interests: The authors have declared that no competing interests exist.

excess zeros and sparsity inherent in microbial abundance counts. Applications of DCMD have shown improved classification performance and robustness, making the proposed method an improved alternative for classification using microbiome data.

This is a *PLOS Computational Biology* Methods paper.

Introduction

The increasing accessibility of high-throughput technology has generated a wide array of data types for analysis. One type of data that has recently gained popularity is the microbiome community data, which is composed of site-specific counts for identified bacteria. There is a steadily growing number of studies that have demonstrated associations between the human microbiome and health outcomes, such as inflammatory bowel disease [1], type 2 diabetes [2], and cardiovascular disease [3], making it an important topic of research. However, the presence of sparsity and skewness, which characterizes this type of data, brings a number of challenges to statistical modelling. These challenges have motivated methodological developments that expand the existing algorithms, particularly for classification tasks related to disease risks.

One popular class of approaches often used with microbiome data are distance-based methods, which differentiate and classify samples using distances derived from multivariate measures. Ubiquitous methods include *k*-means [4] and *k*-Nearest Neighbours (*k*-NN) [5], which have been adapted to such data with variable transformations using Euclidean distances, Manhattan distance, and several other measures [6–8]. Other adaptations such as the distance-based nearest shrunken centroid (NSC) classifier, which was developed for the use of microarray data [9]. NSC takes the average of the relative abundances for each class as class centroids [10,11] and then calculate standardized squared distances between new samples and class centroids.

A number of linear and additive machine learning classifiers; such as LASSO, ridge regression (RR), random forest (RF), gradient boosting (GB), and support vector machines (SVM) are also commonly used for high-throughput data [7,11–13]. Some methods rely on penalization (LASSO and RR) in logistic models [14,15], typically with log-transformed adjusted counts or relative abundances of operational taxonomic units (OTUs) to address skewness [16]. The RF and GB algorithms rely on sequentially constructed classifiers and automatically incorporate feature selection [17,18]. Other recent methodological developments for microbiome data include regression models with a phylogenetic tree-guided penalty term [19] and inverse regression to deal with the over-dispersion of zeros in count data [20]. However, the tree-guided method can be overly influenced by tree information [19], and the phylogenetic tree is not always available. The existing methods incorporate observed count data or relative abundance directly when computing distances or defining covariates, with some kinds of transformation of OTUs to account for skewness. None of the methods explicitly account for and model the underlying uncertainty inherent in sparse count data.

This paper aims to address these problems in a classification framework, where predictors are sparse and heterogeneous count data. Shestopaloff et al. [21] proposed representing count data using a mixture distribution to analyze the differences between microbiome communities. We extend the method to distance-based classification using mixture distributions (DCMD) that specifically addresses the uncertainty in sparse and low-count data. DCMD measures the distance between the sample-specific distributions of OTUs rather than between counts or relative abundances, which better models the structure of microbiome data for the distance measure. DCMD is also able to handle excess zero counts, which can potentially improve the predictive accuracy when using sparse OTUs. In this paper, we use two simulation studies to show the advantage of DCMD for classification over existing distance metrics and compare it against common machine learning methods. We provide a comprehensive comparison of distance-based classification methods (k-means, k-NN, and NSC) and machine learning methods (RF, GB, LASSO, RR, and SVM) in different simulation settings, which to our knowledge has not been studied before. We also illustrate the effectiveness of DCMD on two human microbiome studies [22,23]. The paper concludes with a discussion of the merits, drawbacks and the scope of applicability of the proposed methodology.

Method

In this section we outline the framework of DCMD. The main steps of the model include mixture distribution specification and parameter estimation for modelling observed data, calculation of conditional distributions for each sample, and calculating distances between samples and cluster centres to use in distance-based classification methods. The mixture model and conditional distribution estimation are described in Shestopaloff et al. [21]. It is proposed to model the underlying population rate structure of the observed count data using a mixture distribution with Poisson-Gamma components, then conditioning on observed sample counts and resolution to obtain sample-specific distributions. In the next step, we use the sample-specific distributions for classification by calculating the distances between distributions.

Model specification and estimation

Microbiome data typically consists of OTU counts, as illustrated in Table 1. The notations used in our method formulation are as follows:

n_{ij}, $i = 1,\ldots,I$ for $j = 1,\ldots,J$, the count of the jth OTU of the ith sample.

N_i, the total number of aligned reads of sample i, $N_i = \sum_j n_{ij}$

Without loss of generality, we focus on a specific OTU and omit the jth subscript for subsequent notation. Assume that the observed counts, n_i, are Poisson distributed with rate $r_i = q_i N_i$, $i = 1,\ldots,I$ for sample i, where q_i is the individual-specific relative abundance and is sampled from some general OTU relative abundance distribution G_q. Then we have,

$$r_i = q_i N_i = r_i^* t_i,$$

where $t_i = N_i/\bar{N}$, $r_i^* = q_i\bar{N}$, and $\bar{N} = \sum_i N_i/I$, with r_i^* sampled from $G = G_q\bar{N}$, which is the rate normalized to the average sample reads to make sure that the counts are treated on the

Table 1. An OTU table for microbiome data.

	OTU 1	...	OTU J	Total Reads
Sample 1	n_{11}	...	n_{1j}	N_1
Sample I	n_{I1}	...	n_{IJ}	N_I

https://doi.org/10.1371/journal.pcbi.1008799.t001 same scale. Thus, the observed count for a specific site of OTU is

$$n_i | t_i, r_i^* \sim \text{Poisson}(r_i^* t_i),$$

$$r_i^* \sim G.$$

Since the distribution of OTU is zero-inflated, skewed, and heavy tailed, we propose a mixture distribution to approximate G. For positive rates on a given interval, we specify a set of Gamma components, $\Gamma(\alpha, \beta)$, with shape α and rate β, to cover the range of the data. To separate structural zeros from low-rate and undetected samples, we include a zero-point mass, $n_i | t_i, r_i^* \sim 0$, where $P(n_i = 0) = 1$. Additionally, for sparse high rates, we define a high-count point mass, $n_i | t_i, r_i^* \sim C \cdot \mathbf{1}(n_i > C)$, where $P(n_i > C) = 1$, C is the truncation point and $\mathbf{1}(\cdot)$ is the indicator function. The full set of mixture components is $\Omega = (G_z, G_1, G_2, \ldots, G_M, G_{C+})$ where G_z is a zero-point mass, G_m, $m = 1,2,\ldots,M$, is a set of Gammas components $\Gamma(\alpha_m, \beta_m)$, and G_{C+} is a high-count point mass. The process for defining the mixture model components is described in detail in the Simulation section.

Define the weight of each component as

$$\boldsymbol{w} = (w_z, w_1, w_2, \ldots, w_M, w_{C+})',$$

where w_z is the weight of the zero-point mass, w_m, $m = 1,2,\ldots,M$, is the weight for mth corresponding Gamma component, and w_{C+} is the weight of high-count point mass. Define

$$y_x = \sum\nolimits_i \mathbf{1}(n_i = x),$$

the number of species observed x times across all samples for $x = z, 0,1,2,\ldots,C,C+$. Then our goal is to minimize $\sum_{x=0}^{C+} [y_x - y_x^E]^2$, where y_x^E is the expected aggregate counts of y_x. Note that given $\Gamma(\alpha_m, \beta_m)$, sample counts conditional on t_i are distributed as a negative binomial $NB[\alpha_m, \beta_m/(t_i+\beta_m)]$ [21]. Define

$$p_{xmi} = P_{NB}(X = x | t_i, \alpha_m, \beta_m)$$

as the probability of observing count x from the mth mixture component conditional on the resolution t_i. Then $y_x^E = \sum_{w_m \in \boldsymbol{w}} w_m p_{xm} \cdot I$, where $p_{xm} = \Sigma_i p_{xmi}/I$. Thus, we have the objective function:

$$arg \underset{\overrightarrow{w}}{min} \sum\nolimits_{x=z}^{C+} [y_x - (\sum\nolimits_{w_m \in \boldsymbol{w}} w_m p_{xm}), I]^2, \tag{1}$$

$$s.t. \sum\nolimits_m w_m = 1, w_m \geq 0, \forall m.$$

The estimate, $\hat{\boldsymbol{w}}$, is obtained by optimizing the least-squares objective function (1), using the Broyden-Fletcher-Goldfarb-Shanno (BFGS) algorithm [24] with the augmented Lagrangian method [25] for the constraints.

Due to the sparse nature of the data, we only optimize the weights and fix the Gamma parameters. Attempting to model the low-rate structure by optimizing both weights and Gamma parameters (α_m, β_m) via expectation-maximization (EM) results in biased structural zero estimates and a poor overall fit of the low counts [26]. In this context, the EM is also prone to numerical issues, convergence to local minima and can often be too slow computationally for this type of application [26]. On the other hand, BFGS provides a much faster and robust alternative.

Weighted mixture distribution

To address the uncertainty around specifying components for the mixture model, particularly for the low rates where sparsity is often an issue, we define a set of nested models Φ_l, $l = 1,\ldots, L$, with varying components for modelling the rate structure around zero. We estimate the joint mixture model using a nonparametric bootstrap algorithm. As stated in Shestopaloff et al. [21], we can obtain the weight $v(l)$ of each candidate model, which is the proportion of times each model is selected as optimal relative to the observed data, and calculate the weights for the joint mixture distribution. Let $\boldsymbol{w}_l$ be the estimated weights for each candidate model, Φ_l, with zeros assigned to the weights of components not included in a specific model, then the weights of the joint model are $\boldsymbol{w} = \sum_l v(l)\boldsymbol{w}_l$.

Sample-specific distribution

Once we have a distribution for the OTU, we can estimate sample-specific distributions by conditioning on the observed count n_i, estimated mixture weights $\boldsymbol{w}$, and resolution t_i. We can obtain the probability that sample i sampled from a specific component, as:

$$p_{im} = P(i \in G_m | n_i, t_i, \boldsymbol{w}) = w_{G_m} \frac{\Gamma(n_i + \alpha_m)}{\Gamma(n_i + 1)\Gamma(\alpha_m)} \left(\frac{\beta_m}{t_i + \beta_m}\right)^{\alpha_m} \left(1 - \frac{\beta_m}{t_i + \beta_m}\right)^{n_i}. \quad (2)$$

The probability of being assigned to the zero-point mass is $P(i \in G_0) = \mathbf{1}(n_i = 0)$ and to the high-count point mass is $P(i \in G_{C+}) = \mathbf{1}(n_i > C)$. Define the sample-specific mixture weights as

$$\boldsymbol{w}_i = (w_{iz}, w_{i1}, \ldots, w_{iC}, w_{iC+})',$$

where

$$w_{im} = P(i \in G_m) / \textstyle\sum_m P(i \in G_m) = p_{im} / \textstyle\sum_m p_{im}.$$

Since the sample-specific weights have been adjusted for the individual resolutions t_i through the p_{im} probabilities, the Poisson-Gamma mixture probabilities are $NB(\alpha, \beta/(1+\beta))$. Also note that we have differentiated the zeros in our mixture distribution, which are defined as structural zeros, $x = z$, and observed zeros, $x = 0$. Given the underlying rate distribution from the joint mixture model, we can then calculate the probability of observing count $x = z$, $0,1,\ldots,C$, $C+$ from each mixture component G_m as

$$P_{G_m}(x) = P(X = x | G_m) = P_{NB}(X = x | \alpha_m, \beta_m).$$

For the point masses we have $P(X = x | G_z) = \mathbf{1}(n_i = 0)$ and $P(X = x | G_{C+}) = \mathbf{1}(n_i > C)$, respectively. To simplify the representation of the distribution, define a vector of probabilities

$$\boldsymbol{P}(\boldsymbol{x}) = [P(X = x | G_z), P(X = x | G_1), \ldots, P(X = x | G_m), P(X = x | G_{C+})]$$

$$= [P_{G_z}(x), P_{G_1}(x), \ldots, P_{G_M}(x), P_{G_{C+}}(x)],$$

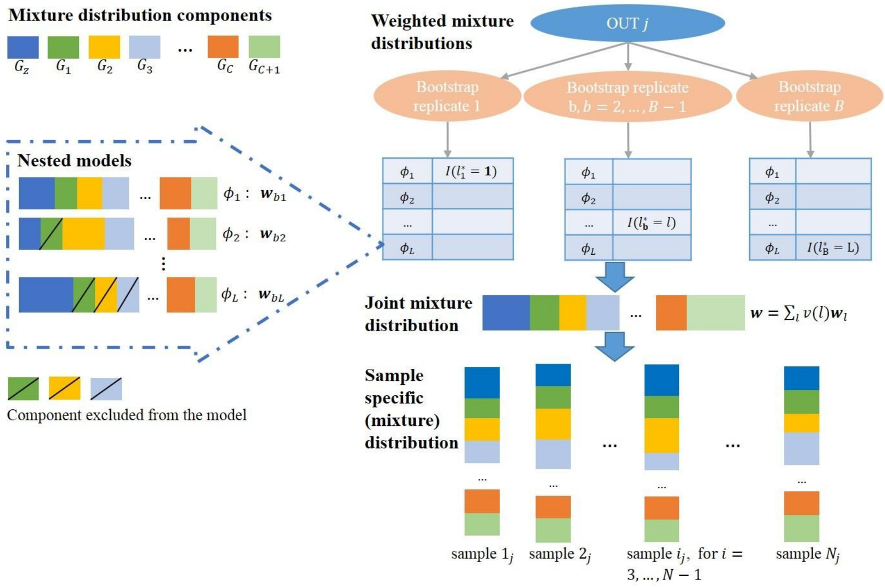
for $x = z$, $0,1,\ldots,C$, $C+$. Then we can define the discrete probability density for sample i as

$$\boldsymbol{P}_i = [P_i(z), P_i(0), \ldots, P_i(C), P_i(C+)] = \boldsymbol{w}_i{}'\boldsymbol{P},$$

where

$$P_i(z) = w_{G_z},$$

$$P_i(C+) = 1 - [\textstyle\sum_{x=0}^{C} P_i(x) + P_i(z)],$$


Mixture distribution components
G_z
G_1
G_2
G_3
...
G_C
G_{C+1}
Nested models
ϕ_1 : $\boldsymbol{w}_{b1}$
ϕ_2 : $\boldsymbol{w}_{b2}$
ϕ_L : $\boldsymbol{w}_{bL}$
Component excluded from the model
Weighted mixture distributions
OUT j
Bootstrap replicate 1
Bootstrap replicate b, $b = 2, \ldots, B-1$
Bootstrap replicate B
ϕ_1
$I(l_1^* = \mathbf{1})$
ϕ_2
...
ϕ_L
$I(l_b^* = l)$
$I(l_B^* = \mathrm{L})$
Joint mixture distribution
$\boldsymbol{w} = \sum_l v(l)\boldsymbol{w}_l$
Sample specific (mixture) distribution
sample 1_j
sample 2_j
sample i_j, for $i = 3, \ldots, N-1$
sample N_j

Notation: l_b^* is optimal model for replicate b; $v(l) = \sum_b I(l_b^* = l)$, l=1, ..., L.

Fig 1. Workflow for obtaining a sample-specific mixture distribution for each sample i in OTU j: 1) Specify a set of nested candidate mixture distributions using a specific set of components; 2) Apply bootstrap to the set of nested models and calculate the weights of each candidate mixture model, then calculate the weights of the joint mixture distribution; 3) Estimate sample-specific distributions conditional on n_i, t_i, and the joint mixture distribution.

https://doi.org/10.1371/journal.pcbi.1008799.g001

$$\boldsymbol{P} = [\boldsymbol{P}(z), \boldsymbol{P}(0), \ldots, \boldsymbol{P}(C), \boldsymbol{P}(C+)].$$

The $\boldsymbol{P}(x)$ vectors in the matrix $\boldsymbol{P}$ are the vectors giving the probability of observing x from each mixture component, which can be pre-calculated for distance calculations. The overview of how to obtain sample-specific mixture distributions given a set of mixture distribution components is shown in Fig 1.

Classification

Once the distribution for each sample has been computed, we use k-means and k-Nearest Neighbours (k-NN) algorithms for classification. In this section, we outline how to apply these algorithms using two distance measures, discrete L^2 (D-L^2) norm and continuous cumulative L^2 (CC-L^2) norm.

Distance measures. Given posterior probability f_i, cumulative posterior probability F_i, and an estimated set of weights w_i for sample i, the distance metrics are:

D-L^2 Norm:

$$\delta_{D-L^2}(f_i, f_j) = \sum_x [f_i(x) - f_j(x)]^2$$

$$= \sum_x [(\boldsymbol{w}_i - \boldsymbol{w}_j)\boldsymbol{P}(x)]^2, \tag{3}$$

where $x = z, 0, 1, \ldots, C, C+$. Note that we include the structural zero component, z, separately and that the distances only depend on the weights. For multiple predictors, j = 1, . . ., J, the total distance between samples i_1 and i_2 is the sum across all predictors, $D(i_1, i_2) = \sum_j \delta_j(f_{i_1}, f_{i_2})$.

CC-$\boldsymbol{L^2}$ Norm:

$$\delta_{CC-L^2}(F_i, F_j) = \int_0^C [F_i(x) - F_j(x)]^2 dx$$

$$= (\boldsymbol{w}_i - \boldsymbol{w}_j)\boldsymbol{G}_{m_1 m_2}(\boldsymbol{w}_i - \boldsymbol{w}_j)', \qquad (4)$$

where $\boldsymbol{G}_{m_1 m_2}$ is a matrix with the (m_1, m_2) entry set to $\int G_{m_1}(x)G_{m_2}(x)dx$ for each of the continuous mixture component. Details of the derivation can be found in Shestopaloff [26].

Distance-based classification. We use the distances calculated in Eqs (3) and (4) in a *k*-means and *k*-NN framework. In *k*-means, the mean of each class is calculated from the training data and points are classified to the nearest class. In *k*-NN, samples are classified as the mode of the labels from *k* closest neighbours of the training set. The steps of *k*-means and *k*-NN algorithms are as follows:

***K*-means**: To adapt the *k*-means algorithm, we estimate the mean distribution for each class by minimizing the distributional distances between it and the class samples, conditional on a specified distance. Since distances are L^2 norms and only depend on the weights, as shown in Eqs (3) and (4), the mean of the weights for each class gives the optimum. The algorithm is implemented as follows:

Step 1: Determine the mean of the weights for the *j*th predictor in class *k*,

$\boldsymbol{w}_{\boldsymbol{\mu_k},\boldsymbol{j}} = \sum_{i \in k,j} \boldsymbol{w}_{\boldsymbol{k},\boldsymbol{j}} / |N_{k,j}|$, where $|N_{k,j}|$ is the number of samples in class *k* of predictor j, $k = 1, \ldots, K$ and $j = 1, \ldots, J$;

Step 2: Compute the distance to the mean for sample *i* across all predictors,

$$D(i, \mu_k) = \sum_j \delta(\boldsymbol{P}_{\boldsymbol{i},\boldsymbol{j}}, \boldsymbol{P}_{\boldsymbol{\mu_k},\boldsymbol{j}});$$

Step 3: Predict the label of sample *i* as the closest mean,

$$\hat{y}_i = \underset{k}{argmin}\ D(i, \mu_k).$$

***K*-NN:** After computing the pairwise distances between samples and summing across predictors, these can be used directly to identify the nearest neighbours for classification. The algorithm for *k*-NN is as follows:

Step 1: Compute the pairwise distance of sample i_1 and i_2, $i_1, i_2 = 1, \ldots, I$, $i_1 \neq i_2$,

$$D(i_1, i_2) = \sum_j \delta(\boldsymbol{P}_{\boldsymbol{i_1},\boldsymbol{j}}, \boldsymbol{P}_{\boldsymbol{i_2},\boldsymbol{j}});$$

Step 2: For sample i, pick the k samples with smallest distance to sample *i*, the optimal *k* can be determined using cross-validation (CV) in the training set or existing heuristics.

Step 3: Tally the labels of the k nearest neighbours, then sample i is predicted as the mode of the *k* labels.

The overall workflow of DCMD within the *k*-means and *k*-NN frameworks is presented in Fig 2.

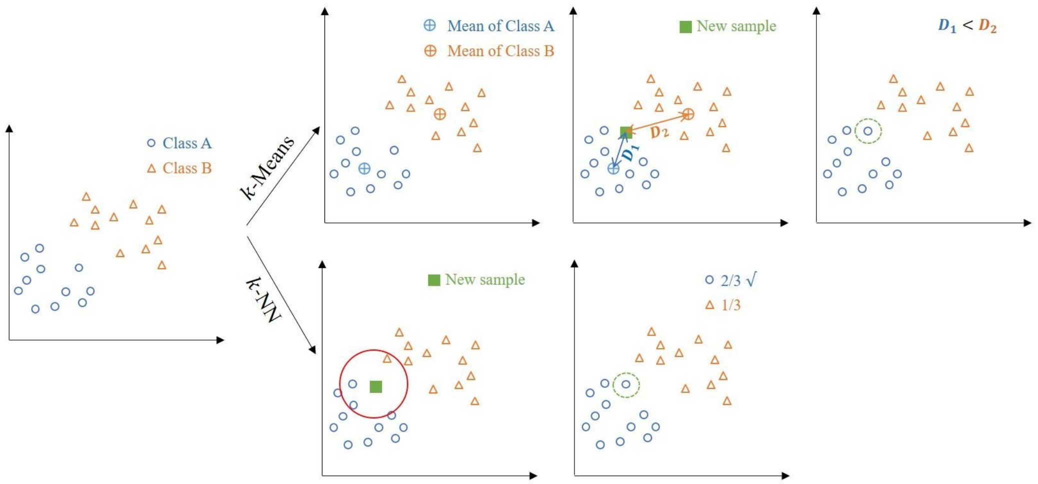


Fig 2. Illustration of k-means and k-NN framework using sample-specific distributions. For k-means (top panel), the distance between the new sample to the mean of class A is smaller than to the mean of class B, hence the new sample is predicted as class A. For k-NN (bottom panel), using 3 nearest neighbours, the new sample is predicted as class A.

https://doi.org/10.1371/journal.pcbi.1008799.g002

Predictive metrics

Let $\hat{y}_i$ be the predicted class for sample *i*. The classification accuracy is defined as the proportion of correctly predicted cases: Accuracy $= \frac{1}{I}\sum_{i=1}^{I} I(\hat{y}_i = y_i)$. For binary outcomes we also include precision, recall, and F1 score as metrics to measure predictive performance. We count the number of true positive (*TP*), false positive (*FP*), and false negative (*FN*) and defined these metrics as follows [27]:

$$\text{Precision} = TP/(TP + FP),$$

$$\text{Recall} = TP/(TP + FN),$$

$$\text{F1 Score} = 2 \times \text{precision} \times \text{recall}/(\text{precision} + \text{recall}).$$

Simulation

Data generation

To evaluate the performance of the DCMD method, we design simulation studies that mimic microbiome community count data and assess classification performance. We simulate a separate mixture distribution for each class, individual sample rates, and resolutions to generate observed counts. For the mixture distribution, the number of components, M, is sampled from *Unif*(5, 15). And the number of samples to be taken from each mixture component is set by binning samples from a *Beta*(α_b, β_b) at uniform intervals, with the α_b varied to give different class means and levels of sparsity and with β_b~*Unif*(2, 6.5) to control dispersion. The observed counts for each sample are then generated as $n_i \sim Poisson(r_i^* t_i)$, where r_i^* is the sampled rate and resolution t_i~*Unif*(2/3, 5/4).

We consider two- and three-class outcomes with several simulation scenarios for each case. For the two-class outcome, parameter settings and summary statistics for each scenario are shown in Table 2. Scenarios 1 and 2 have low sparsity data, and scenarios 3 and 4 are highly sparse. Scenarios 1 and 3 have weakly differentiated classes (small difference in α_b), while scenarios 2 and 4 have strongly differentiated classes (large difference in α_b). The sample size is

Table 2. Two-class outcome: the parameter settings for each scenario and the corresponding summary statistics of each class over 100 replicates.

Scenario	Signal	Sparsity	Class	Size	α_b range	Mean ZP (SD)	Mean
1	Weak	Low	1	400	(1.5, 1.8)	0.32 (0.13)	7.91
			2	400	(1.8, 2.1)	0.26 (0.12)	9.63
2	Strong	Low	1	400	(1.5, 1.8)	0.33 (0.13)	7.74
			2	400	(2.7, 3.0)	0.14 (0.09)	15.12
3	Weak	High	1	400	(0.2, 0.4)	0.84 (0.07)	1.05
			2	400	(0.4, 0.6)	0.74 (0.10)	1.92
4	Strong	High	1	400	(0.2, 0.4)	0.84 (0.07)	0.98
			2	400	(0.8, 1.0)	0.56 (0.13)	3.61

https://doi.org/10.1371/journal.pcbi.1008799.t002

$I = 800$, with 400 samples per class and $J = 25$ OTUs. For the three-class outcome, parameter settings and summary statistics are shown in S1 Table. Scenarios 1–3 have strongly differentiated classes, with varying levels of sparsity. The sample size is $I = 1200$, with 400 samples in each class and $J = 25$ OTUs. A null case scenario is also generated by permuting class labels, and performance metrics for each outcome and scenario are computed over 100 simulation replicates.

Mixture model specification

The specification of the mixture model components should be data-driven, and the main requirement is that the Gammas allow for appropriate coverage of the observed data. We split the count data into five intervals and apply different strategies to specify components on each interval. Modelling of the zeros and low-rate structures is based on [28], and modelling of the higher counts is based on [21].

1. Structural zeros: For data with observed zeros, a zero-point mass $P(X = 0) = 1$ is included to model zero inflation.
2. Low counts ($x \in [0,1,2,3]$): We specify components as Poisson rate posteriors with uniform priors for each of the counts, which is $\Gamma(x+1, 1)$. Hence, we include $\Gamma(1,1)$, $\Gamma(2,1)$, $\Gamma(3,1)$ and $\Gamma(4,1)$. The cut-off is set to $x = 3$ because rate posteriors for higher values have a low probability of observing zero, and we want to differentiate the distributions relevant to modelling zero inflation. We also want to examine whether more mass close to zero improves the fit for the low rates. Therefore, we add exponentials with a higher rate, β, to have more mass near zero. In this case, we include $\Gamma(1,2)$ into the candidate models. We can potentially include $\Gamma(1,3)$ or other terms into the model, then apply the procedure described above to select the optimal mix. A fuller discussion, drawn from modelling sparse counts for total species estimation, can be found in [28].
3. Integer counts ($x \in [4,5,6,7]$): Components in this range are also specified as the rate posterior $\Gamma(x+1, 1)$ at integer intervals. This block exists as a buffer to ensure no gaps in the coverage after the low-count distributions, as this can potentially bias the structural zero and low-rate estimates. This is specified until the last integer component has little overlap with the previous low-rate component. In our formulation, we use an upper limit of $x = 7$ as simulations showed negligible differences between $x = 7$ and $x = 8$. The integer components include $\Gamma(4,1),\ldots,\Gamma(7,1)$.
4. High counts ($x \in [8,\ldots,C]$): The higher counts tend to have a large range, and it's not practical to specify them on integer intervals. In this case, we set the number of components based on the range of the data and specify α_m at uniform intervals on a linear-log scale from

8 to $C = q_p$, a set quantile of the data. Using between 10 and 15 components worked well in past applications [21]. For our modelling, p = 0.85 is an effective threshold, which means C is the 85% quantile of the sample.

5. Extreme high counts ($x>$C): These counts are truncated to a point mass $P(X>C) = 1$, in part because of the low density in this range and the uncertainty in modelling them and in part to decrease computation time. The mixture model specification heuristics described above are primarily for modelling low-abundance OTU, which covers the most information of the microbiome data. Higher abundance OTU can be modelled by restricting the component specification to higher counts and increasing p.

The full model we use for our data includes [Γ(1,2), Γ(1,1),. . .,Γ(7,1), Γ(8,1)], along with varying high-count components. Nested models are generated by progressively excluding Γ (1,2), [Γ(1,2), Γ(1,1)],. . . for a total of five models. A sample model specification for one of the OTU is presented in S2 Table.

Model fitting and comparison methods

The proposed method is compared with *k*-means and *k*-NN using Euclidean and Manhattan distances of relative abundances, distance-based NSC, as well as LASSO, RF, GB, RF and SVM classifiers [29]. Models are trained using a 60/40 training and test set split [30], with the training set remaining the same for all classifiers within each replicate. For the machine learning methods, we use existing packages and tune the hyper-parameters using cross-validation when appropriate, see details in S1 Text.

Simulation results

For the two-class outcome, classification accuracy for each model and scenario is presented in Fig 3. The orange boxplots are the results for the proposed DCMD method in a *k*-means and *k*-NN framework. The blue boxplots are the other distance-based methods, including *k*-means and *k*-NN with Euclidean and Manhattan distance and NSC. The green boxplots give results for the machine learning methods, including RF, GB, LASSO, RR, and SVM. The dashed red line gives the average accuracy of the best method in each scenario. The results show that in Scenarios 1 and 2, when sparsity is low, *k*-means with $CC\text{-}L^2$ norm performs best, followed by *k*-means with D-L^2 norm, while in Scenarios 3 and 4, when sparsity is high, *k*-means with D-L^2 norm gives the best performance, followed by *k*-means with $CC\text{-}L^2$ norm. Overall, DCMD in a *k*-means framework with L^2 norms outperforms the other classification methods for all types of signals and data structures for the two-class outcome. Differences in accuracy within the distance-based methods are also progressively more pronounced in favour of DCMD, among which *k*-means outperforming *k*-NN. The specialized NSC approach performed similarly to DCMD within the *k*-NN framework. However, NSC generally falls short of *k*-means DCMD and other machine learning methods.

Table 3 shows the summary statistics of the F1 Score over 100 replicates for the two-class outcome. The top results in each scenario are highlighted. Similar to accuracy, DCMD with L^2 norms produce the highest F1 Scores (F1 Score (SD) = 0.68 (0.034), 0.92 (0.017), 0.77 (0.028), 0.95 (0.014) in Scenarios 1–4, respectively), which are better than the best machine learning method (GB: 0.64 (0.033), RF and RR: 0.89 (0.019), LASSO: 0.75 (0.030), GB: 0.94 (0.016) in Scenarios 1–4, respectively). DCMD shows consistent good performance in each scenario compared among the methods.

The classification accuracy of each model and scenario for a three-class outcome is presented in Fig 4. For Scenarios 1–3, the classes are differentiated under varying levels of sparsity,

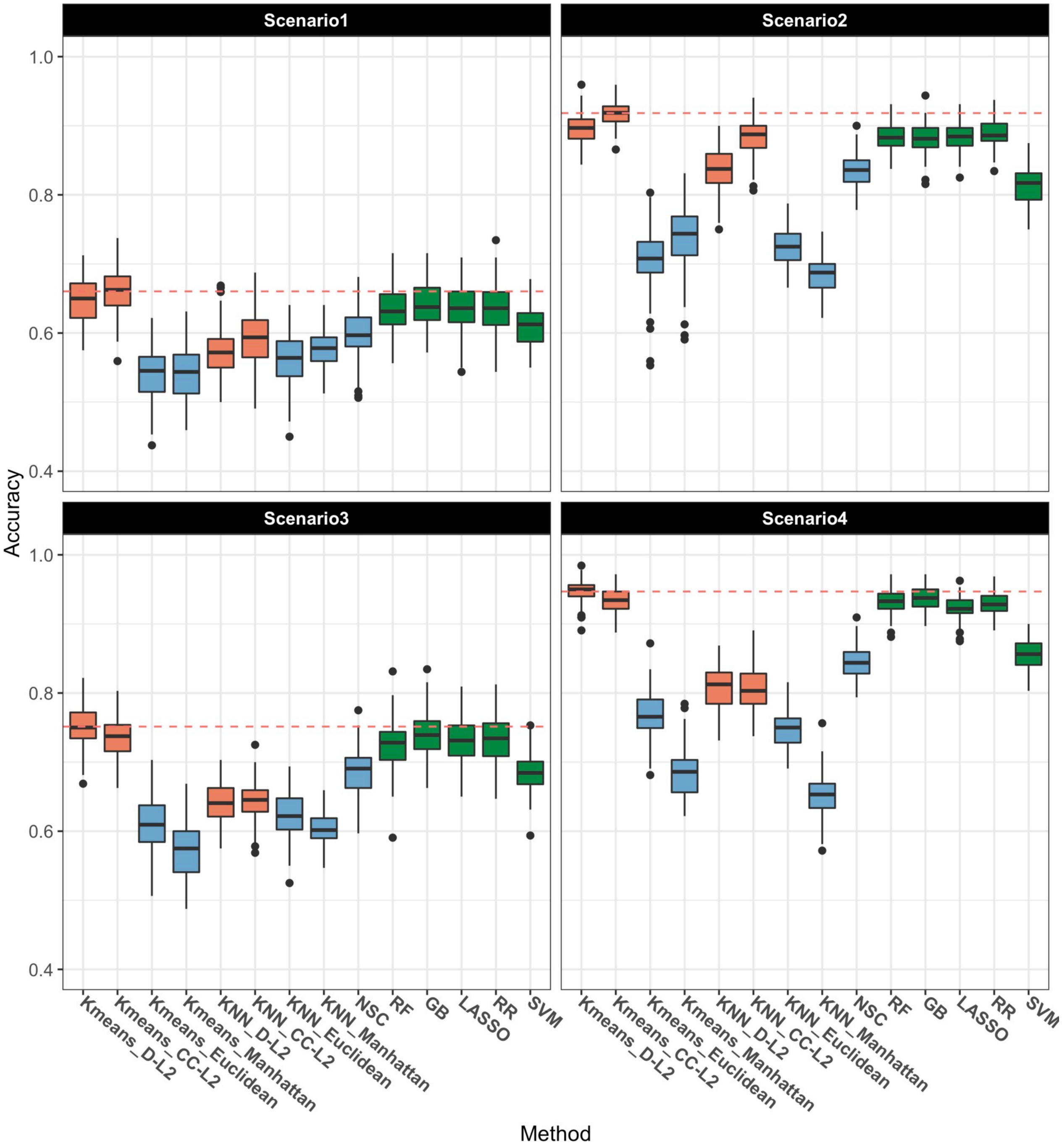


Fig 3. Two-class outcome: boxplot of the accuracy over 100 replicates for each method and scenario. The proposed DCMD method is shown in orange for *k*-means and *k*-NN with D-L^2 and CC-L^2 distances. The other distance-based methods are shown in blue, including *k*-means and *k*-NN with Euclidean and Manhattan distances and NSC. Machine learning methods are shown in green, including random forest (RF), gradient boosting (GB), LASSO, ridge regression (RR), support vector machine (SVM). The dashed red line gives the average accuracy of the best method in each scenario.

https://doi.org/10.1371/journal.pcbi.1008799.g003

and we observe that DCMD is competitive with the optimal machine learning methods. Although RR has similar predictive accuracy to DCMD in Scenario 1 and 3, and GB has similar predictive accuracy in Scenario 2, DCMD is consistently improved over the optimal

Table 3. Two-class outcome: the summary of F1 Scores for each model over 100 replicates.

Model	Scenario 1 (SD)	Scenario 2 (SD)	Scenario 3 (SD)	Scenario 4 (SD)
k-means-D-L^2	0.67 (0.033)	0.90 (0.019)	**0.77 (0.028)**	**0.95 (0.014)**
k-means-CC-L^2	**0.68 (0.034)**	**0.92 (0.017)**	0.76 (0.028)	0.94 (0.016)
k-means-Euclidean	0.56 (0.054)	0.73 (0.047)	0.66 (0.039)	0.79 (0.026)
k-means-Manhattan	0.58 (0.065)	0.78 (0.040)	0.69 (0.030)	0.77 (0.022)
k-NN-D-L^2	0.59 (0.050)	0.85 (0.028)	0.74 (0.022)	0.85 (0.021)
k-NN-CC-L^2	0.59 (0.048)	0.88 (0.028)	0.73 (0.025)	0.85 (0.020)
k-NN-Euclidean	0.60 (0.043)	0.77 (0.023)	0.68 (0.031)	0.79 (0.020)
k-NN-Manhattan	0.66 (0.037)	0.77 (0.015)	0.71 (0.017)	0.75 (0.016)
NSC	0.56 (0.067)	0.85 (0.023)	0.69 (0.046)	0.86 (0.022)
RF	0.63 (0.036)	**0.89 (0.019)**	0.74 (0.038)	0.94 (0.017)
GB	**0.64 (0.033)**	0.89 (0.020)	0.75 (0.033)	**0.94 (0.016)**
LASSO	0.64 (0.039)	0.89 (0.021)	**0.75 (0.030)**	0.93 (0.015)
RR	0.62 (0.052)	**0.89 (0.019)**	0.74 (0.037)	0.93 (0.015)
SVM	0.64 (0.031)	0.82 (0.028)	0.72 (0.026)	0.86 (0.021)

https://doi.org/10.1371/journal.pcbi.1008799.t003

comparison method. None of the models is systematically over-fit, as predictive accuracy in the null case (Scenario 4) is near the baseline accuracy of 0.33.

Application

Data description

We test our method on data from two microbiome studies. The first is a study on colorectal cancer reported by [22]. A total of 190 samples (95 pairs) were collected from 95 patients in Vall d'Hebron University Hospital in Barcelona and Genomics Collaborative. The study aimed to identify associations between tumor microbiome and colorectal carcinoma. Both the colorectal adenocarcinoma tissue and adjacent non-affected tissues were collected. The OTU count table generated by 16S amplification was obtained from the Microbiome Learning Repo [12]. Prior to model training, eighteen samples with total reads less than 100 were dropped from the dataset, and we also excluded OTUs with mean relative abundance less than 0.001, resulting in 149 OTUs and 172 samples (86 pairs) used to differentiate tumour and normal tissue. The second study is a case-control study of Crohn's disease (CD) from a multi-center cohort, which was designed to examine how microbiota contributes to CD pathogenesis [23]. The profiles were obtained using Illumina 16S rRNA sequencing. The dataset was downloaded from the Microbiome Learning Repo [12] and consisted of 140 ileal tissue biopsy samples. Minimal sample depth is set at 100, and OTUs are restricted to less than 90% zero proportion, leaving 140 samples (78 cases and 62 controls) and 31 OTUs for analysis.

Model fitting and evaluation

For both datasets, we compare our proposed L^2-norm based k-means and k-NN classifier with five other distance-based classifiers (k-means-Euclidean, k-means-Manhattan, k-NN-Euclidean, k-NN-Manhattan, NSC) and six machine learning methods (RF, GB, LASSO, RR, SVM). We assess model performance using 10-fold CV. In each iteration, one fold of the data is treated as the test set, and the remaining nine are used for training. The specification of DCMD and other classifiers is the same as that in the simulations (see S1 Text). We calculate accuracy, precision, recall, and F1 score as metrics for comparison.

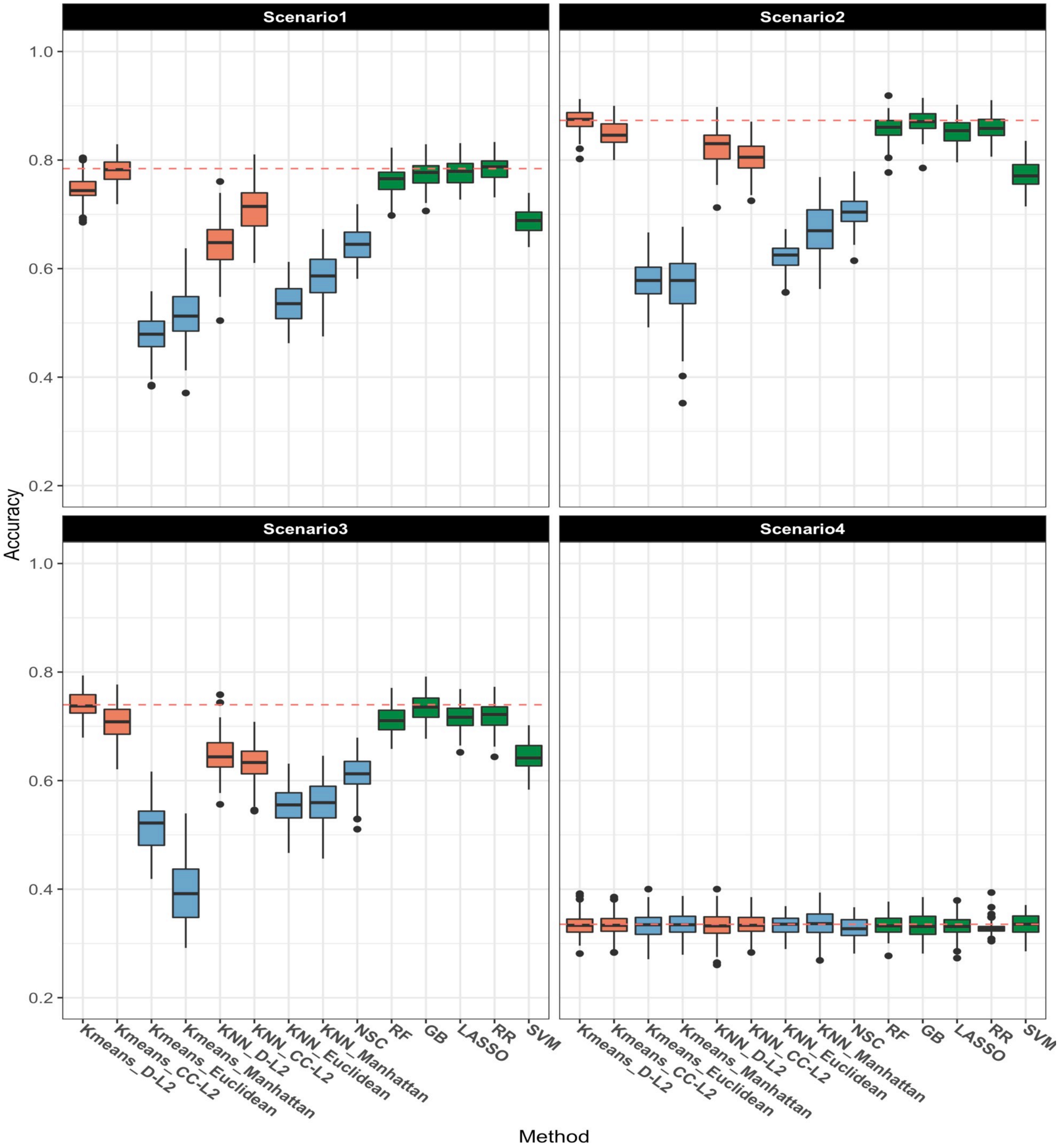


Fig 4. Three-class outcome: boxplot of accuracy over 100 replicates for each method and scenario. The proposed DCMD method is shown in orange for k-means and k-NN with D-L^2 and CC-L^2 distances. The other distance-based methods are shown in blue, including k-means and k-NN with Euclidean and Manhattan distances and NSC. Machine learning methods are shown in green, including random forest (RF), gradient boosting (GB), LASSO, ridge regression (RR), support vector machine (SVM). The dashed red line gives the average accuracy of the best method in each scenario.

https://doi.org/10.1371/journal.pcbi.1008799.g004

For the colorectal cancer data, we reduce the predictor space for distance-based classifiers by the univariate screening of OTUs with a nonparametric Mann-Whitney U test on the training set. To adjust for multiple comparisons, we use q-values obtained by the Benjamini–

Table 4. Dataset 1—Colorectal Cancer: the predictive performance of the 14 classifiers.

Method	Accuracy	Precision	Recall	F1 score
k-means-*D*-L^2	**0.67**	0.66	**0.69**	**0.67**
k-means-*CC*-L^2	0.63	0.62	0.67	0.65
k-means-Euclidean	0.62	0.62	0.60	0.61
k-means-Manhattan	0.65	0.66	0.60	0.63
k-NN-*D*-L^2	0.65	**0.77**	0.43	0.55
k-NN-*CC*-L^2	0.63	0.65	0.57	0.61
k-NN-Euclidean	0.63	0.66	0.55	0.60
k-NN-Manhattan	0.61	0.69	0.40	0.50
NSC	**0.67**	**0.72**	0.56	0.63
RF	0.60	0.61	0.59	0.60
GB	0.63	0.63	0.64	0.64
LASSO	0.59	0.59	0.59	0.59
RR	0.64	0.63	**0.69**	**0.66**
SVM	0.66	0.67	0.63	0.65

https://doi.org/10.1371/journal.pcbi.1008799.t004

Hochberg (BH) method [31] and retain OTUs with q-values less than 0.05 in each training set. The mean number of OTUs selected from each training set is 42 (range: 13–57). For the machine learning approaches, we include all 149 OTUs. For the CD dataset, 31 OTUs are included for all methods.

Applied results

The predictive performance of each classifier for the colorectal cancer and CD studies are presented in Tables 4 and 5, respectively. The accuracy of *k*-means with *D*-L^2 norm is 0.67 for colorectal cancer and 0.73 for CD, which is the best method for colorectal cancer and the second-best for CD. The F1 scores are also among the highest for both datasets, indicating that DCMD has consistently optimal performance and an improvement over the other classifiers. Predictive accuracy of *k*-means with *CC*-L^2 norm is slightly worse, likely due to high zero proportions in the predictors, which is consistent with the simulation results. Similarly, DCMD

Table 5. Dataset 2—Crohn's Disease: the predictive performance of the 14 classifiers.

Method	Accuracy	Precision	Recall	F1 score
k-means-*D*-L^2	**0.73**	**0.75**	0.78	**0.76**
k-means-*CC*-L^2	0.72	**0.75**	0.74	0.75
k-means-Euclidean	0.68	0.69	0.76	0.72
k-means-Manhattan	0.66	0.67	0.76	0.71
k-NN-*D*-L^2	0.62	0.62	**0.85**	0.71
k-NN-*CC*-L^2	0.61	0.61	0.83	0.71
k-NN-Euclidean	0.65	0.66	0.78	0.71
k-NN-Manhattan	0.61	0.62	0.77	0.69
NSC	0.66	0.64	**0.90**	0.74
RF	0.69	0.71	0.76	0.73
GB	0.68	0.69	0.76	0.72
LASSO	**0.74**	**0.76**	0.78	0.73
RR	0.69	0.69	0.78	0.73
SVM	**0.74**	0.74	0.81	**0.77**

https://doi.org/10.1371/journal.pcbi.1008799.t005

outperforms Euclidean and Manhattan distances within *k*-means, and *k*-means outperforms *k*-NN overall. Within *k*-NN, accuracy and F1 score indicate that DCMD has predictive performance comparable to Euclidean or Manhattan distances. The NSC approach has an accuracy of 0.67 and a precision of 0.72 for colorectal cancer, with a recall of 0.56 and an F1 score of 0.63, notably lower than that of the *k*-means classifiers. The performance is unstable in the CD data.

Compared to the machine learning methods, DCMD with *k*-means is superior to RF, GB, LASSO, RR, and SVM in the first dataset. When results are replicated controlling for distance-based classifier variable selection (S3 Table), machine learning methods has improved performance, except for GB. In the second dataset, LASSO and SVM are the best methods with accuracies of 0.74, slightly outperforming the accuracy of 0.73 for *k*-means with $D\text{-}L^2$ norm. Otherwise, DCMD *k*-means with $D\text{-}L^2$ and $CC\text{-}L^2$ norms either equivalent or outperform the machine learning approaches.

Discussion

The results of our simulation studies and microbiome applications indicate that the proposed DCMD method performs well over a range of scenarios, achieving good classification performance when using sparse data as predictors. The predictive accuracy is consistently improved compared to other distances within distance-based classifiers. It is either advantageous or competitive compared to a number of machine learning methods under a wide range of scenarios. The improved performance of DCMD on sparse data results from the use of mixture distributions to represent the observed count data because the mixture distributions can not only model the underlying uncertainty in the observed sample counts but also account for zero inflation. The improvement is particularly significant in comparison to other distances within the regular *k*-means and *k*-NN classifiers.

The performance differences between the $D\text{-}L^2$ and $CC\text{-}L^2$ norms can be attributed to the data structure. In less sparse scenarios, the data structure is better modelled by a continuous rate structure, resulting in a slight advantage for the $CC\text{-}L^2$ metric. While in the higher ZP and low-count scenarios, the $D\text{-}L^2$ norm allows us to use specific differentiation of zeros into structural and non-structural and modelling expected counts directly, which can further capture the general structure of predictors used for differentiation better.

As the DCMD method derives its major improvement from a focus on modelling lower count data and the associated uncertainty, it is necessary to accurately specify an underlying set of mixture components for the low rates. The mixture also has to model low- and high-count data on the same scale, where the density of the latter is often sparse due to sparse observation intervals. Moreover, it is not feasible to apply a transformation to make the data denser due to abundant zeros and the discrete nature of the low counts. However, the weighing of nested candidate models and the suggested heuristic of specifying higher count distributions on a log-linear scale has worked well in our simulations, as it partially mimics the log-transformation commonly applied to such data.

The proposed DCMD method is formulated in a distance-based framework, so it does not include specific mechanisms for variable selection. While different predictors can alternatively be included in the distance sum, the process is not automated. In our case, we used a simple nonparametric Mann-Whitney U test for feature selection, which worked well in the study data. However, more advanced and specialized methods for feature selection can be applied separately for other applications. Additionally, we note that the model is specified to use microbiome site counts, and continuous covariates need to be modelled separately using continuous distributions, while categorical covariates can only be included as dummy variables.

These variables will also be treated on the same scale in the distance metric unless specified otherwise.

Despite these drawbacks, we believe that our core contribution, the representation of observations as distributions to reflect uncertainty and the use of distributional distance metrics, will be valuable to anyone analyzing sparse data. This formulation can compensate for some of the disadvantages inherent in distance-based methods to such an extent that it achieved competitive performance with more sophisticated classifiers, as well as specially designed approaches like NSC. The techniques that made DCMD advantageous for classification when data is expected to be sparse, particularly within a distance-based framework, should be considered for improving model performance.

Conclusion

In this paper, we present a distance-based classification method for microbiome count data. The DCMD approach models the observed data using mixture distributions and calculates L^2-norms for distance-based classification algorithms. The method is specifically designed to accurately model low-count structures, addressing the inherent sparsity by representing each observed count as a distribution, and is demonstrated to have improved performance by simulation studies and two microbiome applications. The importance of accounting for uncertainty in sparse data is emphasized, and the resulting improvements in classification accuracy when using distributions are demonstrated. The performance of the proposed DCMD is competitive to a number of machine learning methods and significantly outperforms other common metrics in distance-based classification models. The consistent and improved performances across a variety of different data structures make this approach a viable alternative for modelling and classification of microbiome count data, particularly within a distance-based framework.

Supporting information

S1 Text. Supporting information for model specification of classifiers.
(DOCX)

S1 Table. Three-class outcome: the parameter setting for each scenario and the corresponding ZP and mean count for each class over 100 replicates.
(DOCX)

S2 Table. The nested models of mixture distribution components used in fitting one of the simulated data.
(DOCX)

S3 Table. Dataset 1—Colorectal Cancer: the predictive performance of the 14 classifiers using the OTUs selected from Mann–Whitney U test on the colorectal cancer data.
(DOCX)

Acknowledgments

The authors acknowledge and are grateful for the support of the Tomcyzk AI and Microbiome Working Group.

Author Contributions

Conceptualization: Konstantin Shestopaloff, Wei Xu.

Data curation: Mei Dong.

Formal analysis: Konstantin Shestopaloff, Mei Dong, Fan Gao, Wei Xu.

Funding acquisition: Konstantin Shestopaloff, Mei Dong, Wei Xu.

Investigation: Wei Xu.

Methodology: Konstantin Shestopaloff, Mei Dong, Wei Xu.

Project administration: Wei Xu.

Resources: Konstantin Shestopaloff, Mei Dong.

Software: Konstantin Shestopaloff, Mei Dong.

Supervision: Wei Xu.

Validation: Konstantin Shestopaloff, Mei Dong, Wei Xu.

Visualization: Mei Dong.

Writing – original draft: Konstantin Shestopaloff, Mei Dong, Fan Gao, Wei Xu.

Writing – review & editing: Konstantin Shestopaloff, Mei Dong, Fan Gao, Wei Xu.

References

1. Morgan XC, Tickle TL, Sokol H, Gevers D, Devaney KL, Ward DV, et al. Dysfunction of the intestinal microbiome in inflammatory bowel disease and treatment. Genome Biol. 2012; 13:R79. https://doi.org/10.1186/gb-2012-13-9-r79 PMID: 23013615
2. Karlsson FH, Tremaroli V, Nookaew I, Bergström G, Behre CJ, Fagerberg B, et al. Gut metagenome in European women with normal, impaired and diabetic glucose control. Nature. 2013; 498:99–103. https://doi.org/10.1038/nature12198 PMID: 23719380
3. Shreiner AB, Kao JY, Young VB. The gut microbiome in health and in disease. Curr Opin Gastroenterol. 2015; 31:69–75. https://doi.org/10.1097/MOG.0000000000000139 PMID: 25394236
4. Cam LML, Neyman J. Proceedings of the Fifth Berkeley Symposium on Mathematical Statistics and Probability: Biology and problems of health. University of California Press; 1967;281–297.
5. Zhang Z. Introduction to machine learning: k-nearest neighbors. Ann Transl Med. 2016; 4. https://doi.org/10.21037/atm.2016.03.37 PMID: 27386492
6. Liu Z, Hsiao W, Cantarel BL, Drábek EF, Fraser-Liggett C. Sparse distance-based learning for simultaneous multiclass classification and feature selection of metagenomic data. Bioinformatics. 2011; 27:3242–9. https://doi.org/10.1093/bioinformatics/btr547 PMID: 21984758
7. Statnikov A, Henaff M, Narendra V, Konganti K, Li Z, Yang L, et al. A comprehensive evaluation of multicategory classification methods for microbiomic data. Microbiome. 2013; 1:11. https://doi.org/10.1186/2049-2618-1-11 PMID: 24456583
8. Rosenthal M, Aiello AE, Chenoweth C, Goldberg D, Larson E, Gloor G, et al. Impact of Technical Sources of Variation on the Hand Microbiome Dynamics of Healthcare Workers. PLoS One. 2014; 9. https://doi.org/10.1371/journal.pone.0088999 PMID: 24551205
9. Tibshirani R, Hastie T, Narasimhan B, Chu G. Diagnosis of multiple cancer types by shrunken centroids of gene expression. Proceedings of the National Academy of Sciences. 2002; 99:6567–72. https://doi.org/10.1073/pnas.082099299 PMID: 12011421
10. Zhang X, Zhao Y, Xu J, Xue Z, Zhang M, Pang X, et al. Modulation of gut microbiota by berberine and metformin during the treatment of high-fat diet-induced obesity in rats. Scientific reports. 2015; 5:14405. https://doi.org/10.1038/srep14405 PMID: 26396057
11. Knights D, Costello EK, Knight R. Supervised classification of human microbiota. FEMS Microbiology reviews. 2011; 35:343–59. https://doi.org/10.1111/j.1574-6976.2010.00251.x PMID: 21039646
12. Vangay P, Hillmann BM, Knights D. Microbiome Learning Repo (ML Repo): A public repository of microbiome regression and classification tasks. Gigascience. 2019; 8. https://doi.org/10.1093/gigascience/giz042 PMID: 31042284
13. Galkin F, Aliper A, Putin E, Kuznetsov I, Gladyshev VN, Zhavoronkov A. Human microbiome aging clocks based on deep learning and tandem of permutation feature importance and accumulated local effects. preprint. Bioinformatics; 2018. *https://doi.org/10.1101/507780*.

14. Tibshirani R. Regression Shrinkage and Selection Via the Lasso. Journal of the Royal Statistical Society: Series B (Methodological). 1996; 58:267–88.

15. Hoerl AE, Kennard RW. Ridge Regression: Biased Estimation for Nonorthogonal Problems. Technometrics. 1970; 12:55–67.

16. Weiss S, Xu ZZ, Peddada S, Amir A, Bittinger K, Gonzalez A, et al. Normalization and microbial differential abundance strategies depend upon data characteristics. Microbiome. 2017; 5:27. https://doi.org/10.1186/s40168-017-0237-y PMID: 28253908

17. Breiman L. Random Forests. Machine Learning. 2001; 45:5–32.

18. Friedman JH. Greedy Function Approximation: A Gradient Boosting Machine. Annals of statistics. 2001; 29:1189–232.

19. Wang T, Zhao H. Constructing Predictive Microbial Signatures at Multiple Taxonomic Levels. Journal of the American Statistical Association. 2017; 112:1022–31.

20. Wang T, Yang C, Zhao H. Prediction analysis for microbiome sequencing data. Biometrics. 2019; 75:875–84. https://doi.org/10.1111/biom.13061 PMID: 30994187

21. Shestopaloff K, Escobar MD, Xu W. Analyzing differences between microbiome communities using mixture distributions: Analyzing Differences Between Microbiome Communities. Statistics in Medicine. 2018; 37:4036–53. https://doi.org/10.1002/sim.7896 PMID: 30039541

22. Kostic AD, Gevers D, Pedamallu CS, Michaud M, Duke F, Earl AM, et al. Genomic analysis identifies association of Fusobacterium with colorectal carcinoma. Genome research. 2012; 22:292–98. https://doi.org/10.1101/gr.126573.111 PMID: 22009990

23. Gevers D., Kugathasan S., Denson L. A, et al. The treatment-naive microbiome in new-onset Crohn's disease. Cell host & microbe. 2014; 15:382–92. https://doi.org/10.1016/j.chom.2014.02.005 PMID: 24629344

24. Nocedal J. Updating Quasi-Newton Matrices with Limited Storage. Mathematics of Computation. 1980; 35:773–82.

25. Conn AR, Gould NIM, Toint Philippe. A Globally Convergent Augmented Lagrangian Algorithm for Optimization with General Constraints and Simple Bounds. SIAM Journal on Numerical Analysis. 1991; 28:545–72.

26. Shestopaloff K. Analysis of Ecological Communities Using Mixture Models [PhD thesis]. Toronto, Canada: University of Toronto. 2017.

27. Goutte C, Gaussier E. A Probabilistic Interpretation of Precision, Recall and F-Score, with Implication for Evaluation. In: Losada DE, Fernández-Luna JM, editors. Advances in Information Retrieval. Berlin, Heidelberg: Springer. 2005;345–59. https://doi.org/10.1016/j.ijmedinf.2004.04.017 PMID: 15694638

28. Shestopaloff K, Xu W, Escobar MD. Estimating total species using a weighted combination of expected mixture distribution component counts. Environmental and Ecological Statistics. 2020; 27:447–65.

29. Suykens JA, Vandewalle J. Least squares support vector machine classifiers. Neural processing letters. 1999; 9:293–300.

30. Pedrycz W, Skowron A, Kreinovich V. Handbook of granular computing. John Wiley & Sons. 2008;133–36.

31. Benjamini Y, Hochberg Y. Controlling the False Discovery Rate: A Practical and Powerful Approach to Multiple Testing. Journal of the Royal Statistical Society Series B (Methodological). 1995; 57:289–300.