ISLAMIC UNIVERSITY OF TECHNOLOGY

UNDER GRADUATE THESIS

Clustering Metagenomes using Expectation Maximization Algorithm

Authors:
Jubair Ibn Malik Rifat (114407)
Istiaque Ahmed (114445)

Supervisor:
Prof. Dr. M.A Mottalib
Head
Department of Computer Science and Engineering

Co-Supervisor:
M. Arifur Rahman
Lecturer
Department of Computer Science and Engineering

A thesis submitted to the Department of CSE in fulfilment of the requirements for the Degree of B.Sc Engineering in CSE.

Acdemic Year: 2014-15

October,2015

Declaration of Authorship

This is to certify that the work presented in this thesis is the outcome of the analysis and investigation carried out by Jubair Ibn Malik Rifat and Istiaque Ahmed under the supervision of Prof Dr. M.A Mottalib and M. Arifur Rahman in the Department of Computer Science and Engineering (CSE), IUT, Dhaka, Bangladesh. It is also declared that neither of this thesis nor any part of this thesis has been submitted anywhere else for any degree or diploma. Information derived from the published and unpublished work of others has been acknowledged in the text and a list of references is given.

Authors:

Jubair Ibn Malik Rifat
Student ID : 114407

Istiaque Ahmed
Student ID : 114445

Supervisor:

Prof. Dr. M.A Mottalib
Head
Department of Computer Science and Engineering
Islamic University of Technology (IUT)

Abstract

Clustering metagenome refers to group genes with similar expression patterns of a metagenomic data set into clusters with the hope that these clusters correspond to groups of functionally related genes. It allows access to uncultivated microbial populations that may have important roles in natural and engineered ecosystems. Proper clustering of Metgenome sequence is a very essential step in recovering genomes and understanding microbial functions. We took the distance matrix from the expression matrix of a metagenomic sequence and used Expectation Maximization (EM) algorithm for clustering the metagenome. After clustering we label the clusters with proper name, we match the cluster nucleotides with reference genome of bacteria in HMPDAC and name the clusters with the bacteria title given in database. Finally for healthy/ patient sample we will show the percentage of bacteria and infer that since this bacteria is higher it might be causing the problem.

Acknowledgements

In full gratitude, we would like to acknowledge the following individuals who encouraged, inspired, supported, assisted and sacrificed themselves to help our pursuit of the successful thesis work.

From our academy,we would like to thank Prof. Dr. M.A.Mottalib, Head, Department of CSE,IUT for the continuous support of our thesis work and related research,for his patience, motivation and immense knowledge. His guidance helped us in all the time of research and writing of this thesis.

We would also like to thank M.Arifur Rahman,Lecturer,CSE,IUT for his insightful comments and encouragement, but also for hard question which incented us to widen our research from various perspectives.

Last but not the list, we would like to thank our families for supporting us spiritually throughout the writing of this thesis and our life in general.

With Regards,

Jubair Ibn Malik Rifat(114407)

Istiaque Ahmed(114445)

Table of Contents

Chapter 1

Introduction

1.1 Overview

Bioinformatics is an interdisciplinary field that develops methods and software tools for understanding biological data. As an interdisciplinary field of science, bioinformatics combines computer science, statistics, mathematics, and engineering to analyze and interpret biological data. Bioinformatics is both an umbrella term for the body of biological studies that use computer programming as part of their methodology, as well as a reference to specific analysis "pipelines" that are repeatedly used, particularly in the fields of genetics and genomics. Common uses of bioinformatics include the identification of candidate genes and nucleotides (SNPs). Often, such identification is made with the aim of better understanding the genetic basis of disease, unique adaptations, desirable properties (esp. in agricultural species), or differences between populations. In a less formal way, bioinformatics also tries to understand the organizational principles within nucleic acid and protein sequences.

Bioinformatics has become an important part of many areas of biology. In experimental molecular biology, bioinformatics techniques such as image and signal processing allow extraction of useful results from large amounts of raw data. In the field of genetics and genomics, it aids in sequencing and annotating genomes and their observed mutations. It plays a role in the text mining of biological literature and the development of biological and gene ontologies to organize and query biological data. It also plays a role in the analysis of gene and protein expression and regulation. Bioinformatics tools aid in the comparison of genetic and genomic data and more generally in the understanding of evolutionary aspects of molecular biology. At a more integrative level, it helps analyze and catalogue the biological pathways and networks that are an important part of systems biology. In structural biology, it aids in the simulation and modeling of DNA, RNA, and protein structures as well as molecular interactions.

In the field of genetics and genomics, it aids in sequencing and annotating genomes and their observed mutations. It plays a role in the textual mining of biological literature and the development of biological and gene ontology's to organize and query biological data. It plays a role in the analysis of gene and protein expression and regulation. Bioinformatics tools aid in the comparison of genetic and genomic data and more generally in the understanding of evolutionary aspects of molecular biology. At a more integrative level, it helps analyze and catalogue the biological pathways and networks that are an important part of systems biology. In structural biology, it aids in the simulation and modeling of DNA, RNA, and protein structures as well as molecular interactions.

Metagenomics [1] is a new field of study that provides a deeper insight into the microbial world compared to the traditional single-genome sequencing technologies. Traditional methods for studying individual genomes are well developed. However, they are not appropriate for studying microbial samples from the environment because traditional methods rely upon cultivated clonal cultures while more than 99% of bacteria are unknown and cannot be cultivated and isolated [2]. Metagenomics uses technologies that sequence uncultured bacterial genomes in an environmental sample directly [3], and thus makes it possible to study organisms which cannot be isolated or are difficult to grow in a lab. It provides hope for a better understanding of natural microbial diversity as well as their roles and interactions. It also opens new opportunities for medicine, biotechnology, agricultural studies and ecology.

Many well-known metagenomics projects use the whole genome shotgun sequencing approach in combination with Sanger sequencing technologies. This approach has produced datasets from the Sargasso Sea[4], Human Gut Microbiome [5] and Acid Mine Drainage Biolm [6]. However, new sequencing technologies have evolved over the past few years. The sequencing process has been greatly parallelized, producing millions of reads with much faster speed and lower cost. Since NGS technologies are much cheaper, they allow sequencing to be performed at a much greater depth. The only drawback is that read length is reduced -NGS reads are usually of lengths 25-150 (Illumina/SOLiD) compared to 800-1000 bps in Sanger reads.

1.2 Problem Statement

The primary goals of metagenomics are to describe the populations of microorganisms and to identify their roles in the environment. Ideally, we want to identify complete genomic sequences of all organisms present in a sample. However, metagenomic data is very complex, containing a large number of sequence reads from many species. The number of species and their abundance levels are unknown. The assembly of a single genome is already a difficult problem, complicated by repeats and sequencing errors which may lead to high fragmentation of contigs and mis-assembly. In a metagenomic data, in addition to repeats within individual genomes, genomes of closely related species may also share homologous sequences, which could lead to even more complex repeat patterns that are very difficult to resolve. A lot of research has been done for assembling single genomes [7-10]. But due to the lack of research on metagenomic assemblers, assemblers designed for individual genomes are routinely used in metagenomic projects [4, 6]. It has been shown that these assemblers may lead not only to mis-assembly, but also severe fragmentation of contigs [11]. A plausible approach to improve the performance of such assemblers is to separate reads from different organisms present in a dataset before the assembly.

1.3 Motivation and Scope

Microorganisms can be found in almost every environment of the Earth’s biosphere and are responsible for numerous biological activities including carbon and nitrogen cycling (1), organic contaminant remediation (2–4) and human health and disease. Many human disorders, such as type 2 diabetes (T2D), obesity, dental cavities, cancer and some immune-related diseases, are known to be related with a single or a group of microorganisms (5–11). In addition, different strains within the same species may have completely different impacts on human health, such as

Escherichia coliO157:H7, which is a highly virulent E. coli strain, whereas most other strains in this same species are non-pathogenic. Thus, characterization and identification of microbial strains/species in the environment and individual human hosts is of crucial importance to reveal human–microbial interactions, especially for patients with microbial-mediated disorders.

Although different technologies have been developed, the characterization and identification of known microorganisms at strain/species levels remain challenging, mainly due to the lack of high-resolution tools and the extremely diverse nature of microbial communities. Currently, the most commonly used approach to characterize and identify microorganisms in complex environments is to sequence 16S ribosomal RNA (rRNA) gene amplicons using universally conserved primers (13). However, owing to the high similarity of 16S rRNA gene sequences among different microorganisms, this approach can only confidently identify microorganisms at high taxonomic levels (e.g. genus and family) but not at the species/strain level, although species identification had been attempted in a few studies with less complex communities(14,15). Even at the genus level, resolution problems with 16S rRNA gene sequences have been reported by many investigators (16). Therefore, it is necessary to use other molecular markers to identify and characterize microorganisms at the strain/species level in complex environments.

To summarize the motivation of our work, we can say- Firstly, metagenomics has become a major issue in Bioinformatics. Secondly, 99% of micro-organisms presents in many natural environments are not readily culturable and therefor, not assessable. Thirdly, novel genes are high potential for use in pharmaceutical products or production processes and those genes can be identified clearly from metagenome. Finally, metagenome study is increasing research scope in bioinformatics.

1.4 Research Challenges

Metagenomes contain a large amount of data and this data are totally unstructured as we saw those data are collected directly from environment. So, for processing metagenome and finally finding valuable information from those we will be needing a good algorithm. Again, as the data of metagenome are unstructured. So it is a unsupervised learning. As a result, for processing these data we need clustering.

Many computational tools have been developed for separating reads from different species or groups of related species (we will refer to the problem as the clustering of reads). Some of the tools also estimate the abundance levels and genome many computational tools have been developed for separating reads from different species or groups of related species (we will refer to the problem as the clustering of reads). Some of the tools also estimate the abundance levels and genome sizes of species. These tools are usually classified as similarity-based (or phylogeny-based) and composition-based. The purpose of similarity-based methods is to analyze the taxonomic content of a sample. Small-scale approaches involving 16S rRNAs and 18S rRNAs [12] are commonly used to determine evolutionary relationships by analyzing fragments that contain marker genes and comparing them with known marker genes. These methods take advantage of small number of fragments containing marker genes and require reads to have at least 1000 bps. Two other tools handle a larger number of fragments: MEGAN and CARMA . MEGAN aligns reads to databases of known sequences using BLAST and assigns reads to taxa by the lowest common ancestor approach. CARMA performs phylogenetic classification of unassembled reads using all Pfam domains and protein families as phylogenetic markers. These two methods work for very short reads (as short as 35 bps for MEGAN and 80 bps for CARMA). However, a large fraction of sequences may remain unclassified by these methods because of the absence of closely related sequences in the databases.

1.5 Thesis Outline

In Chapter 1 we have talked about the introduction of our study in a precised manner. Chapter 2 deals with the basic metagenomic analysis and clustering method and some highlighted evolutionary approaches with a brief discussion about “metagenome” clustering method, an algorithm for clustering metagenome. Chapter 3 will be discussed about our proposed algorithm and some elaborate discussion. Here we showed how “Expectation Maximization Algorithm” can be used for clustering in a fully general manner. Chapter 4 will consist of the experimental analysis and result comparisons. In chapter 5 we discussed conclusion including summary of contribution, limitation and future work.

Chapter 2

Literature Reviews

2.1 Metagenomics

Metagenomics is the study of multiple genomes i.e., metagenomes are taken directly from the environment. While the traditional methods, in which organisms were cultured in predetermined media under the laboratory conditions, were able to produce a diversity profile; they missed the vast majority of biodiversity present in the environment. Recently, Kevin Chen and Lior Pachter (researchers at the University of California, Berkeley) defined metagenomics as "the application of modern genomics techniques to the study of communities of microbial organisms directly in their natural environments, bypassing the need for isolation and lab cultivation of individual species." [1]

Metagenomics is currently the only way to study genetic diversity present in the viral communities as they do not contain any universal phylogenetic marker (like 16S RNA for bacteria) which are typically used to culture bacterial organisms. Culturing the host and then infecting them with specific viruses or viral DNA obtained from the environment in the laboratory condition is not yet streamlined. However, the metagenome sample obtained from the environment directly represent communities of population as opposed to isolated populations and thus, metagenomics may help reveal information about how the populations co-evolve. One key step in understanding our microbiota is to identify lineages that have coevolved with humans (or with mammals in general), and to identify the genomic consequences of this coevolution. Coevolution between a host and a beneficial symbiont, or a pathogen, is defined as reciprocal adaptation of each lineage in response to the other. For example, genetic changes that increase production of a metabolite by an intestinal bacterium may trigger selection of changes in the host genome that promote uptake or prevent synthesis of that metabolite.

2.2 Advancement in Sequencing Technologies

With the advent of powerful and economic next generation sequencing technologies such as Sanger sequencing or massively parallel pyrosequencing, metagenomics has become more popular. Sanger sequencing is based on chain termination with di-oxy-nucleotides whereas pyrosequencing is based on sequencing by synthesis method, i.e., the idea is to detect pyrophosphate release when nucleotide is incorporated. Sanger sequences are longer ~750 base pairs (bp) than pyrosequencing techniques, specifically 454 produces reads of length ~100 to ~200 bp. 454 titanium series produces reads of length ~400 to ~500 bp. Advantages of pyrosequencing over Sanger sequencing include a 10 much lower per-base cost and no requirement for cloning. [10] These generate sequence trace files from which base calling is done.

2.3 Bioinformatics Pipeline for metagenome processing

Once the raw reads are obtained, the data need to be processed and analyzed to see what story is hidden in it.

2.3.1 Sequence Processing

Processing of both, the genomic and metagenomic sequence data, follow common steps like preprocessing the sequence reads, assembly, Gene Prediction and Annotation. However, the main difference between genomes and metagenomes is that the former has a fixed end-point like one or more completed chromosomes. However, in the case of metagenomes, we just get draft assemblies and may be sometimes almost complete genome of dominant populations. [1][7]

2.3.1.1 Preprocessing the sequence reads

This is a very important step in metagenome processing. It involves base calling of raw data, removal of low complexity reads, removal of contaminant sequences, and removal of outliers, i.e, reads with very short length. Base calling involves identifying DNA bases from the DNA sequencing trace files. The most commonly used base calling tool is phred. [2] phred assigns a quality value, q, to each called base based on the per-base error probability, p by using the following formula: $q = -10 \times \log10 (p)$. The other tool which is used in many other researches is Prinseq [15]. Prinseq is a web as well as a standalone tool that allows to filter, trim and reformat the metagenome data. It removes low quality reads based on quality scores obtained from phred to avoid complications in assemblies and downstream analysis. It trims poly-A/T tails, repeats of A's and T's at the end of the sequence because it can result in false positives during similarity searches, since they have a good alignment with low complexity regions or sequences with tails. It removes sequences with a lot of ambiguous bases, i.e., sequences with high number of Ns. A position in the sequence where a base cannot be identified is replaced by the letter N which means it is an ambiguous base. For removing low complexity reads, it calculates the sequence complexity using both DUST and Entropy approached. DUST is the heuristic used to mask low complexity regions during BLAST search. [11] DUST computes scores based on 11 how often different triplets occur in the sequences and are scaled from 0 to 100 and higher scores imply lower complexity. In case of Entropy approach, entropy values of trinucleotides in the sequence is computed and scores are scaled from 0 to 100 where lower entropy would mean low complexity.

2.3.1.2 Assembly

Assembly is the process of combining reads based on similarity to obtain contiguous DNA segments called contigs. There are challenges in assembling metagenomes as there could be problems like co-assembly of reads coming from different species because of non-uniform species distribution. This can happen if there is high sequence similarity between reads coming from closely related species. There are many publicly available assembly programs like Phrap, Celera Assembler, Newbler but these were all designed for assembling genomes from isolates and not for metagenomes which comprise of multiple species with read coverage that is non uniform. Therefore, their performances vary significantly. To mitigate these problems for de novo assembly, we need to pass our data through more than one assembler so that it helps solving mis-assembly of the largest contigs. To further strengthen our assembly, we can perform multiple assemblies by tweaking parameters for a particular assembler. To be absolutely sure of our assembly so that problems do not percolate to further downstream analysis, we can perform manual inspection using scaffolding programs like ScaffViz or visualization programs like Consed. [3] Comparative assemblies are easier to work with; where a reference genome or fully sequenced genome is passed to assembler along with the metagenome. AMOS is an assembler that performs comparative assembly.

2.3.1.3 Gene Prediction and Annotation

The process of identifying protein coding genes and RNA sequences is known as gene prediction. There are two ways of performing gene calling: one is evidenced-based and the other is ab initio gene prediction. The evidenced-based method is based on BLAST similarity search to find homologs against a database of previously found genes. The ab initio gene prediction method allows gene identification based on intrinsic features of the DNA sequence to differentiate between coding regions of a sequence from non-coding regions. This method is useful to identify those genes that do not have homologs to existing database sequences, and to find novel genes. For the ab initio method, there are many gene-prediction tools, some of which requires training data set (fgenes) while some are 12 self-trained on the target sequence (MetaGene, Glimmer, Genemark). MetaGene is the prokaryotic gene prediction tool developed specifically for metagenomes. The program does not require training data set and it estimates di-codon frequency from the GC content of a given sequence. [12] In case of complete genomes, both the ways of gene prediction are employed and the hits to genes in the database act as training sets. In case of unassembled pyrosequencing reads and high complexity metagenomes, evidence-based gene prediction is the only method used because of the fragmented nature and short read lengths of these data sets; as pointed out by Mavromatis [7]. Even in case of less complex communities, it is better to perform gene prediction on both reads and contigs because reads from less abundant organism remains unassembled and these reads may contain important functionality. The most commonly used tool to predict RNA genes like tRNA and rRNA is tRNA scan. [9] Finally, to assign protein function to metagenome data, protein sequences are compared to the database of protein family sequences like TIGRFAM, Pfam, and COGs. [7]

2.3.2 Data Analysis

Depending on the metagenome, there are different data analysis methods. The most common analysis methods are composition analysis on contigs, reclassification of reads after preprocessing, and binning. Next, we cover the topic of binning.

The process of associating sequence data to the contributing species is known as binning. The highly reliable binning is assembly, as reads coming from same species are assembled together. This is not the case in metagenome data sets as there are chances of co-assembly. The two most common ways to bin are based on sequence similarity and sequence composition. In case of sequence similarity, we compare our metagenome data using tools such as BLAST and MEGAN (Huson et al 2007), a metagenome analyzer to separate metagenome fragments based on phylogenetic groups. If the suitable marker genes are present, then assignment of fragments based on taxonomic group is feasible. However, in case of absence of marker genes for your metagenome, the other approach is to use (G+C) content along with phylogenetic information to separate fragments. The other binning method, based on sequence composition is entirely different as it makes use of oligonucleotide frequencies which 13 supposedly are distinct and help separate different genomes. The word length can range from 1 to 8, with longer words giving better resolution but are expensive computationally. Therefore, typical word length range from 3 to 6 bases long. This method is so far the best method. As pointed out by Teeling [19], in their experiment on 9054 genomic fragments generated from 118 complete bacterial genomes the scores and results obtained using tetra-nucleotide analysis were far superior compared to GC content binning method. The standalone tool available online for tetra-nucleotide analysis is called TETRA (Teeling et al 2004). TETRA computes z-scores from the divergence between observed versus expected tetra-nucleotide frequencies. To compute observed values, it counts frequencies of all 44 = 256 possible tetra-nucleotides for DNA sequences (both forward and reverse strand). To compute expected values, it counts expected frequencies for each tetra-nucleotide “by means of a maximal-order Markov model from the sequences' di- and tri-nucleotide composition.” [20]

2.4 Expectation Maximization Algorithm

In statistics, an expectation maximization algorithm is an iterative method for finding maximum likelihood or maximum a posteriori (MAP) estimates of parameters in statistical models, where the model depends on unobserved latent variables. The EM iteration alternates between performing an expectation (E) step, which creates a function for the expectation of the log-likelihood evaluated using the current estimate for the parameters, and a maximization (M) step, which computes parameters maximizing the expected log-likelihood found on the *E* step. These parameter-estimates are then used to determine the distribution of the latent variables in the next E step

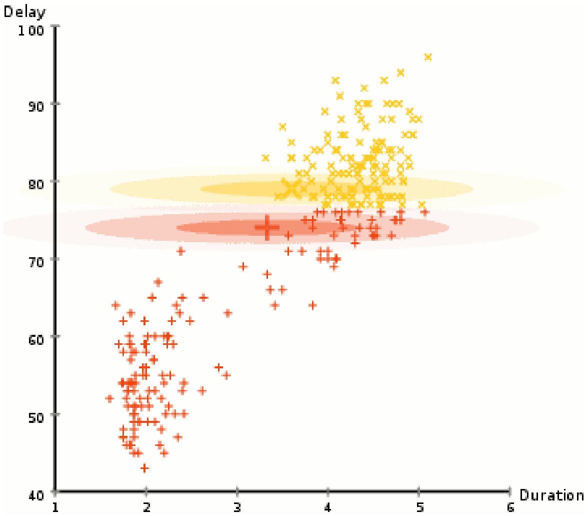


Figure 2.1: Clustering with EM Algorithm

The EM algorithm is used to find (locally) maximum likelihood parameters of a statistical model in cases where the equations cannot be solved directly. Typically these models involve latent variables in addition to unknown parameters and known data observations. That is, either there are missing values among the data, or the model can be formulated more simply by assuming the existence of additional unobserved data points. For example, amixture model can be described more simply by assuming that each observed data point has a corresponding unobserved data point, or latent variable, specifying the mixture component that each data point belongs to.

Finding a maximum likelihood solution typically requires taking the derivatives of the likelihood function with respect to all the unknown values — viz. the parameters and the latent variables — and simultaneously solving the resulting equations. In statistical models with latent variables, this usually is not possible. Instead, the result is typically a set of interlocking equations in which the solution to the parameters requires the values of the latent variables and vice versa, but substituting one set of equations into the other produces an unsolvable equation.

The EM algorithm proceeds from the observation that the following is a way to solve these two sets of equations numerically. One can simply pick arbitrary values for one of the two sets of unknowns, use them to estimate the second set, then use these new values to find a better estimate of the first set, and then keep alternating between the two until the resulting values both converge to fixed points. It's not obvious that this will work at all, but in fact it can be proven that in this particular context it does, and that the derivative of the likelihood is (arbitrarily close to) zero at that point, which in turn means that the point is either a maximum or a saddle point.[12] In general there may be multiple maxima, and there is no guarantee that the global maximum will be found. Some likelihoods also have singularities in them, i.e. nonsensical maxima. For example, one of the "solutions" that may be found by EM in a mixture model involves setting one of the components to have zero variance and the mean parameter for the same component to be equal to one of the data points.

2.4.1 Description

Given a statistical model which generates a set $\mathbf{X}$ of observed data, a set of unobserved latent data or missing values $\mathbf{Z}$, and a vector of unknown parameters $\boldsymbol{\theta}$, along with a likelihood function $L(\boldsymbol{\theta}; \mathbf{X}, \mathbf{Z}) = p(\mathbf{X}, \mathbf{Z}|\boldsymbol{\theta})$, the maximum likelihood estimate (MLE) of the unknown parameters is determined by the marginal likelihood of the observed data

$$L(\boldsymbol{\theta}; \mathbf{X}) = p(\mathbf{X}|\boldsymbol{\theta}) = \sum_{\mathbf{Z}} p(\mathbf{X}, \mathbf{Z}|\boldsymbol{\theta})$$

However, this quantity is often intractable (e.g. if $\mathbf{Z}$ is a sequence of events, so that the number of values grows exponentially with the sequence length, making the exact calculation of the sum extremely difficult).

The EM algorithm seeks to find the MLE of the marginal likelihood by iteratively applying the following two steps:

Expectation step (E step): Calculate the expected value of the log likelihood function, with respect to the conditional distribution of $\mathbf{Z}$ given $\mathbf{X}$ under the current estimate of the parameters $\boldsymbol{\theta}^{(t)}$:

$$Q(\boldsymbol{\theta}|\boldsymbol{\theta}^{(t)}) = \operatorname{E}_{\mathbf{Z}|\mathbf{X},\boldsymbol{\theta}^{(t)}}\left[\log L(\boldsymbol{\theta}; \mathbf{X}, \mathbf{Z})\right]$$

Maximization step (M step): Find the parameter that maximizes this quantity:

$$\boldsymbol{\theta}^{(t+1)} = \underset{\boldsymbol{\theta}}{\operatorname{arg\,max}}\ Q(\boldsymbol{\theta}|\boldsymbol{\theta}^{(t)})$$

Note that in typical models to which EM is applied:

1. The observed data points $\mathbf{X}$ may be discrete (taking values in a finite or countably infinite set) or continuous (taking values in an uncountably infinite set). There may in fact be a vector of observations associated with each data point.

2. The missing values (aka latent variables) $\mathbf{Z}$ are discrete, drawn from a fixed number of values, and there is one latent variable per observed data point.

3. The parameters are continuous, and are of two kinds: Parameters that are associated with all data points, and parameters associated with a particular value of a latent variable (i.e. associated with all data points whose corresponding latent variable has a particular value).

However, it is possible to apply EM to other sorts of models.

The motivation is as follows. If we know the value of the parameters $\boldsymbol{\theta}$, we can usually find the value of the latent variables $\mathbf{Z}$ by maximizing the log-likelihood over all possible values of $\mathbf{Z}$, either simply by iterating over $\mathbf{Z}$ or through an algorithm such as the Viterbi algorithm for hidden Markov models. Conversely, if we know the value of the latent variables $\mathbf{Z}$, we can find an estimate of the parameters $\boldsymbol{\theta}$ fairly easily, typically by simply grouping the observed data points according to the value of the associated latent variable and averaging the values, or some function of the values, of the points in each group. This suggests an iterative algorithm, in the case where both $\boldsymbol{\theta}$ and $\mathbf{Z}$ are unknown:

1. First, initialize the parameters $\boldsymbol{\theta}$ to some random values.

2. Compute the best value for $\mathbf{Z}$ given these parameter values.

3. Then, use the just-computed values of $\mathbf{Z}$ to compute a better estimate for the parameters. Parameters associated with a particular value of will use only those data points whose associated latent variable has that value.

4. Iterate steps 2 and 3 until convergence.

The algorithm as just described monotonically approaches a local minimum of the cost function, and is commonly called *hard EM*. The *k*-means algorithm is an example of this class of algorithms.

However, one can do somewhat better: Rather than making a hard choice for $\mathbf{Z}$ given the current parameter values and averaging only over the set of data points associated with a particular value of $\mathbf{Z}$, one can instead determine the probability of each possible value of $\mathbf{Z}$ for each data point, and then use the probabilities associated with a particular value of $\mathbf{Z}$ to compute a weighted average over the entire set of data points. The resulting algorithm is commonly called *soft EM*, and is the type of algorithm normally associated with EM. The counts used to compute these weighted averages are called *soft counts* (as opposed to the *hard counts* used in a hard-EM-type algorithm such as *k*-means). The probabilities computed for $\mathbf{Z}$ are posterior probabilities and are what is computed in the E step. The soft counts used to compute new parameter values are what is computed in the M step.

2.4.2 Properties

Speaking of an expectation (E) step is a bit of a misnomer. What is calculated in the first step are the fixed, data-dependent parameters of the function *Q*. Once the parameters of *Q* are known, it is fully determined and is maximized in the second (M) step of an EM algorithm.

Although an EM iteration does increase the observed data (i.e. marginal) likelihood function there is no guarantee that the sequence converges to a maximum likelihood estimator.
For multimodal distributions, this means that an EM algorithm may converge to a local maximum of the observed data likelihood function, depending on starting values. There are a variety of heuristic or meta-heuristic approaches for escaping a local maximum such as random restart (starting with several different random initial estimates $\theta^{(t)}$), or applying simulated annealing methods.

EM is particularly useful when the likelihood is an exponential family: the E step becomes the sum of expectations of sufficient statistics, and the M step involves maximizing a linear function. In such a case, it is usually possible to derive closed form updates for each step, using the Sundberg formula (published by Rolf Sundberg using unpublished results of Per Martin-Löf and Anders Martin-Löf).

The EM method was modified to compute maximum a posteriori (MAP) estimates for Bayesian inference in the original paper by Dempster, Laird, and Rubin.

There are other methods for finding maximum likelihood estimates, such as gradient descent, conjugate gradient or variations of the Gauss–Newton method. Unlike EM, such methods typically require the evaluation of first and/or second derivatives of the likelihood function.

2.5 Clustering

2.5.1 What is clustering?

Clustering can be considered the most important unsupervised learning problem; so, as every other problem of this kind, it deals with finding a structure in a collection of unlabeled data.
A loose definition of clustering could be “the process of organizing objects into groups whose members are similar in some way”.
A cluster is therefore a collection of objects which are “similar” between them and are “dissimilar” to the objects belonging to other clusters.
We can show this with a simple graphical example:

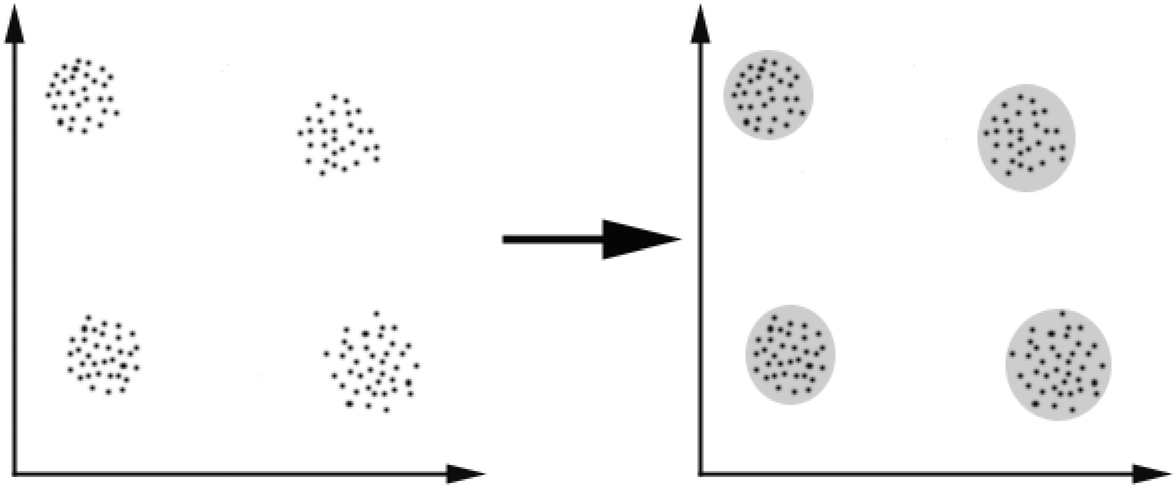


Figure 2.2: Clustering Example

In this case we easily identify the 4 clusters into which the data can be divided; the similarity criterion is distance: two or more objects belong to the same cluster if they are “close” according to a given distance (in this case geometrical distance). This is called distance-based clustering.

Another kind of clustering is conceptual clustering: two or more objects belong to the same cluster if this one defines a concept common to all that objects. In other words, objects are grouped according to their fit to descriptive concepts, not according to simple similarity measures.

2.5.2 The Goals of Clustering

So, the goal of clustering is to determine the intrinsic grouping in a set of unlabeled data. But how to decide what constitutes a good clustering? It can be shown that there is no absolute “best” criterion which would be independent of the final aim of the clustering. Consequently, it is the user which must supply this criterion, in such a way that the result of the clustering will suit their needs.
For instance, we could be interested in finding representatives for homogeneous groups (data reduction), in finding “natural clusters” and describe their unknown properties (“natural” data types), in finding useful and suitable groupings (“useful” data classes) or in finding unusual data objects (outlier detection).

2.5.3 Possible Applications

Clustering algorithms can be applied in many fields, for instance:

1. Marketing: finding groups of customers with similar behavior given a large database of customer data containing their properties and past buying records;

2. Biology: classification of plants and animals given their features;
3. Libraries: book ordering;
4. Insurance: identifying groups of motor insurance policy holders with a high average claim cost; identifying frauds;
5. City-planning: identifying groups of houses according to their house type, value and geographical location;
6. Earthquake studies: clustering observed earthquake epicenters to identify dangerous zones;
7. WWW: document classification; clustering weblog data to discover groups of similar access patterns.

2.5.4 Requirements

The main requirements that a clustering algorithm should satisfy are:

- scalability;
- dealing with different types of attributes;
- discovering clusters with arbitrary shape;
- minimal requirements for domain knowledge to determine input parameters;
- ability to deal with noise and outliers;
- insensitivity to order of input records;
- high dimensionality;
- interpretability and usability

2.5.4 Problems

There are a number of problems with clustering. Among them:

- current clustering techniques do not address all the requirements adequately (and concurrently);
- dealing with large number of dimensions and large number of data items can be problematic because of time complexity;
- the effectiveness of the method depends on the definition of “distance” (for distance-based clustering);
- if an obvious distance measure doesn’t exist we must “define” it, which is not always easy, especially in multi-dimensional spaces;
- the result of the clustering algorithm (that in many cases can be arbitrary itself) can be interpreted in different ways

Chapter 3

Proposed Method

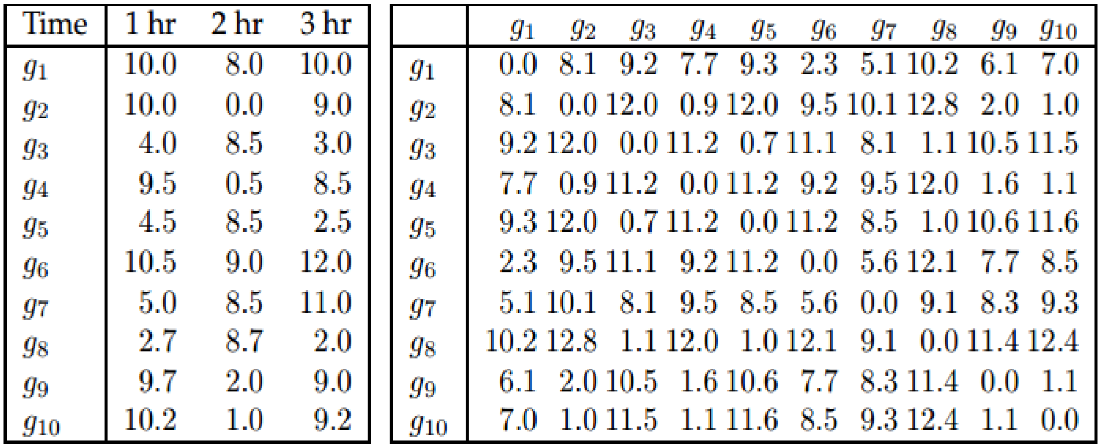
3.1 Skeleton of Proposed Method

Initially DNA is extracted from the environment directly and it is known as metagenomics. Metagenomes are manipulated using an enzyme called “Restriction Endonucleases”. After that a library of metagenomics is constructed. Finally DNA analysis is performed.

3.1.1 Sequence Analysis

The metagenome or the DNA sequence generally consists of a large number of nucleotides. A new approach to analyzing gene functions has emerged. DNA arrays allow one to analyze the expression levels (amount of mRNA produced in the cell)of many genes under many time points and conditions and to reveal which genes are switched on and switched off in the cell. The outcome of the study is an n X m expression matrix, I with the n rows corresponding to genes, and the m columns corresponding to different time points and different conditions. The expression matrix I represents intensities of hybridization signals as provided by a DNA array. In reality, expression matrices usually represent transformed and normalized intensities rather than the raw intensities obtained as a result of a DNA array experiment.

Clustering algorithms group genes with similar expression patterns into clusters with the hope that these clusters correspond to groups of functionally related genes. To cluster the expression data, the n × m expression matrix is often transformed into an n×n distance matrix $d=(d_{i,j})$ where $d_{i,j}$ reflects how similar the expression patterns of genes i and j are.



Time	1 hr	2 hr	3 hr
g_1	10.0	8.0	10.0
g_2	10.0	0.0	9.0
g_3	4.0	8.5	3.0
g_4	9.5	0.5	8.5
g_5	4.5	8.5	2.5
g_6	10.5	9.0	12.0
g_7	5.0	8.5	11.0
g_8	2.7	8.7	2.0
g_9	9.7	2.0	9.0
g_{10}	10.2	1.0	9.2

(a) Intensity matrix, **I**

	g_1	g_2	g_3	g_4	g_5	g_6	g_7	g_8	g_9	g_{10}
g_1	0.0	8.1	9.2	7.7	9.3	2.3	5.1	10.2	6.1	7.0
g_2	8.1	0.0	12.0	0.9	12.0	9.5	10.1	12.8	2.0	1.0
g_3	9.2	12.0	0.0	11.2	0.7	11.1	8.1	1.1	10.5	11.5
g_4	7.7	0.9	11.2	0.0	11.2	9.2	9.5	12.0	1.6	1.1
g_5	9.3	12.0	0.7	11.2	0.0	11.2	8.5	1.0	10.6	11.6
g_6	2.3	9.5	11.1	9.2	11.2	0.0	5.6	12.1	7.7	8.5
g_7	5.1	10.1	8.1	9.5	8.5	5.6	0.0	9.1	8.3	9.3
g_8	10.2	12.8	1.1	12.0	1.0	12.1	9.1	0.0	11.4	12.4
g_9	6.1	2.0	10.5	1.6	10.6	7.7	8.3	11.4	0.0	1.1
g_{10}	7.0	1.0	11.5	1.1	11.6	8.5	9.3	12.4	1.1	0.0

(b) Distance matrix, **d**

Figure 3.1 : Matrixes used for clustering

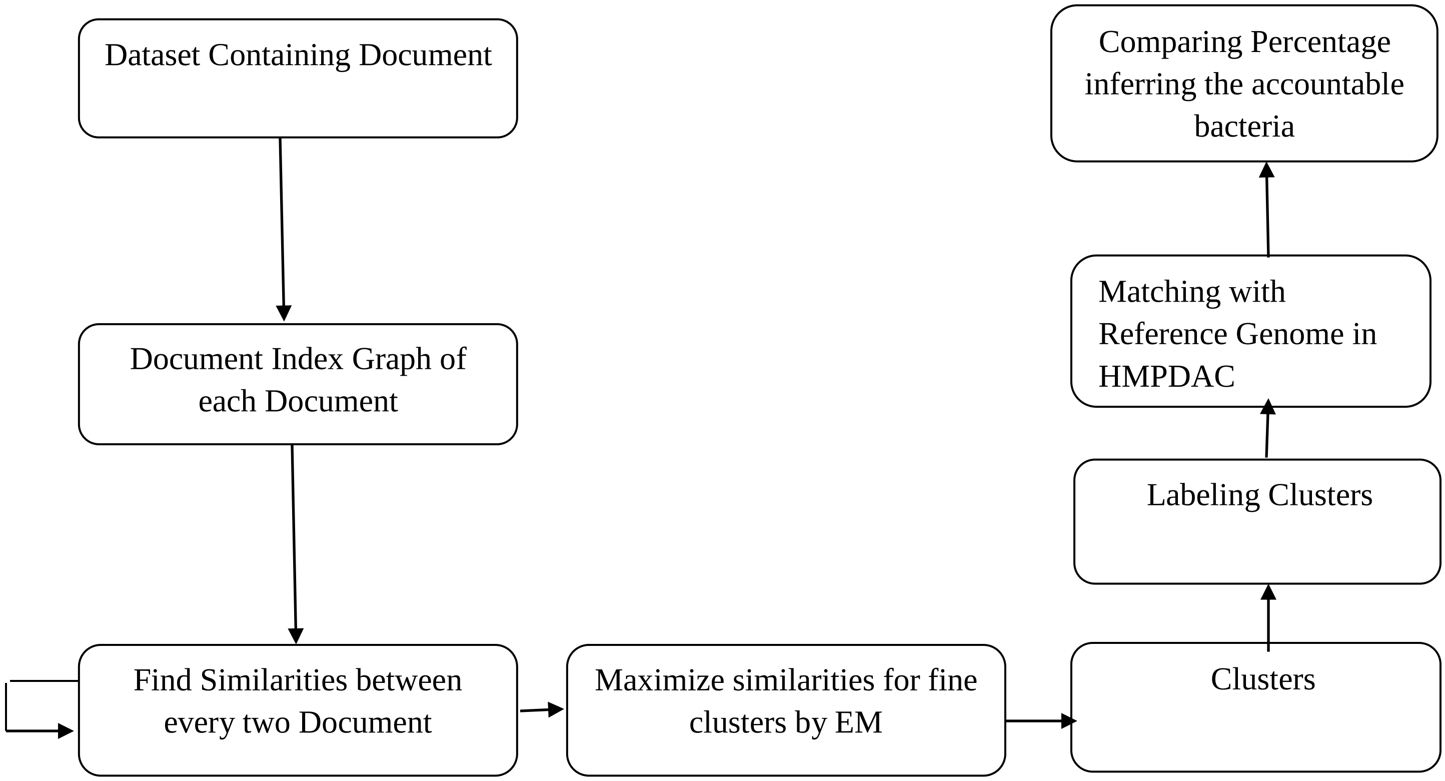
3.1.2 Clustering & Cluster Manipulation

Then we run Expectation Maximization Algorithm on the values of the distance matrix and cluster the different values. Then we will label the clusters with proper name. If there is a cluster than all the nucleotide sequences in the cluster are similar. Then we have to find the name of the bacteria where those cluster of neucleotides are present. Then provide the percentage of bacteria which is present in a human sample. After that we will take cluster neucleotides, match with reference genome of bacteria in HMPDAC, name them with the bacteria title given in database. Then for healthy/patient sample we will show the percentage of bacteria and infer that since this bacteria is higher it might be causing the problem.

3.2 Proposed Algorithm

Here we are using Expectation Maximization Algorithm to cluster those metagenomes. EM Algorithm has a great use of clustering. First the algorithm will consider the expectation of keeping a gene in a cluster and iteratively it will maximize the probability of being the gene in that cluster .

3.3 Flow Chart



Dataset Containing Document
Document Index Graph of each Document
Find Similarities between every two Document
Maximize similarities for fine clusters by EM
Clusters
Labeling Clusters
Matching with Reference Genome in HMPDAC
Comparing Percentage inferring the accountable bacteria

Figure 3.2: Flow chart of how to cluster

Chapter 4

Results and Discussion

4.1 Clustering Algorithm Overview

Clustering is an important means of data mining and of algorithms that separate data of similar nature. Unlike the classification algorithm, clustering belongs to the unsupervised type of algorithms. Two representatives of the clustering algorithms are the *K*-means algorithm and the expectation maximization (EM) algorithm. EM and *K*-means are similar in the sense that they allow model refining of an iterative process to find the best congestion. However, the *K*-means algorithm differs in the method used for calculating the Euclidean distance while calculating the distance between each of two data items; and EM uses statistical methods. The EM algorithm is often used to provide the functions more effectively.

An important question is how to decide what constitutes good clustering, since it is commonly acknowledged that there is no absolute 'best' criterion which would be independent of the final aim of the clustering.[2,4] Consequently, it is the user who must supply the criterion that best suits their particular needs, and the result of the clustering algorithm can be interpreted in different ways. There are different types of clustering, which have been extensively reviewed.[2] Briefly, one approach is to group data in an exclusive way, so that if a certain item of data belongs to a definite cluster, then it could not be included in another cluster. Another approach, the so-called overlapping clustering, uses fuzzy sets to cluster data in such a way that each item of data may belong to two or more clusters with different degrees of membership. In this case, data will be associated to an appropriate membership value. Alternatively, in the third approach (hierarchical clustering), the algorithm begins by setting each item of data as a cluster and proceeds by uniting the two nearest clusters.[2] After a few iterations it reaches the final clusters wanted. Finally, the fourth kind of clustering uses a completely probabilistic approach. We examined the performance of two of the most used clustering algorithms: *K*-means and EM as follows.

4.1.1 K-means Clustering

The cluster analysis procedure is analysed to determine the properties of the data set and the target variable. It is typically used to determine how to measure similarity distance. Basically, it functions as follows:

- Input: The number of *k* and a database containing *n* objects.
- Output: A set of *k*-clusters that minimize the squared-error criterion.
- Method:
    1. arbitrarily choose *k* objects as the initial cluster centres;
    2. repeat;
    3. (re)assign each object to the cluster to which the object is the most similar based on the mean value of the objects in the cluster;
    4. update the cluster mean, i.e. calculate the mean value of the object for each cluster;
    5. until no change.

To start using the clustering method, it can be divided into two methods: hierarchical and non-hierarchical methods. One of the clustering approaches could be selected after analysis. In other words, the desired number of clusters, *k*, is specified in advance, and each of the cases is assigned to one of the *k*-clusters to minimize the variance of the clustering of the internal techniques. In the non-hierarchical approach, for creating good communities, *k* is defined in advance so that the measurement items are based on the homogeneity of the communities. They are not nested clusters; hierarchical clustering is used to divide the samples.

4.1.2 EM Clustering:

The concept of the EM algorithm stems from the Gaussian mixture model (GMM). The GMM method is one way to improve the density of a given set of sample data modelled as a function of the probability density of a single-density estimation method with multiple Gaussian probability density function to model the distribution of the data. In general, to obtain the estimated parameters of each Gaussian blend component if given a sample data set of the log-likelihood of the data, the maximum is determined by the EM algorithm to estimate the optimal model. Principally, the EM clustering method uses the following algorithm:

Input: Cluster number *k*, a database, stopping tolerance.

Output: A set of *k*-clusters with weight that maximize log-likelihood function.

1. Expectation step: For each database record *x*, compute the membership probability of *x* in each cluster $h = 1,\dots, k$.
2. Maximization step: Update mixture model parameter (probability weight).
3. Stopping criteria: If stopping criteria are satisfied stop, else set $j = j + 1$ and go to (1).

In the analytical methods available to achieve probability distribution parameters, in all probability the value of the variable is given. The iterative EM algorithm uses a random variable and, eventually, is a general method to find the optimal parameters of the hidden distribution function from the given data, when the data are incomplete or has missing values.[5,6]

4.2 Cluster Quality Measure

SSE: sum of the square error from the items of each cluster. The less the SSE is the better the cluster will be.
Inter cluster distance: sum of the square distance between each cluster centroid. The more the ICD is the better the cluster will be.

Intra cluster distance for each cluster: sum of the square distance from the items of each cluster to its centroid.

Maximum Radius: largest distance from an instance to its cluster centroid.

Quality	EM	GSM	IMM
SSE	38.47	45.87	53.03
Inter Cluster Distance	105.36	93.71	98.89

4.3 Comparative Analysis:

4.3.1 K-Means on Two-Dimensional, Two Gaussian Data

The K-Means algorithm works very well on this data set, effectively converging in three or four iterations (see figure 4):

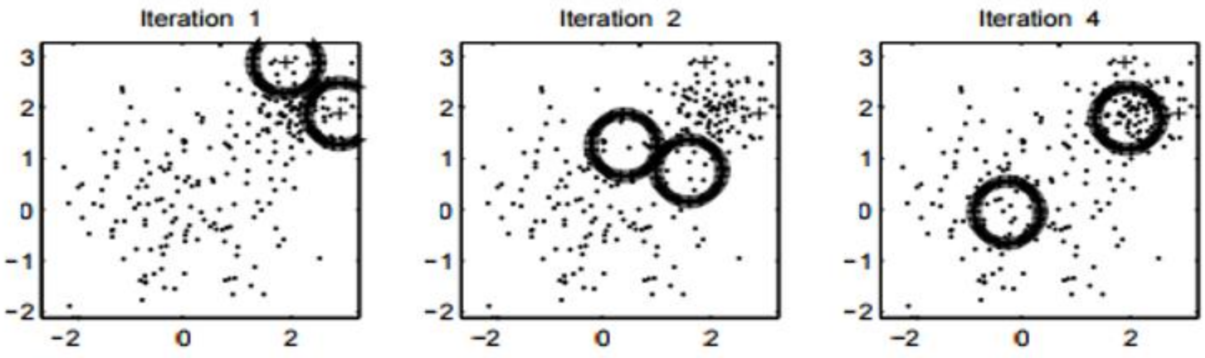


Figure 4.1: Process of K-means algorithm with k = 2

4.2.2 EM on Two-Dimensional, Two Gaussian Data

The EM algorithm also performs well, typically converging within 5 iterations (see figure 5).

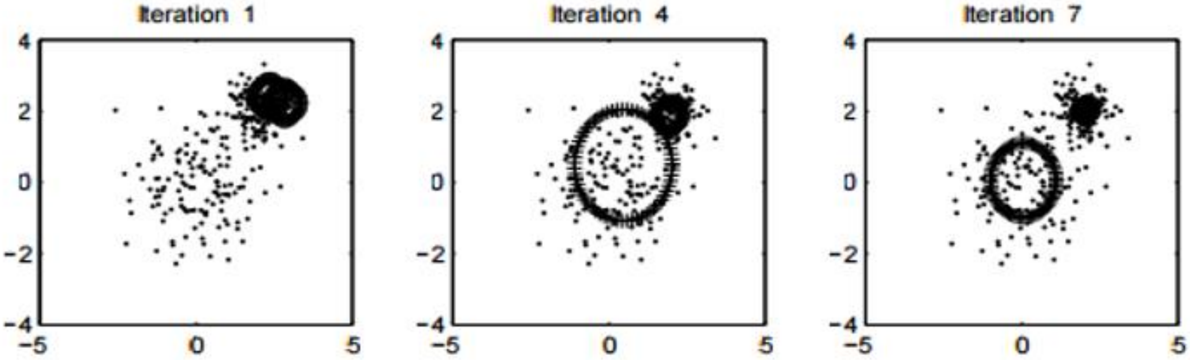


Figure 4.2: Process of EM algorithm with k = 2

Chapter 5

Conclusion

5.1 Summary of Contribution:

We from the report, concluding that this attempt of forming clusters without giving the parameter beforehand is successful to most of extent. The process time for forming clusters have shown its result. Here, the formation of clusters is can be varied. Initially, when we used K-means with EM found total number of clusters are four and after applied GMM with EM observed finally, three clusters (there two are specifying very similar so, they are merged into one cluster).So, that after refinement we got an finite clusters. Now, for the future enhancement of the results the Gaussian mixture model of expectation-maximization can be used. In the process frequency of each term in every document with its inverse document frequency is taken.

5.2 Limitations and Future works

In our report there are some limitations like finding convergence can be slow, Maximum likelihood Estimator (MLE) may overfit. In future we will label the clusters properly. Then we have to find the name of the bacteria where those cluster of nucleotides are present.

We also have to provide the percentage of bacteria which is present in a human sample. After that, we will take cluster nucleotides, match with reference genome of bacteria in HMPDAC, name the clusters with the bacteria title given in database. Then for healthy/ patient sample we will show the percentage of bacteria and infer that since this bacteria is higher, it might be reasonable for the problem.

References

[1] David Koslicki1,*, Simon Foucart2 and Gail Rosen3 1Mathematical Biosciences Institute, The Ohio State University, Columbus, OH 43201, USA and 2Department of Mathematics and 3Department of Electrical and Computer Engineering, Drexel University, Philadelphia, PA 19104, USA, Advance Access publication June 20, 2013.

[2] Genivaldo Gueiros Z. Silva, Daniel A. Cuevas, Bas E. Dutilh and Robert A. Edwards**,**Computational Science Research Center, San Diego State University, San Diego, CA, USA;Department of Computer Science, San Diego State University, San Diego, CA, USA, Accepted 21 May 2014 ,Published 5 June 2014

[3] Turnbaugh,P.J., Hamady,M., Yatsunenko,T., Cantarel,B.L.,Duncan,A., Ley,R.E., Sogin,M.L., Jones,W.J., Roe,B.A., Affourtit,J.P. et al. (2009) A core gut microbiome in obese and lean twins. Nature, 457, 480–484.

[4] N. Diaz, L. Krause, A. Goesmann, and et al., \TACOA - Taxonomic classification of environmental genomic fragments using a kernelized nearest neighbor approach," BMC Bioinformatics, vol. 10, no. 1, pp. 56+, 2009.

[5] S. D. Bentley and J. Parkhill, \Comparative genomic structure of prokaryotes," Annual Review of Genetics,vol. 38, pp. 771791, December 2004.

[6] Y.-W. Wu and Y. Ye, \A novel abundance-based algorithm for binning metagenomic sequences using l-tuples,"in Proceedings of the 14th annual international conference RECOMB'10, pp. 535{549, Springer, 2010.

[7] D. L. Wheeler, T. Barrett, D. A. Benson, and et al., \Database resources of the National Center for Biotechnology Information.," Nucleic Acids Research, vol. 35, January 2007.

[8] D. A. Benson, I. Karsch-Mizrachi, D. J. Lipman, and et al., \GenBank.," Nucleic acids research, vol. 37, pp. D26 31, January 2009.

[9] Qichao Tu1, Zhili He1,* and Jizhong Zhou1,2,3,* 1Department of Microbiology and Plant Biology, Institute for Environmental Genomics, University of Oklahoma, Norman, OK 73072, USA, 2Earth Science Division, Lawrence Berkeley National Laboratory, Berkeley, CA 94720, USA and 3State Key Joint Laboratory of Environmental Simulation and Pollution Control, School of Environment, Tsinghua University, Beijing 100084, China. Published online 12 February 2014

[10] Ley,R.E. (2010) Obesity and the human microbiome. Curr. Opin.Gastroenterol., 26, 5–11 .

[11] Larsen,N., Vogensen,F.K., van den Berg,F.W.J., Nielsen,D.S.,Andreasen,A.S., Pedersen,B.K., Al-Soud,W.A., Sørensen,S.J.,Hansen,L.H. and Jakobsen,M. (2010) Gut microbiota in human adults with type 2 diabetes differs from non-diabetic adults. PLoS One, 5, e9085.

[12] Qin,J., Li,Y., Cai,Z., Li,S., Zhu,J., Zhang,F., Liang,S., Zhang,W.,Guan,Y., Shen,D. et al. (2012) A metagenome-wide association study of gut microbiota in type 2 diabetes. Nature,490, 55–60.

[13] M. Wendl and R. Waterston, \Generalized gap model for bacterial articial chromosome clone ngerprint mapping and shotgun sequencing," Genome Res, vol. 12, no. 1, p. 19431949, 2002.

[14] X. Li and M. S. Waterman, \Estimating the Repeat Structure and Length of DNA Sequences Using l-Tuples,"Genome Research, vol. 13, pp. 1916{1922, August 2003.

[15] Karlsson,F.H., Tremaroli,V., Nookaew,I., Bergstrom,G.,Behre,C.J., Fagerberg,B., Nielsen,J. and Backhed,F. (2013) Gut metagenome in European women with normal, impaired and diabetic glucose control. Nature, 498, 99–103.

[16] Kau,A.L., Ahern,P.P., Griffin,N.W., Goodman,A.L. and Gordon,J.I. (2011) Human nutrition, the gut microbiome and the immune system. Nature, 474, 327–336.

[17] Schwabe,R.F. and Jobin,C. (2013) The microbiome and cancer. Nat. Rev. Cancer, 13, 800–812.

[18] S. van Dongen, Graph Clustering by Flow Simulation. PhD thesis, University of Utrecht, May 2000.

[19] D. Wu, S. C. Daugherty, S. E. Van Aken, and et al., \Metabolic Complementarity and Genomics of the Dual Bacterial Symbiosis of Sharpshooters," PLoS Biol, vol. 4, pp. e188+, June 2006.

[20] D. C. Richter, F. Ott, A. F. Auch, and et al., \MetaSim: a Sequencing Simulator for Genomics and Metagenomics,"PLoS ONE, vol. 3, pp. e3373+, October 2008