



TRAINING MANUAL

DBT Funded Training Program

BIOINFORMATICS FOR METAGENOME DATA ANALYSIS

(NGSDAT 2019)

March 19-22, 2019



Bioinformatics Centre

Indian Institute of Spices Research

Kozhikode, Kerala

Department of Genomic Science, Central University of Kerala,

Kasaragod, Kerala

Published by

Dr. Nirmal Babu

Director, Indian Institute of Spices Research, Kozhikode, Kerala.

Organized by

Dr. Santhosh J Eapen, Co-ordinator

Dr. Prasath, Co-ordinator

The manual is an in-house publication intended for training purposes only and is not for public circulation.

Copyright©2019 IISR. All rights reserved.

Reproduction and Redistribution is prohibited without approval.

CONTENTS

S.NO		Page
1	METAGENOMICS–THE NGS TOOL FOR MICROBIOME RESEARCH	1
2	A TAXONOMIST’S BRUSH WITH METAGENOMICS	4
3	NEXT GENERATION SEQUENCING CHEMISTRY AND PLATFORMS	6
4	RHIZOSPHERE SOIL METAGENOMICS – A CASE STUDY IN BLACK PEPPER	19
5	BASIC LINUX COMMANDS	25
6	NGS DATA FORMATS AND QUALITY CHECK	30
7	GENERAL WORKFLOW FOR METAGENOME DATA ANALYSIS	50
8	MOTHUR	56
9	QIIME	71
10	TOOL LINKS	73



DBT Funded Training Program -Bioinformatics for Metagenome data analysis

March 19-22, 2019

Bioinformatics Centre, ICAR-Indian Institute of Spices Research, Kozhikode, Kerala
Department of Genomic Science, Central University of Kerala, Kasaragod, Kerala

Schedule

Day 1: Tuesday, 19 March 2019

09:30 am – 10:00 am	Registration	
10:00 am – 10.30 am	Welcome Inaugural session and workshop background	Chief Guest Dr. Nirmal Babu Director – ICAR-Indian Institute of Spices Research.
10:30 am – 11:00 am	Pre evaluation & group photo	
11.00 am – 11.45 pm	Lecture: Metagenomics: An NGS tool for exploring the microbiomes	Dr. Santhosh J Eapen Principal Scientist and Head, Division of Crop Protection & Co-ordinator (Bioinformatics) ICAR-IISR
11:45 am – 12.45 pm	Lecture: Next Generation Sequencing-Chemistry and Platforms	Dr. A. Ishwara Bhat Principal Scientist, Plant Pathology, ICAR-Indian Institute of Spices Research.
12:45 pm – 1.30 pm	Lecture: Metagenome analysis	Dr. Tony Grace Assistant Professor, Department of Genetics, Central University of Kerala
01:30 pm - 02:30 pm	Lunch break	
02:30pm -03:30pm	(Lecture & Practicals) Introduction to the Linux system; command lines for data analysis	Ms. Neethu Isaac Junior Research Fellow, ICAR-IISR
3.30 pm – 3.45 pm	Tea/Coffee	
3.45 pm – 5.00 pm	(Lecture & Practicals) Introduction to various data formats, Quality Control, Trimming, Filtering of Low Quality Reads.	Mrs. Blessy M. Baby Ph.D. Scholar, ICAR-IISR

Day 2: Wednesday, 20 March 2019

09:30 am – 10.00 am	Lecture: General workflow of Metagenome data analysis	Dr. Muhammed Manzoor A.P. Research Associate, ICAR-IISR
---------------------	--	---

10:00 am – 10.15 am	Setting up your computers	
10.15 am to 11:00 am	Introduction to MOTHUR & practicals	Dr. Muhammed Manzoor A.P. Research Associate, ICAR-IISR Dr. Sudeep Ghate Post Doctoral Fellow, Yenepoya University
11.00 am- 11: 20 pm	Tea/Coffee	
11.20 am -1.00 pm	MOTHUR contd.	
01:00 pm - 02:00 pm	Lunch Break	
02:00 pm - 03:30 pm	MOTHUR contd.	
3.30 pm – 03.45 pm	Tea/Coffee	
03.45 pm – 04.30 pm	MOTHUR contd.	

Day 3: Thursday, 21 March 2019

09:30 am – 10:30 am	MOTHUR: General statistical data analysis	
10: 30 am – 10:45 am	Tea break	
10.45 am-12.00 pm	Data analysis using Microbiome Analyst	
12.00 pm- 1: 00 pm	Data analysis using other offline/online tool	
01:00 pm - 02:00 pm	Lunch Break	
3.15 pm – 5:00 pm	Practical's on QIIME	Mr. Kumar Arvind. Ph.D student, Central University of Kerala

Day 4: Friday, 22 March 2019

09:30 am – 10:30 am	Lecture Rhizosphere soil metagenomics: A case study in black pepper	Dr. P. Uma Devi Scientist, Biotechnology, ICAR-IISR.
10.30 am to 11.15 am	Practical's on QIIME	
11:15 am – 11:30 am	Tea break	
11: 20 am – 1:00 pm	Practicals on QIIME	
01:00 pm - 02:00 pm	Lunch Break	
02:00 pm-03:30 pm	Lecture A taxonomist's brush with metagenomics Valedictory session, Post evaluation, Feedback submission and certificate distribution	Chief Guest: Dr. Belle Damodara Shenoy Scientist, CSIR-National Institute of Oceanography, Visakhapatnam.

1. METAGENOMICS – THE NGS TOOL FOR MICROBIOME RESEARCH

Dr. Santhosh J. Eapen

Microbes are ubiquitous in nature, inhabiting almost every conceivable environment, and play an important role in human life. The invisible microbes play an essential role in ecosystem functioning, modulating key ecosystem processes such as plant growth, soil nutrient cycling, and marine biogeochemical cycling. They conduct significant functions that greatly benefit the health of planet, as well as its inhabitants. A functioning microbiome (= meaning all the microbes in a community) is obligatory for host organisms, as it contributes to the smooth functioning of important physiological processes. In fact, host organisms have co-evolved with their microbiota and by analysing host-microbe interactions, the core characteristics of the interaction, including their identification, classification, profile prediction, and mechanisms of interaction can be understood. Classic approaches to microbial ecology have relied on cultivation-dependent techniques to study host-microbe interactions. Recently, however, a number of culture-independent techniques, mainly PCR-based methods, have evolved for the qualitative and quantitative identification of microbes.

What is metagenomics?

Metagenomics is the study of microbial communities via DNA extracted from the whole population rather than looking at the individual organisms within it. Metagenomics is expected to play a major role in advancing our understanding of microbes and microbial communities in environments that are challenging to study, including in soil or the human gut. Metagenomics is providing new and unparalleled insights into microbial communities, advancing our knowledge of microbial ecology and microbiology in general, uncovering a wealth of new biodiversity, and discovering new microbial species leading to an expansion of the tree of life. Meta-omics has also promoted collaborative efforts in a grand vision across the international research community, as exemplified by the Earth Microbiome Project (EMP) and Human Microbiome Project (HMP).

Role of NGS in metagenomics

The advent of Next-Generation Sequencing (NGS) technologies has led to a substantial increase in the number of metagenomic studies during the last ten years. Indeed, NGS-based methods have provided the capability of simultaneously analysing communities of microorganisms and has literally removed the limitations and boundaries associated with classic culture-based approaches. These advances now provide a fast, cost-effective approach to delivering large volumes of highly-accurate data. The advantages of NGS over classic Sanger sequencing are that it is cost-effective, devoid of a cloning step, offers high throughput, and requires minimal technical expertise. Several different NGS platforms like the Roche 454 GS FLX, Illumina (MiSeq and HiSeq), Ion Torrent/Ion Proton/Ion Proton, SOLiD 5500 series, and Oxford Nanopore have been developed and are commonly used. These are discussed in detail in the next lecture. There are different approaches for metagenomic sequencing:

Bacterial identification by 16S rRNA gene sequencing: The molecular-based taxonomic investigation for bacteria employs direct sequencing of PCR-amplified small sequences of 16S rRNA gene from extracted DNA, generally using universal primers annealing conserved nucleotides to amplify one or more fragments of variable regions. The sequences at a pre-defined level of identity stand for grouped clusters of similar sequencing reads, known as Operational Taxonomic Unit (OTU), which corresponds to a group of very similar 16S sequences. Reference databases (GreenGenes, myRDP, NCBI) are used to classify OTUs providing identification of taxonomy, relative frequencies and diversity of community composition in samples obtained. This approach allows identification of new species and investigation of low-abundance and uncultivated bacteria from a single analysis. In addition, they are faster and more accurate compared to classical identification methods (cloning and culture). However, this approach has some limitations regarding information about the microbiome function, mainly because several species of bacteria have not been characterized yet and secondly due to a great variability found among individuals.

Shotgun sequencing: In this method, multiple continuous overlapping sequences (contigs), which are assembled from fragmented sequences and obtained from total purified genomic

DNA, are used for identifying genes through alignment with bacterial reference genomes and databases (KEGG, SEED and NCBI). Shotgun approach is a versatile approach allowing Functional characterization in complex environments like the human gut. This method also provides identification of variants and polymorphisms and gives a more comprehensive understanding on the functional information of microorganism communities. A major limitation of this strategy is that metagenomic sequencing of multiple individuals is extremely expensive in comparison with 16S rRNA sequencing and generation of a large amount of data demands intense computational analysis, most of time to be performed by bioinformatics specialists.

Metagenomics analysis

But the major challenge with NGS data, however, is the analysis of millions of sequences that allows one to achieve statistically and scientifically meaningful conclusions. Bioinformatic tools used for metagenomic analysis, especially for translating raw sequences into meaningful data, are continually developing with the aim of providing the ability to examine both the taxonomic and the functional composition of diverse metagenomes. Today a number of specialized software programs are available for analysing the metagenomic data (Please see Annexure 1 for a list). The basic steps that are followed in the analysis pipeline starting from pre-processing of the sequencing data to the final extraction, storage, and presentation of the data are more or less similar and will be discussed by subsequent speakers.

Conclusion

The integration of several approaches to biological studies will be necessary to answer questions about the diversity and ecology of microbial flora. And we require better bioinformatic tools and expertise for analysing metagenomic data. Metagenomics will play an increasing role in the fields of medicine, biotechnology, and environmental science.

2. A TAXONOMIST'S BRUSH WITH METAGENOMICS

Dr. Belle Damodara Shenoy

Taxonomy is a science which deals with the classification of organisms based on evolutionary traits. The incorporation of DNA sequence information in taxonomic processes (identification, correlation and classification) has revolutionized the microbial taxonomy. In case of fungi, the classical groups of fungi and fungi-like organisms have now been replaced with a phylogeny-centric delimitation of true fungi, resulting in the reclassification of traditional lower fungi, and in some extreme cases discontinuation of well-known taxa such as Zygomycota. The old practice of dual nomenclature, in which asexual and sexual fungi used to be classified separately, has been discontinued thanks to the unifying feature of DNA sequence-data.

Traditional morphotaxonomic characters are proven to be phylogenetically uninformative and misleading in many fungal groups. Several morphologically-similar asexual genera, including *Diplococcium-Spadicoides* and *Sporidesmiumsensulato*, are polyphyletic, thus their taxonomy is being revisited. Conversely, cryptic species have been detected in many fungal genera, resulting in description of morphotaxonomically-indistinguishable, phylogenetically-distinct novel species. For example, *Colletotrichumgloeosporioidessensulato*, a globally documented taxon, has now been redefined to include *C. gloeosporioidessensulato* and morphologically-similar 25 distinct phylogenetic species. There is a paradigm shift in the way the modern fungal taxonomy is being practiced.

Microbial alpha-taxonomy is undergoing rapid changes, also due to the development of culture-independent metagenomic approach. There is a limit to which microbes can be cultured on artificial media. Metagenomic tools have become popular in unravelling novel insights into microbial community structure and possible ecosystem functioning. For example, culture-based studies on bacteria associated with tarballs (marine pollutants) from Goon beaches have so far revealed 30 bacterial genera (Shinde et al. 2017). In comparison, recent metagenomic analysis by Fernandes et al. (2019) has revealed the presence of around 250 bacterial genera in tarball samples collected from Goon beaches. This included several genera previously reported for hydrocarbon degradation and pathogenic behaviour.

The metagenomic approach, however, cannot be substituted for a comprehensive and robust taxonomic analysis of microbes of our interest. The combination of metagenomics and polyphasic taxonomy is needed for the development of a high-value, more predictive microbial taxonomy.

NGSDAT '19

3. NEXT GENERATION SEQUENCING CHEMISTRY AND PLATFORMS

Dr. A.I. Bhat

NGS technologies have demonstrated the capacity to sequence DNA at unprecedented speed, thereby enabling previously unimaginable scientific achievements and novel biological applications. But, the massive data produced by NGS also presents a significant challenge for data storage, analyses, and management solutions. Advanced bioinformatics tools are essential for the successful application of NGS technology. NGS technologies will have a striking impact on genomic research and the entire biological field. With its ability to tackle the unsolved challenges unconquered by previous genomic technologies, NGS is likely to unravel the complexity of the genomes in terms of genetic variations, some of which may be confined to susceptible loci for some common conditions. Basic concepts, general applications, and the potential impact of next-generation sequencing (NGS) technologies on genomics, with particular reference to currently available and possible future platforms are discussed here.

Since the time DNA was discovered as the code to all biological life on earth, man has sought to unravel its mysteries. The “original” sequencing methodology, known as Sanger chemistry, uses specifically labelled nucleotides to read through a DNA template during DNA synthesis. This sequencing technology requires a specific primer to start the read at a specific location along the DNA template, and record the different labels for each nucleotide within the sequence. After a series of technical innovations, the Sanger method has reached the capacity to read through 1000–1200 basepair (bp); however, it still cannot surpass 2 kilo basepair (Kbp) beyond the specific sequencing primer.

In order to sequence longer sections of DNA, a new approach called shotgun sequencing was developed during Human Genome Project (HGP). In this approach, genomic DNA is enzymatically or mechanically broken down into smaller fragments and cloned into sequencing vectors in which cloned DNA fragments can be sequenced individually. The complete sequence of a long DNA fragment can be eventually generated by these method by alignment and reassembly of sequence fragments based on partial sequence overlaps.

Shotgun sequencing was a significant advantage from HGP, and made sequencing the entire human genome possible. The core philosophy of massive parallel sequencing used in next-generation sequencing (NGS) is adapted from shotgun sequencing.

New NGS technologies read the DNA templates randomly along the entire genome. This is accomplished by breaking the entire genome into small pieces, then ligating those small pieces of DNA to designated adapters for random read during DNA synthesis (sequencing- by-synthesis). Therefore, NGS technology is often called massively parallel sequencing.

The read length (the actual number of continuous sequenced bases) for NGS is much shorter than that attained by Sanger sequencing. At present, NGS only provides 50–500 continuous basepair reads, which is why sequencing results are defined as short reads. These short reads are a major limitation in current technology; however, developing NGS technologies, such as single-molecule sequencing, may surpass Sanger methodologies and have the potential to read several continuous kilo base pairs (Kbps) (Table 1). Since next-generation technologies currently produce short reads, coverage is a very important issue. Coverage is defined as the number of short reads that overlap each other within a specific genomic region. For example, a 30-fold coverage for a gene means that every nucleotide within this gene region is represented in at least 30 distinct and overlapping short reads. Sufficient coverage is critical for accurate assembly of the genomic sequence. In addition to the need for adequate coverage, short reads create many sequences that cannot be interpreted or “mapped” to any reference DNA or be accurately assembled. This is simply because some of the short reads are too short and may match with many different regions of the genome and are not unique to any specific region of the sequence. Short-read sequences that can be assembled and matched with a reference sequence are generally called “mappable reads”.

Next-generation sequencing (NGS) is also known as massively parallel sequencing or multiplex cyclic sequencing. Previously, DNA sequencing was performed almost exclusively by the Sanger method, which has excellent accuracy and reasonable read length but very low throughput. Sanger sequencing was used to obtain the first consensus sequence of the human genome in 2001 and the first individual human diploid sequence. Shortly thereafter,

the second complete individual genome was sequenced using next-generation technology, which marked the first human genome sequenced with new NGS technology. Since then, several additional diploid human genomes have been sequenced with NGS utilizing a variety of related techniques to rapidly sequence genomes with varying degrees of coverage. A common strategy for NGS is to use DNA synthesis or ligation process to read through many different DNA templates in parallel. Therefore, NGS reads DNA templates in a highly parallel manner to generate massive amounts of sequencing data but, as mentioned above, the read length for each DNA template is relatively short (35–500 bp) compared to traditional Sanger sequencing (1000–1200 bp).

Several NGS methods recently developed allow larger-scale DNA sequencing. The number of large short-read sequences from NGS is increasing at exponential rates. Currently, five NGS platforms are commercially available, including the Roche GS-FLX 454 Genome Sequencer (originally 454 sequencing), the Illumina Genome Analyzer (originally Solexa technology), the ABI SOLiD analyzer, the Helicos HeliScope and nanopore platforms. These NGS instruments generate different base read lengths, different error rates, and different error profiles relative to Sanger sequencing data and to each other. NGS technologies have increased the speed and throughput capacities of DNA sequencing and, as a result, dramatically reduced overall sequencing costs.

History of DNA Sequencing

First- and second-generation sequencing technologies have led the way in revolutionizing the field of genomics and beyond, motivating an astonishing number of scientific advances, including enabling a more complete understanding of whole genome sequences and the information encoded therein, a more complete characterization of the methylome and transcriptome and a better understanding of interactions between proteins and DNA. Nevertheless, there are sequencing applications and aspects of genome biology that are presently beyond the reach of current sequencing technologies, leaving fertile ground for additional innovation in this space. A new generation of single-molecule sequencing technologies (third generation sequencing) that is emerging to fill this space, with the potential for dramatically longer read lengths, shorter time to result and lower overall cost.

First-generation sequencing

First-generation sequencing was originally developed by Sanger in 1975 (the chain-termination method) and in parallel by Maxam and Gilbert in 1977 (a chemical sequencing method). From these first-generation methods, Sanger sequencing ultimately prevailed given it was less technically complex and more amenable to being scaled up. For Sanger sequencing practiced today, during sample preparation, different-sized fragments of DNA are generated each starting from the same location. Each fragment ends with a particular base that is labelled with one of four fluorescent dyes corresponding to that particular base. Then all of the fragments are distributed in the order of their length via capillary electrophoresis. Information regarding the last base is used to determine the original sequence. This method results in a read length that is ~800 bases on average, but may be extended to above 1000 bases. While fully automated implementations of this approach were the mainstay for the original sequencing of the human genome, their chief limitation was the small amounts of DNA that could be processed per unit time, referred to as throughput, as well as high cost, resulting in it taking roughly 10 years and three billion dollars to sequence the first human genome.

Second-generation sequencing

Commercial SGS tools emerged in 2005 in response to the low throughput and high cost of first-generation methods. To address this problem, SGS tools achieve much higher throughput by sequencing a large number of DNA molecules in parallel. With most SGS technologies, tens of thousands of identical strands are anchored to a given location to be read in a process consisting of successive washing and scanning operations. The 'wash-and-scan' sequencing process involves sequentially flooding in reagents, such as labelled nucleotides, incorporating nucleotides into the DNA strands, stopping the incorporation reaction, washing out the excess reagent, scanning to identify the incorporated bases and finally treating the newly incorporated bases to prepare the DNA templates for the next 'wash-and-scan' cycle. This cycle is repeated until the reaction is no longer viable. The array of DNA anchor locations can have a very high density of DNA fragments, leading to extremely high overall throughput and a resultant low cost per identified base when such instruments are run at high capacity. For example, Illumina's HiSeq 2000 instrument can

generate upwards of 300 or more gigabases of sequence data in a single run. The time-to-result for these SGS methods is generally long (typically taking many days), due to the large number of scanning and washing cycles required. Furthermore, because step yields for the addition of each base are, 100%, a population of molecules becomes more asynchronous as each base is added. This loss of synchronicity (called dephasing) causes an increase in noise and sequencing errors as the read extends, effectively limiting the read length produced by the most widely used SGS systems to significantly less than the average read lengths achieved by Sanger sequencing.

Further, in order to generate this large number of DNA molecules, PCR amplification is required. The amplification process can introduce errors in the template sequence as well as amplification bias. The effects of these pathologies are that neither the sequences nor the frequencies with which they appear are always faithfully preserved. In addition, the process of amplification increases the complexity and time associated with sample preparation. Finally, the massively high throughput achieved by SGS technologies per run generates mountains of highly informative data that challenge data storage and informatics operations, especially in light of the shorter reads (compared with Sanger sequencing) that make alignment and assembly processes challenging. First-generation sequencing and SGS technologies have led the way in revolutionizing the field of genomics and beyond, motivating an astonishing number of scientific advances. Nevertheless there are sequencing applications and aspects of genome biology that are presently beyond the reach of current sequencing technologies, leaving fertile ground for additional innovation in this space.

Next-generation sequencing platforms

Among the five commercially available platforms, the Roche/454 FLX, the Illumina/Solexa Genome Analyzer, and the Applied Biosystems (ABI) SOLiD Analyzer are currently dominating the market. The other two platforms, the Polonator G.007 and the Helicos HeliScope, have just recently been introduced and are not widely used. Additional platforms from other manufacturers are likely to become available within the next few years and bring. Methodologies used by each of the current available NGS systems is discussed below.

(i) Roche GS-FLX 454 Genome Sequencer

The Roche GS-FLX 454 Genome Sequencer was the first commercial platform introduced in 2004 as the 454 Sequencer. The second complete genome of an individual was sequenced with this platform). The 454 Genome Sequencer uses sequencing-by-synthesis technology known as pyrosequencing. The key procedure in this approach is emulsion PCR in which single-stranded DNA binding beads are encapsulated by vigorous vortexing into aqueous micelles containing PCR reactants surrounded by oil for emulsion PCR amplification. During the pyrosequencing process, light emitted from phosphate molecules during nucleotide incorporation is recorded as the polymerase synthesizes the DNA strand. Initially, the 454 Sequencer had a read length of 100 bp but now can produce an average read length of 400 bp. The maximum ~600 bp capacity of 454 systems approaches the halfway of current Sanger sequencing capacities (~1200 bp). At 600 bp, the 454 Sequencer has the longest short reads among all the NGS platforms; and generates ~400–600 Mb of sequence reads per run; critical for some applications such as RNA isoform identification in RNA-seq and de novo assembly of microbes in metagenomics. Raw base accuracy reported by Roche is very good (over 99%); however, the reported relatively error-prone raw data sequence, especially associated with insertion-deletions, is a major concern. Low yield of sequence reads could translate into a much higher cost if additional coverage is needed to define a genetic mutation.

(ii) Illumina/Solexa Genome Analyzer

The Illumina/Solexa Genome Analyzer was the second platform to reach market, and currently is the most widely used system. The Illumina platform uses sequencing-by-synthesis approach in which all four nucleotides are added simultaneously into oligo-primed cluster fragments in flow-cell channels along with DNA polymerase. Bridge amplification extends cluster strands with all four fluorescently labelled nucleotides for sequencing. The Genome Analyzer is widely recognized as the most adaptable and easiest to use sequencing platform. Superior data quality and proper read lengths have made it the system of choice for many genome sequencing projects. To date, the majority of published NGS papers have described methods using the short sequence data produced with the Genome Analyzer. At present, the new Illumina HiSeq 2000 Genome Analyzer is capable of producing single reads

of 2×100 basepairs (pair-end reads), and generates about 200 giga basepair (Gbp) of short sequences per run. The raw base accuracy is greater than 99.5%.

(iii) Ion Torrent

Ion Torrent developed an entirely new approach to sequencing based on the well-characterized biochemistry that when a nucleotide is incorporated into a strand of DNA by a polymerase, a hydrogen ion is released as a by-product. They have developed an ion sensor that can detect hydrogen ions and directly convert the chemical information to digital sequence information. In essence, their NGS platform can be defined as the world's smallest solid-state pH meter.

Transitioning from SGS to TGS

The genomics community has been enormously enabled by first- and second-generation sequencing (SGS) technologies in comprehensively characterizing DNA sequence variation, de novo sequencing of a number of species, sequencing of microbiomes, detecting methylated regions of the genome, quantitating transcript abundances, characterizing different isoforms of genes present in a given sample and identifying the degree to which mRNA transcripts are being actively translated. One of the hallmark features of the SGS technologies is their massive throughput at a modest cost, with hundreds of gigabases of sequencing now possible in a single run for several thousand dollars. Despite the recent and rapid acceptance of SGS technologies, a new generation of single-molecule sequencing (SMS) technologies is emerging. Unlike major SGS sequencing by synthesis (SBS) technologies that rely on PCR to grow clusters of a given DNA template, attaching the clusters of DNA templates to a solid surface that is then imaged as the clusters are sequenced by synthesis in a phased approach, the new generation of SBS technologies interrogate single molecules of DNA, such that no synchronization is required (a limitation of SGS), thereby overcoming issues related to the biases introduced by PCR amplification and dephasing. More importantly, this new generation of sequencing technologies has the potential to exploit more fully the high catalytic rates and high processivity of DNA polymerase or avoid any biology or chemistry altogether to radically increase read length (from tens of bases to tens of thousands of bases per read) and time to result (from days to hours or minutes). The promises then of this new, third generation of sequencing

technologies in offering advantages over current sequencing technologies are (i) higher throughput; (ii) faster turnaround time (e.g. sequencing metazoan genomes at high fold coverage in minutes); (iii) longer read lengths to enhance de novo assembly and enable direct detection of haplotypes and even whole chromosome phasing; (iv) higher consensus accuracy to enable rare variant detection; (v) small amounts of starting material (theoretically only a single molecule may be required for sequencing); and (vi) low cost.

There may not yet be consensus on what constitutes a third generation, or next-next-generation sequencing instrument, given advances are being made on rapid time scales that do not easily fit into generational time scales. However, SMS without the need to halt between read steps (whether enzymatic or otherwise), where reads from SMS instruments represent sequencing of a single molecule of DNA. SMS technologies that do not purposefully pause sequencing reaction after each base incorporation represent the most thoroughly explored TGS approaches in hopes of increasing sequencing rates, throughput and read lengths, lowering the complexity of sample preparation and ultimately decreasing cost. However, as a result of using these criteria to define TGS, a number of exciting technologies do not fit neatly into this definition, but are nevertheless exciting in terms of how they complement current SGS technologies.

Table 1. Comparison of First generation, second generation and third generation sequencing techniques

	First generation	Second generation ^a	Third generation ^a
Fundamental technology	Size-separation of specifically end-labeled DNA fragments, produced by SBS or degradation	Wash-and-scan SBS	SBS, by degradation, or direct physical inspection of the DNA molecule
Resolution	Averaged across many copies of the DNA molecule being sequenced	Averaged across many copies of the DNA molecule being sequenced	Single-molecule resolution
Current raw read accuracy	High	High	Moderate
Current read length	Moderate (800–1000 bp)	Short, generally much shorter than Sanger sequencing	Long, 1000 bp and longer in commercial systems
Current throughput	Low	High	Moderate
Current cost	High cost per base Low cost per run	Low cost per base High cost per run	Low-to-moderate cost per base Low cost per run
RNA-sequencing method	cDNA sequencing	cDNA sequencing	Direct RNA sequencing and cDNA sequencing
Time from start of sequencing reaction to result	Hours	Days	Hours
Sample preparation	Moderately complex, PCR amplification not required	Complex, PCR amplification required	Ranges from complex to very simple depending on technology
Data analysis	Routine	Complex because of large data volumes and because short reads complicate assembly and alignment algorithms	Complex because of large data volumes and because technologies yield new types of information and new signal processing challenges
Primary results	Base calls with quality values	Base calls with quality values	Base calls with quality values, potentially other base information such as kinetics

Next- Next-generation sequencing platforms (Third Generation Sequencing)

SMS technologies can roughly be binned into three different categories: (i) SBS technologies in which single molecules of DNA polymerase are observed as they synthesize a single molecule of DNA; (ii) nanopore-sequencing technologies in which single molecules of DNA are threaded through a nanopore or positioned in the vicinity of a nanopore, and individual bases are detected as they pass through the nanopore; (iii) direct imaging of individual DNA molecules using advanced microscopy techniques etc. Each of these technologies provides novel approaches to sequencing DNA and has advantages and disadvantages with respect to specific applications. If the full potential of these technologies is realized, in several years' time, whole genome sequencing will likely be fast enough and inexpensive enough to re-sequence genomes as needed for any application. Since single DNA molecule sequencing technology can read through DNA templates in real time without amplification, it provides accurate sequencing data with potentially long-reads and efforts have focused recently in this new direction. Several unique single- molecule DNA sequencing technologies are currently under development; however, little information has been made publically available.

(i) Single molecule sequencing (SMS) by synthesis

Single-molecule real-time sequencing: The single-molecule real-time (SMRT) sequencing approach developed by Pacific Biosciences is the first TGS approach to directly observe a single molecule of DNA polymerase as it synthesizes a strand of DNA, directly leveraging the speed and processivity of this enzyme to address many of the shortcomings of SGS. Given that a single DNA polymerase molecule is of the order of 10 nm in diameter, two important obstacles needed to be overcome to enable direct observation of DNA synthesis as it occurs in real time are: (i) confining the enzyme to an observation volume that was small enough to achieve the signal-to-noise ratio needed to accurately call bases as they were incorporated into the template of interest; and (ii) labeling the nucleotides to be incorporated in the synthesis process such that the dye–nucleotide linker is cleaved after completion of the incorporation process so that a natural strand of DNA remains for continued synthesis and so that multiple dyes are not held in the confinement volume at a time (something that

would destroy the signal-to-noise ratio). The problem of observing a DNA polymerase working in real time, detecting the incorporation of a single nucleotide taken from a large pool of potential nucleotides during DNA synthesis, was solved using zero-mode waveguide (ZMW) technology.

(ii) Nano-technologies for single-molecule sequencing

Thousands of nano-tunnels on a chip can be used to monitor the movement of a polymerase molecule on a single DNA strand during replication to perform single-molecule DNA sequencing-by-synthesis. Oxford Nanopore Technologies has developed nanopore-based DNA and RNA sequencing technology. Protein nanopores are tiny holes that in nature form gateways across membranes. In this technology, protein nanopores are embedded into a synthetic membrane bathed in an electrophysiological solution and an ionic current is passed through the nanopores. As molecules such as DNA or RNA move through the nanopores, they cause disruption in the current. This signal can be analysed in real-time to determine the sequence of bases in the strands of DNA or RNA passing through the pore.

(iii) Electronic detection for single-molecule sequencing

Reveo is developing a technology to stretch out DNA molecules on conductive surfaces for electronic base detection. A stretched and immobilized strand of DNA will be read through by multiple nano-knife edge probes. Each nano-knife edge probe specifically recognizes only one nucleotide for single-molecule sequencing. Intelligent Biosystems is also developing a platform using the electronic detection approach which will allow for high speed and high sensitivity single-molecule analysis with decreased background noise.

(iv) Electron microscopy for single-molecule sequencing

Electron microscopy (EM) was the first proposed and attempted approach to sequence DNA molecules before the Sanger sequencing was established and this concept has recently been reevaluated with the emergence of new technologies. Since scanning tunnelling microscopy (STM) can reach atomic resolution, STM for single-molecule sequencing is being explored. Light Speed Genomics is developing a microparticle approach by capturing sequence data with optical detection technology and new sequencing chemistry from a large field of view

to reduce the time consuming sample and detector rearrangement. Halcyon Molecular is developing a DNA sequencing technology by atom-by-atom identification and EM analysis. The key advantage of this technology is very long read lengths. ZS Genetics is also developing EM-based technologies for single-molecule DNA sequencing.

Other approaches for single-molecule sequencing

Focusing on resequencing specific sections of the human genome combined with genome-region enrichment, GenizonBioSciences is developing a sequencing-by-hybridization technology based on known reference sequences. Avantome (acquired by Illumina) is also exploring the single-molecule sequencing technologies. Detailed information of the NGS technologies and platform discussed above is summarized in Table 2.

Table 2. Platforms and detailed information on NGS technologies

Technology	Amplification	Read length	Throughput	Sequence by synthesis
<i>Currently available</i>				
Roche/GS-FLX Titanium	Emulsion PCR	400–600 bp	500 Mbp/run	Pyrosequencing
Illumina/HiSeq 2000, HiScan	Bridge PCR (Cluster PCR)	2 × 100 bp	200 Gbp/run	Reversible terminators
ABI/SOLiD 5500xl	Emulsion PCR	50–100 bp	>100 Gbp/run	Sequencing-by-ligation (octamers)
Polonator/G.007	Emulsion PCR	26 bp	8–10 Gbp/run	Sequencing-by-ligation (monomers)
Helicos/Helioscope	No	35 (25–55) bp	21–37 Gbp/run	True single-molecule sequencing (tSMS)
<i>In development</i>				
Pacific BioSciences/RS	No	1000 bp	N/A	Single-molecule real time (SMRT)
Visigen Biotechnologies	No	>100 Kbp	N/A	Base-specific FRET
U.S. Genomics	No	N/A	N/A	Single-molecule mapping
Genovox	No	N/A	N/A	Single-molecule sequencing by synthesis
Oxford Nanopore Technologies	No	35 bp	N/A	Nanopores/exonuclease-coupled
NABsys	No	N/A	N/A	Nanopores
Electronic BioSciences	No	N/A	N/A	Nanopores
BioNanomatrix/nanoAnalyzer	No	400 Kbp	N/A	Nanochannel arrays
GE Global Research	No	N/A	N/A	Closed complex/nanoparticle
IBM	No	N/A	N/A	Nanopores
LingVitae	No	N/A	N/A	Nanopores
Complete Genomics	No	70 bp	N/A	DNA nanoball arrays
base4innovation	No	N/A	N/A	Nanostructure arrays
CrackerBio	No	N/A	N/A	Nanowells
Reveo	No	N/A	N/A	Nano-knife edge
Intelligent BioSystems	No	N/A	N/A	Electronics
LightSpeed Genomics	No	N/A	N/A	Direct-read sequencing by EM
Halcyon Molecular	No	N/A	N/A	Direct-read sequencing by EM
ZS Genetics	No	N/A	N/A	Direct-read sequencing by TEM
Ion Torrent/PostLight	No	N/A	N/A	Semiconductor-based pH sequencing
Genizon BioSciences/CGA	No	N/A	N/A	Sequencing-by-hybridization

References:

1. Ansorge WJ. Next-generation DNA sequencing techniques. *Nat. Biotechnol.* 2009; 25:195–203.
2. Chistoserdova L. Recent progress and new challenges in metagenomics for biotechnology. *Biotechnol. Lett.* 2010; 32:1351–1359. [PubMed: 20495950]
3. Mardis ER. The impact of next-generation sequencing technology on genetics. *Trends Genet.* 2008a; 24:133–141. [PubMed: 18262675]
4. Pettersson E, Lundeberg J, Ahmadian A. Generations of sequencing technologies. *Genomics.* 2009; 93:105–111. [PubMed: 18992322]
5. Schuster SC. Next-generation sequencing transforms today's biology. *Nat. Methods.* 2008; 5:16–18. [PubMed: 18165802]
6. Shendure J, Ji H. Next-generation DNA sequencing. *Nat. Biotechnol.* 2008; 26:1135–1145. [PubMed: 18846087]
7. Treffer R, Deckert V. Recent advances in single-molecule sequencing. *Curr. Opin. Biotechnol.* 2010; 21:4–11. [PubMed: 20202812]
8. Tucker T, Marra M, Friedman JM. Massively parallel sequencing: the next big thing in genetic medicine. *Am. J. Hum. Genet.* 2009; 85:142–154. [PubMed: 19679224]
9. Shendure, J. and Ji, H. (2008) Next-generation DNA sequencing. *Nat. Biotechnol.*, 26, 1135–1145.
10. Metzker, M.L. (2010) Sequencing technologies—the next generation. *Nat. Rev. Genet.*, 11, 31–46.
11. Derrington, I.M., Butler, T.Z., Collins, M.D., Manrao, E., Pavlenok, M., Niederweis, M. and Gundlach, J.H. (2010) Nanopore DNA sequencing with Msp.A. *Proc. Natl Acad. Sci. USA*, 107, 16060–16065.
12. Krivanek, O.L., Chisholm, M.F., Nicolosi, V., Pennycook, T.J., Corbin, G.J., Dellby, N., Murfitt, M.F., Own, C.S., Szilagy, Z.S., Oxley, M.P. et al. (2010) Atom-by-atom structural and chemical analysis by annular dark-field electron microscopy. *Nature*, 464, 571–574.
13. Eid, J., Fehr, A., Gray, J., Luong, K., Lyle, J., Otto, G., Peluso, P., Rank, D., Baybayan, P., Bettman, B. et al. (2009) Real-time DNA sequencing from single polymerase molecules. *Science*, 323, 133–138.

14. Luan, B., Peng, H., Polonsky, S., Rossnagel, S., Stolovitzky, G. and Martyna, G. (2010) Base-by-base ratcheting of single stranded DNA through a solid-state nanopore. *Phys. Rev. Lett.*, 104, 8103.
15. Sanger, F. and Coulson, A.R. (1975) A rapid method for determining sequences in DNA by primed synthesis with DNA polymerase. *J. Mol.Biol.*, 94, 441–448.
16. Sanger, F., Nicklen, S. and Coulson, A.R. (1977) DNA sequencing with chain-terminating inhibitors. *Proc. Natl Acad. Sci. USA*, 74, 5463–5467.
17. Maxam, A.M. and Gilbert, W. (1977) A new method for sequencing DNA. *Proc. Natl Acad. Sci. USA*, 74, 560–564.
18. Hert, D.G., Fredlake, C.P. and Barron, A.E. (2008) Advantages and limitations of next generation sequencing technologies: a comparison of electrophoresis and non-electrophoresis methods. *Electrophoresis*, 29, 4618–4626.
19. Levene, M.J., Korlach, J., Turner, S.W., Foquet, M., Craighead, H.G. and Webb, W.W.(2003) Zero-mode waveguides for single-molecule analysis at high concentrations. *Science*, 299, 682–686.

4. RHIZOSPHERE SOIL METAGENOMICS – A CASE STUDY IN BLACK PEPPER

Dr.P. Umadevi & Dr.M. Anandaraj

Introduction

Lorenz Hiltner in the year 1904, coined the term “rhizosphere” to emphasize that the plant nutrition is considerably influenced by the microbial composition of the rhizosphere. He suggested that in the rhizosphere both beneficial as well as the pathogenic microbes which are able to adjust to the specific root exudates are attracted. He relied on the hypothesis that the resistance of plants towards pathogenesis & the quality plant products are very much dependent on the composition of the rhizosphere microflora (Hartmann et al. 2008). The portion/volume of soil in which the microbe mediated processes are under the influence of plant roots is termed as the rhizosphere. The microbial composition of rhizosphere soil differs from the bulk soil due to the selective recruitment of the organism by the root exudates that are specific to the plants. This brings major impact on plant fitness and productivity. Hence, the manipulation of rhizosphere microbes is a viable opportunity in improving the plant health & to restore the ecosystem function under biotic/ abiotic stress & poor land management practices. Exploration and management of rhizosphere soil microbiomes was the difficult task before as the majority of soil microbes have not yet been isolated and molecular details underlying their functions are largely unknown. But the advent of NGS has opened the way to analyze and find out the community pattern & functions in the soil rhizosphere. The techniques such as partial community analysis and whole community metagenome analysis are the useful technologies for the study of rhizosphere microbiome. “Metagenome” is the collective genome of microorganisms present in specific environment that provides the information on the diversity, ecology & functionality of microbes.

Soil whole genome metagenomics

The soil metagenomics can be of whole genome sequencing (shotgun) and amplicon sequencing (16s-rRNA/gene targeted metagenomics). Due to the difference in the data obtained from each method, the bioinformatics tools in data analysis would differ. The

application of whole-genome shotgun (WGS) sequencing is a major development in metagenomics. Unlike the targeted 16S rRNA approach, which specifically study a single gene or individual genomes, WGS approach brings a global view of the community, levels of phylogenetic diversity and intraspecies polymorphism. WGS also has the potential to discover new genes, identify the full gene complement, metabolic pathways and to construct draft genome sequences is also possible (Jansson and Hofmockel 2018).

The high throughput sequencing methods viz., Roche 454, Illumina HiSeq, SOLid, Ion Torrent and PacBio RSII & integrated bioinformatics platforms like MEGAN, MG-RAST offers enables identification of community patterns, abundance, interaction network and functional dynamics of microbes in the soil sample and thereby provides information on community ecology. MG-RAST (Meyer et al. 2008) is one of the prominent public domain platforms which supports deposition and analysis of metagenomic datasets.

Though there are plenty of sequencing & bioinformatics platforms available, the attempt on soil whole genome metagenomics is very limited. The whole genome metagenome shotgun sequencing using Roche 454 and Illumina platforms was done from forest, desert, grass land, Arctic and mangrove sediment & 33 metagenome datasets were generated and analysed using integrated bioinformatics tools. Apart from microbial community profile the indicator metagenomic biomarkers were developed for differentiating the soil communities (Xu et al. 2014). The selective microbial enrichment for biogeochemical cycling of nutrients was demonstrated towards the bioremediation at Loktak Lake, the largest freshwater lakes of India. The fresh water rhizospheric microbial community metagenomes was found to be different from four other aquatic habitats, varying from pristine to highly polluted eutrophic habitats (Puranik et al. 2016).

In another elegant study, Ascomycota was found to be the dominant phylum (76%–85%) in fungi using MG-RAST in the rhizosphere metagenome of gray mangroves of the Red sea along with several commercially-used fungi (Simoes et al. 2015). Alzubaidy et al. 2016 showed the metabolism of aromatic compounds and potassium metabolism & pathway that utilize osmolytes from the rhizosphere microbiome metagenomics of grey mangroves in the red sea using 454 GS FLX Titanium technology coupled with MG - RAST analysis. The comparative metagenomics of the vineyards in Central Chile and the native forest soil prior to the establishment of the vineyard showed that bacteria were more abundant than fungi

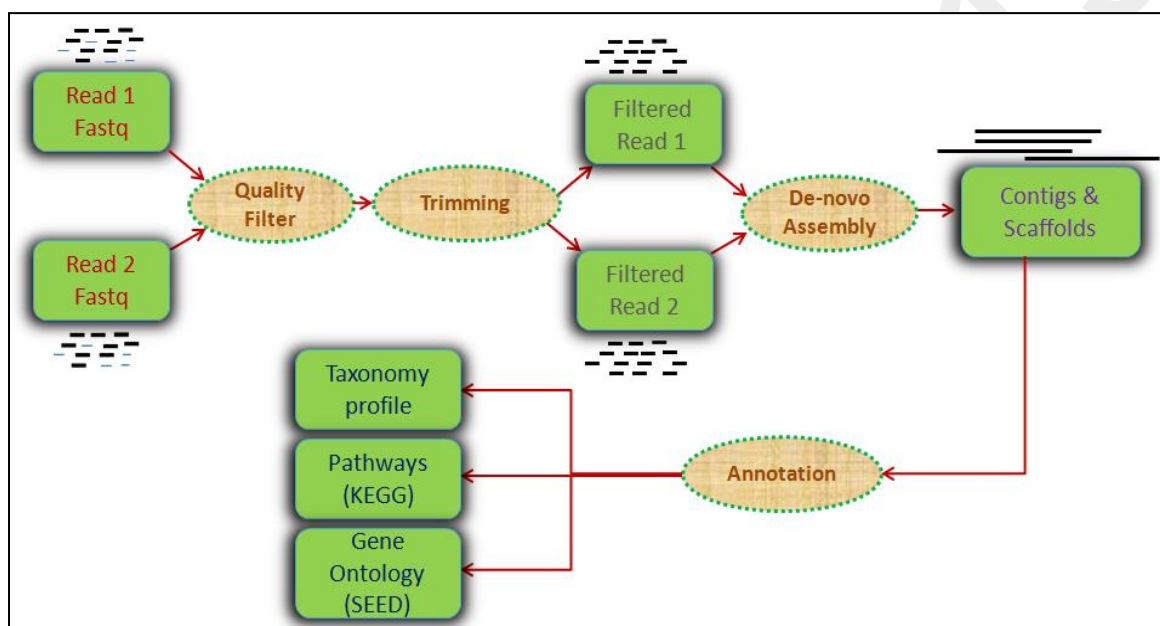
in both types of habitats. Functional analysis showed that the genes for metabolism of potassium as enriched vineyard soils & genes for metabolism of amino acids, fatty acids, nucleotides and secondary metabolism as enriched in forest soils (Castaneda and Barbosa 2016). In line of this, recently we attempted the whole genome shotgun sequencing of black pepper rhizosphere and elucidated the “Trichorhizosphere” (Umadevi et al. 2017).

Black pepper “Trichorhizosphere”:

Black pepper (*Piper nigrum* L.), the king of spices – is an export oriented important spice crop grown in tropical countries. The foot rot disease caused by *Phytophthora capsici*, an oomycete pathogen contributes to the major crop loss as it infects the vine both in the nursery and fields. The previous elegant studies on *Trichoderma harzianum* (MTCC 5179) toward its growth promotion and disease suppression activities had placed this fungus in the integrated disease management module of the cultivation of black pepper in India. Thus, we hypothesized that the probiotic application of *Trichoderma* would alter the community composition or dynamics of other soil fungi and bacteria at the rhizosphere of black pepper; and that might contribute to the plant health in a better way than the rhizosphere community without *Trichoderma*. In the light of this hypothesis, using the Illumina HiSeq whole genome metagenome sequencing approach, we studied the impact of *Trichoderma harzianum* on altering the microbial community and its functional dynamics in the rhizosphere soil of black pepper. The metagenomic datasets from the rhizosphere with (treatment) and without (control) *T. harzianum* inoculation were annotated using dual approach, i.e., stand alone and MG-RAST. The statistical analysis for relative population abundance and functional abundance was done using STAMP tool. The enriched metabolic pathways were derived from KEGG pathway analysis. The probiotic application of *T. harzianum* in the rhizosphere soil of black pepper impacted the population dynamics of rhizosphere bacteria, archae, eukaryote as reflected through the selective recruitment of bacteria and fungi in terms of abundance in population. The bacterial chemotaxis, iron metabolism with the reduction in abundance for pathogenicity islands, phages and prophages with regard to functional abundance. Interestingly, we found that the enriched functional metagenomic signatures on phytoremediation such as benzoate transport and degradation and degradation of heterocyclic aromatic compounds in the treatment influenced the rhizosphere micro ecosystem favoring growth and health of pepper plant.

The population dynamics and functional richness of rhizosphere ecosystem in black pepper influenced by the treatment

With *T. harzianum* provided the ecological importance of *T. harzianum* in the cultivation of black pepper. From the metagenomics study, we suggested that as mycorrhizosphere, another microecological niche, viz., ‘trichorhizosphere’ is also coexists in altering the community dynamics of bacteria and soil fungi; and thus, the rhizosphere microecosystem developed by *T. harzianum* might contribute a pivotal role in imparting plant health, which is



unlike the lone effect of *T. harzianum* (Umadevi et al. 2017).

Fig 1: Metagenomics data analysis workflow

Fig: 2 Metagenome identity in MG-RAST mgp14681 (Control) and mgp14668 (*Trichoderma* treated)

The figure shows two screenshots of the MG-RAST web interface. The left screenshot is for 'Sample1_normal_soil' and the right is for 'Sample14_trichoderma_treated'. Both pages display project details, a map of the location (Kochi, India), and a table of metagenomic data. The table columns include name, size, sequence count, material, sample ID, library, location, country, coordinates, type, method, and download options.

name	size	seq count	material	sample	library	location	country	coordinates	type	method	download
sample1_prime_seq_005E	1,054,544,053	4,243,868	soil	mgs71113	mgs71115	kochi, kerala	India	11.2587531, 75.78841	WGS	Illumina	download
sample14_prime_seq_005E	1,058,756,059	4,247,071	soil	mgs71116	mgs71118	kochi, kerala	India	11.2587531, 75.78841	WGS	Illumina	download

Opportunities for the future

Modifying growing conditions through cultural practices is an important crop improvement platform in agriculture, to get the desired performance/ plant fitness. The manipulation of rhizosphere microbiome known as “Rhizosphere engineering” is an important component for reshaping the soil rhizosphere microbial community. Reshaping the rhizosphere microbiome by introducing beneficial microorganisms will protect the host plant against pathogen infections. The cost effective sequencing technologies and use of integrated bioinformatics databases makes the metagenomics of non-model crop-associated rhizosphere microbiomes as a possible, widely accessible research method (Umadevi et al. 2017). The soil rhizosphere metagenomics will tell us the major microbes in the microclimate and that could be used to engineer the rhizosphere by probiotic application of major microbes as single or consortia. The understanding on the interaction of soil microbe and the root exudates will open up enormous translational research in rhizosphere engineering since the host genotype play a significant role in the formation of microbial communities, is action, interactions (Zachow et al. 2015) and it is heritable (Pfeifer et al. 2013) also. Genes mediating beneficial interactions with associated microbes can be manipulated in the elite germplasm to increase the efficiency of the rhizosphere by the gene editing techniques. Study on the quantitative trait loci (QTL) governing the responses of plant to beneficial rhizosphere microbes will help in marker assisted breeding towards developing the varieties with improved fitness against the biotic and abiotic stresses.

References:

- Umadevi P, Anandaraj M, Vivek Srivastav and Benjamin S 2017. *Trichoderma harzianum* MTCC 5179 impacts the population and functional dynamics of microbial community in the rhizosphere of black pepper (*Piper nigrum* L.). *Brazilian Journal of Microbiology*, 49 (3), 463-470. doi.org/10.1016/j.bjm.2017.05.011
- Alzubaidy, H., Essack, M., Malas, T.B., Bokhari, A, Motwalli, O., Kamanu, F.K., Jamhor, S.A., Mokhtar, N.A., Antunes, A., Simoes, M.F., Alam, I., Bougouffa, S., Lafi, F.F., Bajic, V.B., Archer, J., A., C., Rhizosphere microbiome metagenomics of gray mangroves (*Avicennia marina*) in the Red Sea. *Gene*, 576, 626–636, 2016.

- Bhattacharyya, P., Roy, K.,S., Das, M., Ray, S., Balachandar, D., Karthikeyan,S., Nayak, A.,K., Mohapatra, T., Elucidation of rice rhizosphere metagenome in relation to methane and nitrogen metabolism under elevated carbon dioxide and temperature using whole genome metagenomic approach. *Sci Total Environ*, 542, 886-898, 2016.
- Hartmann, A., Rotballer, M., Schmid, M., (2008) LoenzHiltner, a pioneer in rhizosphere microbial ecology and soil bacteriology research. *Plant Soil*, 312, 7-14, 2008.
- Jansson, J., K., Hofmockel, K., S., The soil microbiome - from metagenomics to metaphenomics. *Current Opinion in Microbiology*, 43, 162–168, 2018.
- Meyer, F., Parman, D., D'Souza, The metagenomics RAST server- a public resource for the automatic phylogenetic and functional analysis of metagenomes. *BMC Bioinformatics*, 9, 386, 2008.
- Pfeifer, J., A., Spor, A., Koren, O., Jin, Z., Tringe, S., G., Dangl, J., L., Buckler, E., S., Ley, R., E., Diversity and heritability of the maize rhizosphere microbiome under field conditions PNAS, 110, 6548-6553, 2013.
- Puranik, S., Pal, R., R., More, R., P., Purohit, H., J., Metagenomic approach to characterize soil microbial diversity of Phumdi at Loktak Lake. *Wat Sci Tech*, 74(9), 2075-2086, 2016.
- Simoes, M., F., Antunes, A., Cristiane A., Ottoni, C., A.,Amini, M., S., Alam, I., Alzubaidy, H., Mokhtar, N., John A.,C., Archer, J., A.,C., Bajic, V., B., Soil and Rhizosphere Associated Fungi in Gray Mangroves (*Avicennia marina*) from the Red Sea — A Metagenomic Approach. *Genomics Proteomics Bioinformatics*, 13, 310–320, 2015.
- Umadevi, P., Anandaraj, M., Vivek, S., Benjamin, S., *Trichoderma harzianum* MTCC 5179 impacts the population and functional dynamics in microbial community in rhizosphere of black pepper (*Piper nigrum* L.). *Braz J Microbiol*, 49(3), 463-470, 2017.
- Xu, Z., Hansen, M., A, Hansen, L., H., Jacquioid, S., Sørensen, S.J., Bioinformatic Approaches Reveal Metagenomic Characterization of Soil Microbial Community. *PLoS ONE*, 9(4), e93445, 2014.
- Umadevi, P., Anandaraj M, Vivek Srivastav and Benjamin S 2017. *Trichoderma harzianum* MTCC 5179 impacts the population and functional dynamics of microbial community in the rhizosphere of black pepper (*Piper nigrum* L.). *Brazilian Journal of Microbiology*, 49 (3), 463-470. doi.org/10.1016/j.bjm.2017.05.011

5. BASIC LINUX COMMANDS

Ms. Neethu Issac

Linux is an operating system. It is open source software whereby the users have the freedom to study, modify and redistribute it. Many distributions of Linux comes free of cost too. Users can freely download them from the respective websites.

Linux Distributions

Linux comes in different versions or flavors, called “distributions” as per the jargon. Which distribution to choose depends on the needs and expertise of the user. Some of the popular distributions include:

- Debian
- Ubuntu
- Linux Mint
- Fedora
- OpenSUSE

GUI vs CLI

A Graphical User Interface (GUI) is the most common interface of an OS. Here the user interacts with the computer through windows and dialogue boxes which contain menus, and the input is given mostly using the mouse. In contrast, a Command Line Interface (CLI) makes minimum use of the mouse and the user has to type in “commands” using the keyboard to get their work done. The use of CLI gives more power and flexibility to the user which comes at the cost of memorizing a good number of commands to work efficiently.

Many of the bioinformatics tools work in a CLI environment and a basic knowledge of the CLI behaviour and environment and some of the basic commands is indispensable for running such tools.

The Linux file system

In Linux, everything is a file. A folder is called a directory in a CLI environment. The file system of Linux is organized as a hierarchy which means a “parent directory” contains “child directories” which in turn contain their own child directories and so on. The “root”, denoted as “/” is the parent directory of all other directories/files. At any given time, we are inside a single directory which is called the “current working directory”. Every “user” of the OS has his/her own “home” folder named after the username itself.

Pathnames

A pathname of a file/folder is the route to be followed to reach that particular file or folder in the hierarchy. The directories along the path are separated by a forward slash “/” as opposed to a backward slash “\” in Windows OS. There are two types of paths.

- Absolute path – Starts at the root (/) and is the full path of the file. e.g. /home/Desktop/myfile.txt
- Relative path – Starts at the “working directory”.

The special character ‘.’ (dot) represents the current working directory (and can be omitted in some cases).

e.g. ./NGSDAT19/sequences/phyto.fasta

NGSDAT19/sequences/phyto.fasta

The special characters ‘..’ (double dots) represent the parent directory of working directory.

e.g. ../Downloads/ral.fasta

Note: The tilde symbol (~) represents the “home” folder of the current logged in user.

e.g. ~/Desktop/Bioinformatics/manual.pdf

The Linux Command Line (CLI)

In Linux, the program (software) which we use to run commands is called the “shell”. “Bash” is a version of shell used in some distributions of Linux. In a GUI environment, a program called “terminal” is used to invoke the shell or bash.

Shell prompt - appears whenever the shell is ready to accept input. It usually takes the format: your username@machinename, followed by the current working directory and a dollar sign (\$) and a blinking cursor. eg. [raj@dell-inspiron](#):~\$

Command history – is a useful feature of the shell. A user can navigate through the previously typed commands using the up and down arrow keys. Simply press the up arrow key repeatedly to see the previously entered commands.

General format of a command -

Command -options arguments

Options – modify the behaviour of a command.

- Single character options are preceded by a single hyphen (-) and multiple such options can be combined. e.g. ls -lt
- Long options consist of a word and are preceded by two dots (--). e.g. ls -lt --reverse

Arguments – items upon which a command acts.

Options and arguments are both optional, meaning that they need not be supplied always.

ls -lt --reverse

For example, in the above line, “ls” is the command and it displays the contents of the current working directory. “l” and “t” are two single character options. Option “l” causes the output to be displayed in what is called the “long format”. Option “t” causes it to be displayed sorted in the order of modification time. The long option “reverse” causes the output to be displayed in the reverse order. In the above command, there are no arguments. But in the following line, we have an argument namely “~/Downloads”. It displays the contents of the “Downloads” folder of the current logged in user. The tilde (~) denotes the home folder of the current logged in user as mentioned earlier.

ls -lt --reverse ~/Downloads

Command	Description	Usage/Example	Comments
mkdir	Create a Directory	mkdirLinux_commands	Creates directory with name Linux_commands
ls	List all the files/folder in current directory	ls	Displays all the files in current folder/Directory
cd	Change Directory	cd Linux_commands	Changes current directory location to Linux_commands
		cd ..	Returns to previous directory location
rmdir	Remove Directory	rmdir Linux_commands	Removes folder with the name Linux_commands
vi	View and edit files	vi text1	View and edit text1
rm	Remove file	rmttext1	Removes file with the name text1
cp	Copy files/folders	cp text1 ./Linux_commands /	Copy text1 to Linux_commands folder
mv	Move file or folder to a specified location	mv text1 text2	It will rename text1 to text2 i,e the name is changed.
		mv text1 ./Linux_commands	It will move the file text1 to Linux_commands folder
cat	Used to display contents of a file and concatenate	cat text1 cat text1text2>text3	Displays the content of file text1 Concatenates contents of file text1, text2to text3
pwd	Display present working directory	pwd	Displays the current path/working directory
grep	Search for a specific text in a given file	grep 'hello' text1	Searches for 'hello' in ' text1 '

		<code>grep -c 'hello' text1</code>	Displays number of times 'hello' is present in text1
wc	Prints word count of newlines, words, etc	<code>wc -l text1</code>	Prints number of lines in ' text1 '
tar	To compress and uncompress folders	<code>tar -cvf new text1 text2</code>	Compresses text1 and text2 and create new.tar file
		<code>tar -xvf new</code>	Uncompress folder new
gunzip	To uncompress zipped file	<code>gunzip filename.gz</code>	Uncompress the folder

6. NGS DATA FORMATS AND QUALITY CHECK

Mrs. Blessy M Baby

(i) Read Formats

- **FASTQ** format is a common format for short reads with quality scores. It is supported in EMBOSS 6.1.0 as a sequence format. Quality scores are also used if the format is more explicitly named in EMBOSS: fastqsanger or fastqillumina
- **SFF** Standard Flowgram Format, to hold the "trace" data for 454 reads
- **SRF** Sequence Read Format (also called Short Read Format), format specification has not been updated since February 2008. Applied Biosystems SRF Conversion Tool (solid2srf) converts SOLiD™ system reads into SRF format.
- **SCARF**, Solexa Compact ASCII Read Format. This format contains all information for one read in a single line. From left to right each line contains the read name, nucleotide sequence, quality scores for each position, and more information. Illumina's pipeline can produce SCARF files with quality scores in ASCII or numeric format.
- **SCF** first version was described in 1992, since then it has undergone several important changes such as a major reorganization of the ordering of the data items in the file and also in the way they are represented

(ii) Assembly Formats

- MAQ .map format (a compressed binary file format designed for short read alignment)
- MAF, MIRA Assembly Format
- AMOS A Modular Open-Source Assembler assembly format, used by velvet

- SAM/BAM (Sequence Alignment/Map) format is a generic format for storing large nucleotide sequence alignments

(iii) Read Formats

Sequencers use their own output formats, and even multiple different formats per technology.

(i) Output formats from Roche 454 sequencer

.sff is 454's binary (=compressed) format, including the raw data. Not the very raw images, but flow values (flowgram), base calls, base-call qualities, read trimming and quality. There are many simple tools that "unzip" .sff into some readable textual format

.fna is a FASTA (or FASTA-like) format including the base-called reads.

```
>000007_1940_1402 length=172 uaccno=E4UQSRD01E0MP4
TAACAATCGAGGCGAAGTCCCGTGAGAAGCTGTTTACTTCTCATGATCACACAGGCGCTG
GCTCCTCAGGCAAACAGGTACGTCTACGATAGGTTCCATGAAAAGTCCAAGTTTGGCCGA
GCTCTGGCTCCTTTTGACGCACAGTGGAACTTCCTTGTTACGGAAATTGCA
```

Read with .fna format.

.qual includes qualities corresponding to bases in .fna.

```
>000007_1940_1402 length=172 uaccno=E4UQSRD01E0MP4
28 35 28 27 34 27 26 25 25 28 31 24 26 27 32 25 27 27 32 28 6 28 27 27 27 27 33 26
27 26 27 27 34 30 10 27 25 34 27 28 22 28 27 26 26 27 27 26 27 25 22 23 28 27 18 20
23 27 27 29 21 25 25 34 26 27 24 25 32 24 22 33 28 7 25 20 30 22 28 27 24 25 28 28
28 27 28 26 27 25 23 33 25 35 28 34 27 27 25 28 38 34 21 8 25 27 34 27 31 23 22 36
32 17 29 21 32 24 24 27 28 19 27 28 26 34 28 23 25 35 28 38 34 21 8 26 26 27 25 27
21 28 28 27 27 34 27 34 27 25 30 21 34 26 33 25 26 35 28 20 28 25 34 27 37 33 15 33
25 23 28 25
```

Read with .qual format.

(ii) Output formats from Illumina: Illumina is over-creative with their own formats.

.seq.txt: base-called sequences

```
File: s_1_0001_seq.txt
1 1 137 689 AACATAATGTGTTCACTGAGAACACATTGCACTCAA
1 1 87 649 TATTGCAACTTGTTTAATTTTTCATGCCATTATCA
1 1 121 642 TACATGATTTGCATTTGGTAAATAGCTACTTTTTAT
1 1 6 591 C...T.....T.....
```

Read with .seq.txt format.

.prb.txt: qualities for each nucleotide/each cycle (is some kind of semi-raw data?)

40	-40	-40	-40	40	-40	-40	-40	-40	40	-40	-40	40	-40	-40	-40
-40	-40	-40	40	40	-40	-40	-40	40	-40	-40	-40	-40	-40	-40	40
-40	-40	40	-40	-40	-40	-40	40	-40	-40	40	-40	-40	-40	-40	40
-40	-40	-40	40	-40	40	-40	-40	40	-40	-40	-40	-40	40	-40	-40
-40	-40	-40	40	-40	-40	40	-40	40	-40	-40	-40	-40	-40	40	-40
40	-40	-40	-40	40	-40	-40	-40	-40	40	-40	-40	40	-40	-40	-40
-40	40	-40	-40	40	-40	-40	-40	-40	-40	-40	40	-40	-40	-40	40
-40	-40	40	-40	-40	40	-40	-40	40	-40	-40	-40	-40	40	-40	-40
-40	-40	-40	40	-40	40	-40	-40	40	-40	-40	-40	37	-37	-40	-40

Read with .prb.txt format.

Illumina FASTQ is not real FASTQ, but looks similarly (to cause confusion). The quality score is an own Illumina score, computed by subtracting 64 from the ASCII code of the character. For example the 'h' in the example is $\text{ord}('h') - 64 = 104 - 64 = 40$.

```
@ILMN-GA001 3 208HWAAXX 1 1 110 812
ATACAAGCCAAGTATAAGTTCGTATGCCGTCTT
+ILMN-GA001 3 208HWAAXX 1 1 110 812
hhhYhh}NYhhhhhhYIhhaZT[hYHNSPKXR
@ILMN-GA001 3 208HWAAXX 1 1 111 879
GGAGGCTGGAGTTGGGGACGTATCGGCATAG
+ILMN-GA001 3 208HWAAXX 1 1 111 879
hSWhrNJ\hFhLdhVohAIB@NFKD@PAB?N?
```

Read with .fastq format.

The Illumina score can be transformed to Phred score. Given a character ch, the following expression should give the Phred quality q (please double-check :) $q = 10 * \log(1 + 10^{(\text{ord}(ch) - 64) / 10.0}) / \log(10)$

Qseq: Not looking like FASTQ, but the scores are proper Phred scores. Their encoding is however still ASCII - 64.

HWUSI-EAS521	2	1	26	0	76	0	1	.GGCAGCGGGCAGGGCAGCCAATGCGTGTGGGGGGGGGGCTCGCAGTGGGGGGGGAACGGCGAGTGGGGGGG	0
HWUSI-EAS521	2	1	26	0	562	0	1	.GGGAAATAGCTTTCACGCCTTATGATAATTTCAATAAAATCATAGCGCCAATGGGGAGCAAACCTACATACC	0
HWUSI-EAS521	2	1	26	0	1113	0	1	.ATCTTTAACAGACCAAGACTGGGCCCAAGCCTCCAGACTGTAACTGCTTCAAAGAGGCTTAGGCAGGCAGG	0
HWUSI-EAS521	2	1	26	0	244	0	1	.GGCTGGGTATGAGTCAGGGGGCTCCAGAGAGACAGAACAGTCCGACATCGACAGATAGATCGGGGGGAGTTAT	0

Read with .Qseq format.

Illumina “**single line format**” and **SCARF**: base-called sequences and quality scores in a bit handier format.

```
>1-1-137-689 AACATAATGTTCACTGAGAACACATTGCACTCAA U0
>1-1-87-649 TATTGCAACTGTTTAAATTTTTCATGCCATTATCA U1
>1-1-121-642 TACATGATTGCACTTGGTAAATAGCTACTTTTAT U0
```

Read with single line format

```
HWI-EAS102_3 : 6 : 1 : 897 : 791 : AATGTCAATCTGAGTT ... TTT : 40 40 40 40 40 ..
HWI-EAS102_3 : 6 : 1 : 930 : 291 : AATGTACTTTTTCTAA ... CTA : 40 29 14 17 16 ..
HWI-EAS102_3 : 6 : 1 : 944 : 665 : AATCGATCCCCTTCCC ... TTC : 40 34 33 40 40 ..
```

Read with SCARF format**(iii) Output formats from SOLiD**

SOLiD outputs **CSFASTA**. It’s a semi-raw format similar to FASTQ. Read starts with the first base, but colour calls follow.

```
@ERR000451.1 VAB_S0103_20080915_542_14_17_70_F3
T33023230203102103223330020300233001
+
T%245719<.6353&:%0#%&%2 (--27*%&%,
```

Read with CsFASTA format**Assembly Format Sample**

Read aligners output a number of more or less similar own formats. However many of them output data in the common SAM/BAM format,

For the typically large data, BAM is currently the most recommended and most “standard” format.

Metadata Storage

Nucleotide sequence databases (GenBank, EMBL, and DDBJ, gathered under the INSDC consortium) store a number of types of metadata about the genomic sequences.

However, additional metadata is crucial to enable efficient future research using the submitted genome & metagenome-sequencing data. GSC consortium has therefore defined the MIGS and MIMS standards for additional metadata that should be submitted to a public repository.

The minimum information about a genome sequence (MIGS) specification .Extended to the Minimum Information about a Metagenomic Sequence/Sample: MIMS. MIENS (environmental sequencing) follow. GCDML is an XML format for MIGS/MIMS/MIENS metadata, (Using XML Schema and SAWSDL annotations.)

It's however crucial for each data set to decide what data should be deleted, and after what time! When and whether to delete the raw data, flows, colours, when and whether the base-called reads & qualities, alignments. The data is often enormously huge and it will cost you and/or the national/global institutes a lot of money and energy to store and back them up. Hence we use SRA format (XML), SRF format.

NCBI SRA submission

SRA submission using the interactive submission interface

Researchers often need to submit their original sequence data to NCBI SRA (short read archive) for review. Sequence data to be submitted have to be accompanied by metadata which hold information of the study, experimental design, samples, run parameters, etc. In general there are two ways to submit metadata to the archive:

- ⌘ Understand SRA terminology: While you are waiting for your account, go through SRA starter quick guide: <http://www.ncbi.nlm.nih.gov/books/NBK47529/> basically, in each submission, you can include one or multiple studies. You will be asked for an abstract describing each study. Associated with each study are one

or multiple samples. A sample describes the nature and origin of the material which has been sequenced. You can have multiple experiments associated with one particular sample within a study. The experiment holds details on how the sequencing library was prepared. For each experiment, you can have multiple sequencing runs (NCBI suggest to include only one lane in each "run").

- ⌘ Use the interactive submission interface on their web site:

http://trace.ncbi.nlm.nih.gov/Traces/sra_sub/sub.cgi?&m=submissions&s=default

t. The interactive submission interface allows tracking of your submissions and informs you about errors in the submission process.

- ⌘ Generate a set of XML files which hold all relevant data and submit them using ftp or ascp (something similar to and faster than ftp). In any cases the actual sequencing data are submitted via ftp or ascp.

If you have just a few lanes of sequencing data we suggest using the interactive tool for generation of the metadata and submission of your files. If you have large data sets we provide help with the generation of XML files and the transfer via ascp.

Quality check

Next generation sequencing (NGS) technologies provide a high-throughput means to generate large amount of sequence data. However, quality check (QC) of sequence data generated from these technologies is extremely important for meaningful downstream analysis. Further, highly efficient and fast processing tools are required to handle the large volume of datasets. Most sequencers will generate a QC report as part of their analysis pipeline, but this is usually only focused on identifying problems which were generated by the sequencer itself. QC aims to get a QC report which can spot problems which originate either in the sequencer or in the starting library material.

After obtaining reads (Fast Q, Fast A) from sequencer machine, first step is to check for QC data

- ⌘ Next generation high throughput sequencers generate huge amount of sequencing data (100Mb to few GB's).

- ☞ Quality check and primary analysis of raw sequence data is vital prior to the in-depth analysis.

FastQC

FastQC aims to provide a QC report which can spot problems which originate either in the sequencer or in the starting library material. FastQC can be run in one of two modes. It can either run as a standalone interactive application for the immediate analysis of small numbers of FastQ files, or it can be run in a non-interactive mode where it would be suitable for integrating into a larger analysis pipeline for the systematic processing of large numbers of files.

Steps:

1. Basic analysis

(i) Opening a Sequence file

To open one or more Sequence files interactively simply run the program and select File > Open. You can then select the files you want to analyze.

Newly opened files will immediately appear in the set of tabs at the top of the screen. Because of the size of these files it can take a couple of minutes to open them. FastQC operates a queuing system where only one file is opened at a time, and new files will wait until existing files have been processed.

FastQC supports files in the following formats

- ☞ FastQ (all quality encoding variants)
- ☞ CasavaFastQ files*
- ☞ ColorspaceFastQ
- ☞ GZip compressed FastQ
- ☞ SAM
- ☞ BAM
- ☞ SAM/BAM Mapped only (normally used for color space data)

* Casavafastq format is the same as regular fastq except that the data is usually split across multiple files for a single sample. The files contain poor quality sequences which have been flagged to be removed. In Casava mode the program will exclude these flagged sequences from the report.

By default FastQC will try to guess the file format from the name of the input file. Anything ending in .sam or .bam will be opened as a SAM/BAM file (using all sequences, mapped and unmapped), and everything else will be treated as FastQ format.

(ii) Evaluating Results

The analysis in FastQC is performed by a series of analysis modules.

It is important to stress that although the analysis results appear to give a pass/fail result, these evaluations must be taken in the context of what you expect from your library. A 'normal' sample as far as FastQC is concerned is random and diverse.

(iii) Saving a Report

To create a report simply select File > Save Report from the main menu. By default a report will be created using the name of the fastq file with _fastqc.zip appended to the end. The report will be created for whichever file tab was active when the menu option was selected. The report file which is created is actually a zip file.

Analysis modules

(i) Basic Statistics

The Basic Statistics module generates some simple composition statistics for the file analyzed.

- ⊗ Filename: The original filename of the file which was analysed
- ⊗ File type: Says whether the file appeared to contain actual base calls or color space data which had to be converted to base calls
- ⊗ Encoding: Says which ASCII encoding of quality values was found in this file.

- ⊘ Total Sequences: A count of the total number of sequences processed. There are two values reported, actual and estimated.
- ⊘ Filtered Sequences: If running in Casava mode sequences flagged to be filtered will be removed from all analyses. The number of such sequences removed will be reported here. The total sequences count above will not include these filtered sequences and will be the number of sequences actually used for the rest
- ⊘ Sequence Length: Provides the length of the shortest and longest sequence in the set. If all sequences are the same length only one value is reported. %GC: The overall %GC of all bases in all sequence

(ii) Per Base Sequence Quality

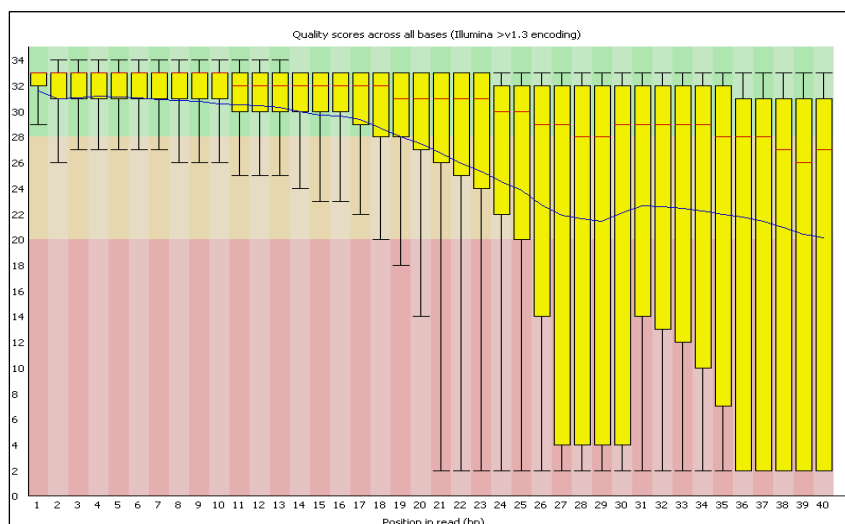


Figure 1: This view shows an overview of the range of quality values across all bases at each position in the FastQ file.

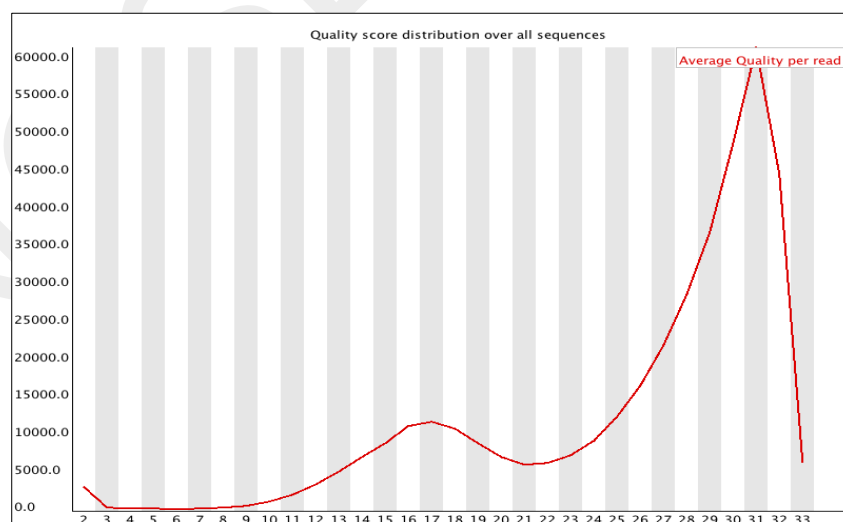
For each position a Box Whisker type plot is drawn. The elements of the plot are as follows:

- ⊘ The central red line is the median value
- ⊘ The yellow box represents the inter-quartile range (25-75%)
- ⊘ The upper and lower whiskers represent the 10% and 90% points
- ⊘ The blue line represents the mean quality

The y-axis on the graph shows the quality scores. The higher the score the better the base call. The background of the graph divides the y axis into very good quality calls (green) calls of reasonable quality (orange), and calls of poor quality (red). The quality of calls on most platforms will degrade as the run progresses, so it is common to see base calls falling into the orange area towards the end of a read. It should be mentioned that there are number of different ways to encode a quality score in a Fast Q file. FastQC attempts to automatically determine which encoding method was used, but in some very limited datasets it is possible that it will guess this incorrectly (ironically only when your data is universally very good!). The title of the graph will describe the encoding FastQC thinks your file used. A warning will be issued if the lower quartile for any base is less than 10, or if the median for any base is less than 25. This module will raise a failure if the lower quartile for any base is less than 5 or if the median for any base is less than 20.

(iii) Per Sequence Quality Scores

The per sequence quality score report allows you to see if a subset of your sequences have universally low quality values. It is often the case that a subset of sequences will have universally poor quality, often because they are poorly imaged (on the edge of the field of view etc), however these should represent only a small percentage of the total sequences.

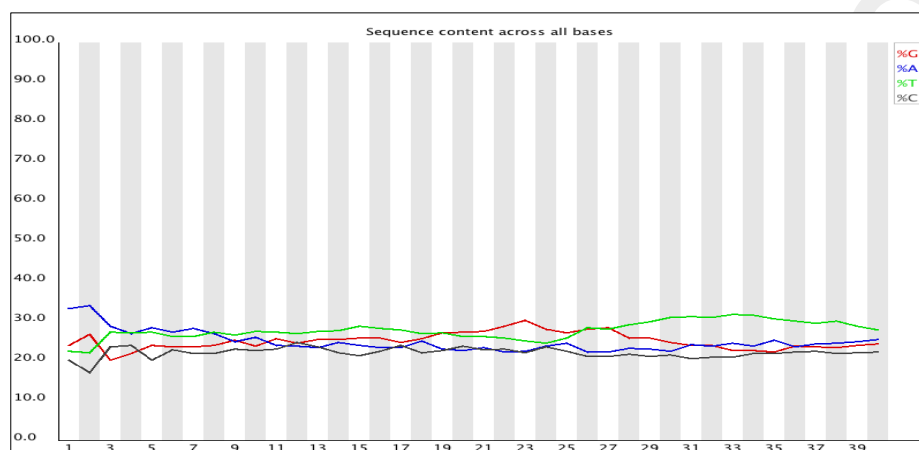


If a significant proportion of the sequences in a run have overall low quality then this could indicate some kind of systematic problem - possibly with just part of the run (for example one end of a flowcell). A warning is raised if the most frequently observed mean quality is

below 27 - this equates to a 0.2% error rate. An error is raised if the most frequently observed mean quality is below 20 - this equates to a 1% error rate.

(iv) Per Base Sequence Content

Per Base Sequence Content plots out the proportion of each base position in a file for which each of the four normal DNA bases has been called.

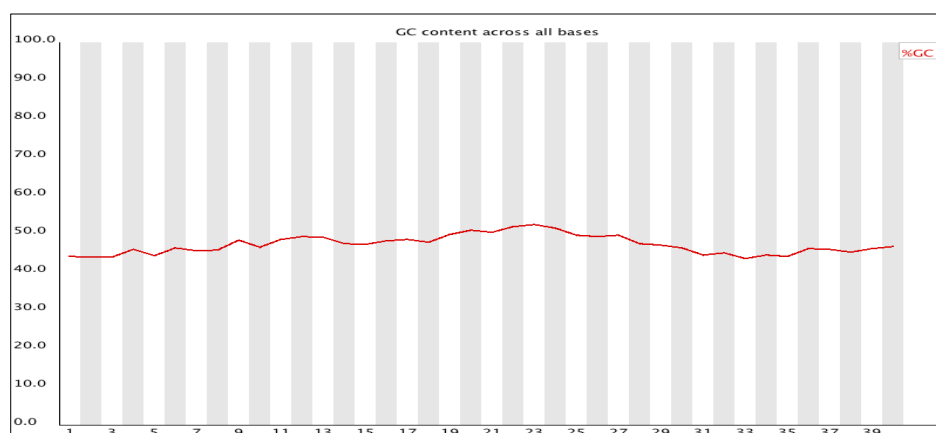


In a random library you would expect that there would be little to no difference between the different bases of a sequence run, so the lines in this plot should run parallel with each other. The relative amount of each base should reflect the overall amount of these bases in your genome, but in any case they should not be hugely imbalanced from each other. If you see strong biases which change in different bases then this usually indicates an overrepresented sequence which is contaminating your library. A bias which is consistent across all bases either indicates that the original library was sequence biased, or that there was a systematic problem during the sequencing of the library. This module issues a warning if the difference between A and T, or G and C is greater than 10% in any position. This module will fail if the difference between A and T, or G and C is greater than 20% in any position.

(V) Per Base GC Content

Per Base GC Content plots out the GC content of each base position in a file. In a random library you would expect that there would be little to no difference between the different

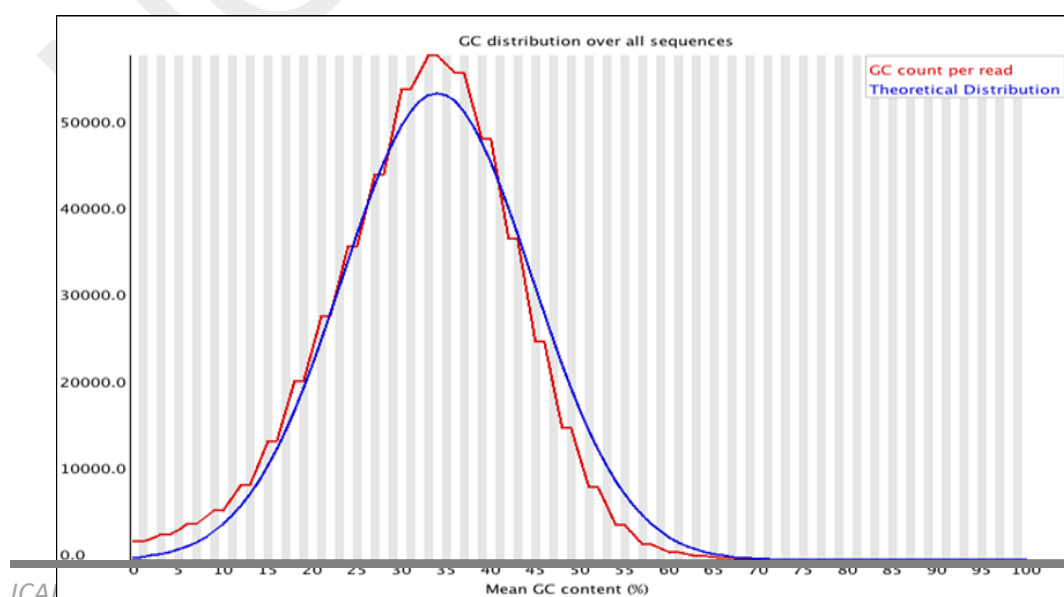
bases of a sequence run, so the line in this plot should run horizontally across the graph. The overall GC content should reflect the GC content of the underlying genome.



If you see a GC bias which changes in different bases then this could indicate an over presented sequence which is contaminating your library. A bias which is consistent across all bases either indicates that the original library was sequence biased, or that there was a systematic problem during the sequencing of the library. This module issues a warning if the GC content of any base strays more than 5% from the mean GC content. This module will fail if the GC content of any base strays more than 10% from the mean GC content.

(VI) Per Sequence GC Content

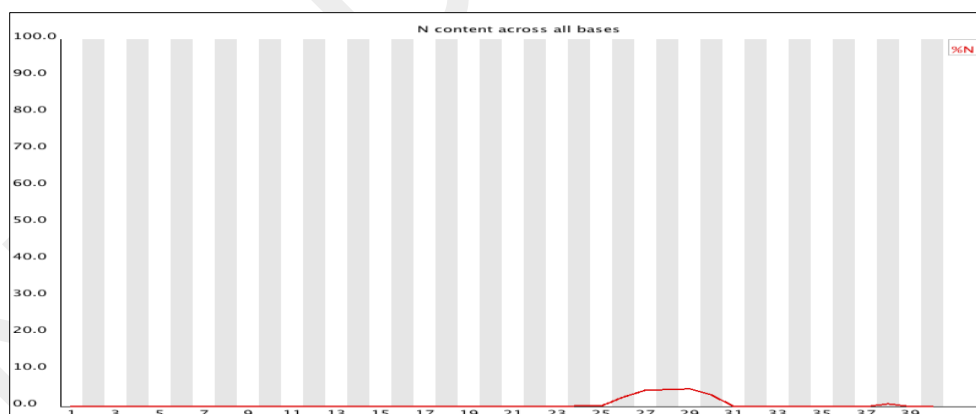
This module measures the GC content across the whole length of each sequence in a file and compares it to a modelled normal distribution of GC content.



In a normal random library you would expect to see a roughly normal distribution of GC content where the central peak corresponds to the overall GC content of the underlying genome. Since we don't know the GC content of the genome the modal GC content is calculated from the observed data and used to build a reference distribution. An unusually shaped distribution could indicate a contaminated library or some other kinds of biased subset. A normal distribution which is shifted indicates some systematic bias which is independent of base position. If there is a systematic bias which creates a shifted normal distribution then this won't be flagged as an error by the module since it doesn't know what your genome's GC content should be. A warning is raised if the sum of the deviations from the normal distribution represents more than 15% of the reads. This module will indicate a failure if the sum of the deviations from the normal distribution represents more than 30% of the reads.

(vii) Per Base N Content

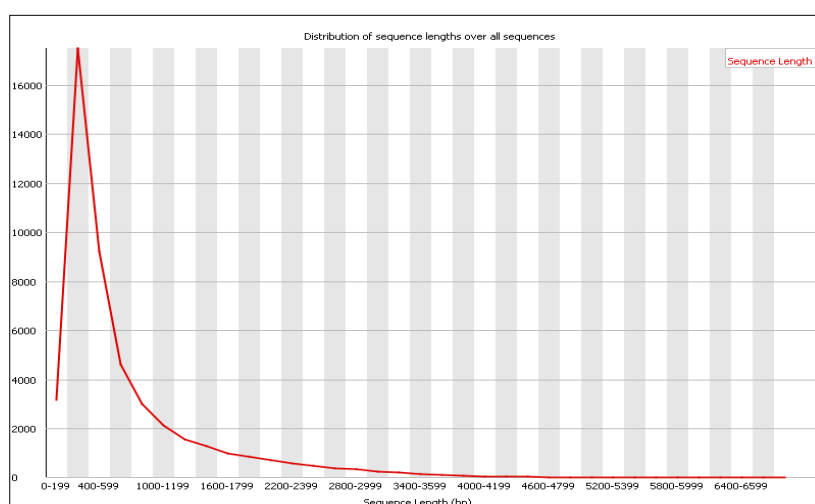
If a sequencer is unable to make a base call with sufficient confidence then it will normally substitute an N rather than a conventional base call. This module plots out the percentage of base calls at each position for which an N was called.



It's not unusual to see a very low proportion of Ns appearing in a sequence, especially nearer the end of a sequence. However, if this proportion rises above a few percent it suggests that the analysis pipeline was unable to interpret the data well enough to make valid base calls. This module raises a warning if any position shows an N content of >5%. This module will raise an error if any position shows an N content of >20%.

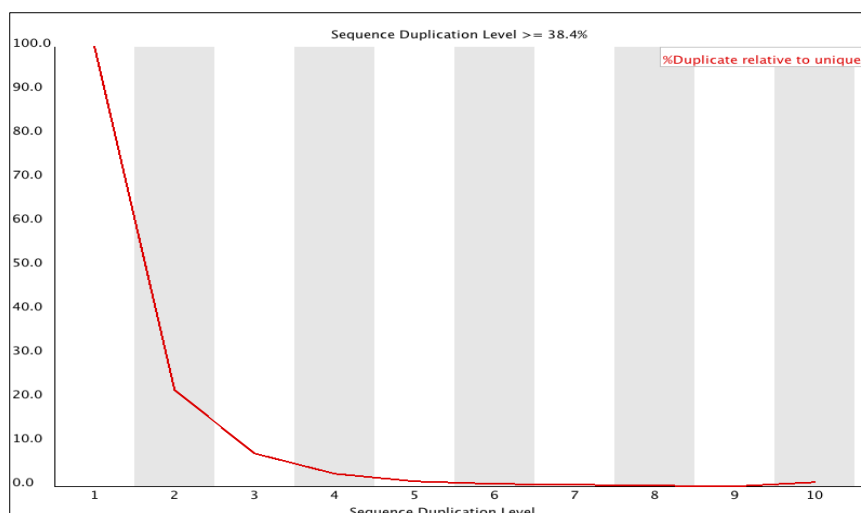
(viii) Sequence Length Distribution

Some high throughput sequencers generate sequence fragments of uniform length, but others can contain reads of wildly varying lengths. Even within uniform length libraries some pipelines will trim sequences to remove poor quality base calls from the end. This module generates a graph showing the distribution of fragment sizes in the file which was analyzed. In many cases this will produce a simple graph showing a peak only at one size, but for variable length FastQ files this will show the relative amounts of each different size of sequence fragment. This module will raise a warning if all sequences are not the same length. This module will raise an error if any of the sequences have zero length.



(ix) Duplicate Sequences

In a diverse library most sequences will occur only once in the final set. A low level of duplication may indicate a very high level of coverage of the target sequence, but a high level of duplication is more likely to indicate some kind of enrichment bias (eg PCR over amplification). This module counts the degree of duplication for every sequence in the set and creates a plot showing the relative number of sequences with different degrees of duplication.




Each sequence is tracked to the end of the file to give a representative count of the overall duplication level. To cut down on the amount of information in the final plot any sequences with more than 10 duplicates are placed into the 10 duplicates category. If you see a big rise in this final category then it means you have a large number of sequences with very high levels of duplication. Because the duplication detection requires an exact sequence match over the whole length of the sequence any reads over 75bp in length are truncated to 50bp for the purposes of this analysis. Even so, longer reads are more likely to contain sequencing errors which will artificially increase the observed diversity and will tend to under represent highly duplicated sequences. This module will issue a warning if non-unique sequences make up more than 20% of the total. This module will issue an error if non-unique sequences make up more than 50% of the total.

(x) Overrepresented Sequences

A normal high-throughput library will contain a diverse set of sequences, with no individual sequence making up a tiny fraction of the whole. Finding that a single sequence is very overrepresented in the set either means that it is highly biologically significant, or indicates that the library is contaminated, or not as diverse as you expected. This module lists all of the sequence which makes up more than 0.1% of the total. To conserve memory only sequences which appear in the first 200,000 sequences are tracked to the end of the file. It is therefore possible that a sequence which is overrepresented but doesn't appear at the start of the file for some reason could be missed by this module. For each overrepresented

sequence the program will look for matches in a database of common contaminants and will report the best hit it finds. Hits must be at least 20bp in length and have no more than 1 mismatch. Finding a hit doesn't necessarily mean that this is the source of the contamination, but may point you in the right direction. Because the duplication detection requires an exact sequence match over the whole length of the sequence any reads over 75bp in length are truncated to 50bp for the purposes of this analysis. Even so, longer reads are more likely to contain sequencing errors which will artificially increase the observed diversity and will tend to under represent highly duplicated sequences. This module will issue a warning if any sequence is found to represent more than 0.1% of the total. This module will issue an error if any sequence is found to represent more than 1% of the total.

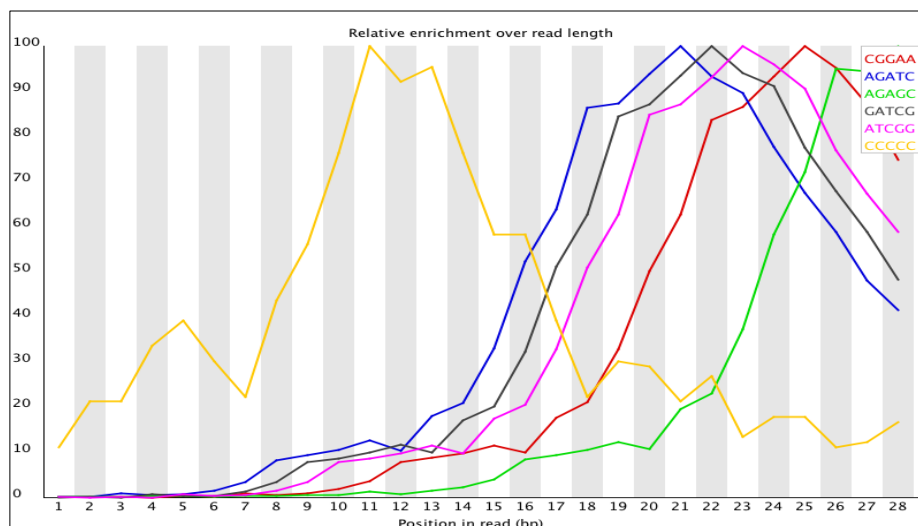
 **Overrepresented sequences**

Sequence	Count	Percentage	Possible Source
AGAGTTTTATCGCTCCATGACGCAGAAAGTTAACACTTTC	2065	0.5224039181558763	No Hit
GATTGGCGTATCCAACCTGCAGAGTTTTATCGCTTCCATG	2047	0.5178502762542754	No Hit
ATTGGCGTATCCAACCTGCAGAGTTTTATCGCTTCCATGA	2014	0.5095019327680071	No Hit
CGATAAAATGATTGGCGTATCCAACCTGCAGAGTTTTAT	1913	0.4839509420979134	No Hit
GTATCCAACCTGCAGAGTTTTATCGCTCCATGACGCAGA	1879	0.47534961850600066	No Hit
AAAAATGATTGGCGTATCCAACCTGCAGAGTTTTATCGCT	1846	0.4670012750197325	No Hit
TGATTGGCGTATCCAACCTGCAGAGTTTTATCGCTTCCAT	1841	0.46573637449150995	No Hit
AACCTGCAGAGTTTTATCGCTTCCATGACGCAGAAAGTTAA	1836	0.46447147396328753	No Hit

(xi) Over represented Kmers

The analysis of overrepresented sequences will spot an increase in any exactly duplicated sequences, but there is a different subset of problems where it will not work. If you have very long sequences with poor sequence quality then random sequencing errors will dramatically reduce the counts for exactly duplicated sequences. If you have a partial sequence which is appearing at a variety of places within your sequence then this won't be seen either by the per base content plot or the duplicate sequence analysis. This module counts the enrichment of every 5-mer within the sequence library. It calculates an expected level at which this k-mer should have been seen based on the base content of the library as a whole and then uses the actual count to calculate an observed/expected ratio for that k-mer. In addition to reporting a list of hits it will draw a graph for the top 6 hits to show the

pattern of enrichment of that Kmer across the length of your reads. This will show if you have a general enrichment, or if there is a pattern of bias at different points over your read length.



Any k-mer showing more than a 3 fold overall enrichment or a 5 fold enrichment at any given base position will be reported by this module. To allow this module to run in a reasonable time only 20% of the whole library is analyzed and the results are extrapolated to the rest of the library. This module will issue a warning if any k-mer is enriched more than 3 fold overall, or more than 5 fold at any individual position. This module will issue an error if any k-mer is enriched more than 10 fold at any individual base position.

Practical:

Quality control checks on raw sequence data using FASTQC

Quality check of sequences using FastQC, helps checking quality parameters for data generated using NGS platform

Command line:

`$FastQC_DIR/fastqc *.fq` (The wildcard * is used for executing fastqc for all fastq files present in current folder)

Using Interface:

\$FastQC_DIR/fastqc ---> Open ---> Browse your fastq file ---->Fastqc process ---> Analyse result

Adapter trimming and quality filtration of sequences

Trimming of adapter and low quality sequences using Trimmomatic

```
$java -jar /share/apps/Trimmomatic-0.36/trimmomatic-0.36.jar PE -phred33
Sp.ds.left.fqSp.ds.right.fqSp.ds.left.HQ.fqSp.ds.left.unpair.fqSp.ds.right.HQ.fqSp.ds.right.unpair.fq ILLUMINACLIP:adapter.fasta:2:40:15 LEADING:20 TRAILING:20
SLIDINGWINDOW:20:20 MINLEN:40
```

```
TrimmomaticPE: Started with arguments:
-phred33 Sp.ds.left.fq Sp.ds.right.fq Sp.ds.left.HQ.fq Sp.ds.left.unpair.fq Sp.ds.right.HQ.fq Sp.ds.right.unpair.fq ILLUMINACLIP:adapter.fasta:2:40:15 LEADING:20 TRAILING:20 SLIDINGWINDOW:20:20 MINLEN:40
Using Long Clipping Sequence: 'CAAGCAGAAGACGGCATAACGAGAT'
Using Long Clipping Sequence: 'AATGATACGGCGACCACCGAGATCTACACTCTTTCCCTACACGACGCTCTTCCGATCT'
Using Long Clipping Sequence: 'ATCTCGTATGCCGTCTTCTGCTTG'
Using Long Clipping Sequence: 'ACACTCTTTCCCTACACGACGCTCTTCCGATCT'
Using Long Clipping Sequence: 'AGATCGGAAGAGCGGTTCAGCAGGAATGCCGAG'
Using Long Clipping Sequence: 'GATCGGAAGAGCACACGTCTGAACTCCAGTCAC'
ILLUMINACLIP: Using 0 prefix pairs, 6 forward/reverse sequences, 0 forward only sequences, 0 reverse only sequences
Input Read Pairs: 101577 Both Surviving: 93692 (92.24%) Forward Only Surviving: 1245 (1.23%) Reverse Only Surviving: 4697 (4.62%) Dropped: 1943 (1.91%)
TrimmomaticPE: Completed successfully
```

BBMap:

BBMap is a short read aligner, as well as various other bioinformatic tools. It is written in pure Java, can run on any platform, and has no dependencies other than Java being installed (compiled for Java 6 and higher). All tools are efficient and multithreaded. Some of the important tools related to quality trimming and filtering:

BBMap: Short read aligner for DNA and RNA-seq data. Capable of handling arbitrarily large genomes with millions of scaffolds. Handles Illumina, PacBio, 454, and other reads; very high sensitivity and tolerant of errors and numerous large indels. Very fast.

BBNorm: Kmer-based error-correction and normalization tool.

Dedupe: Simplifies assemblies by removing duplicate or contained subsequences that share a target percent identity.

Reformat: Reformats reads between fasta/fastq/scarf/fasta+qual/sam, interleaved/paired, and ASCII-33/64, at over 500 MB/s.

BBDuk

“Duk” stands for Decontamination Using Kmers. BBDuk was developed to combine most common data-quality-related trimming, filtering, and masking operations into a single high-performance tool.

It is capable of quality-trimming and filtering, adapter-trimming, contaminant-filtering via kmer matching, sequence masking, GC-filtering, length filtering, entropy-filtering, format conversion, histogram generation, subsampling, quality-score recalibration, kmer cardinality estimation, and various other operations in a single pass.

Specifically, any combination of operations is possible in a single pass, with the exception of kmer-based operations (kmer trimming, kmer masking, or kmer filtering); at most 1 kmer-based operation can be done in a single pass. BBDuk2 allows multiple kmer-based operations in a single pass, and is otherwise equivalent to BBDuk.

Features

- ⌘ Pure Java, runs on any platform; already compiled, just unzip and run.
- ⌘ Fast, efficient, and multithreaded.
- ⌘ Usage information displayed when running a shell script with no parameters.
- ⌘ Highest sensitivity of any short-read aligner.
- ⌘ Easy to install - just unzip/untar.
- ⌘ Easy to use. Example: `bbmap.sh ref=ecoli.fa in=reads.fq out=mapped.sam`
- ⌘ Handles all common formats: fasta, fastq, sam, scarf, fasta+qual, ASCII-33, ASCII-64, gzip.
- ⌘ Used by the Joint Genome Institute.
- ⌘ BB stands for Bestus Bioinformatics.

REFERENCE links:

FASTQC:<https://www.bioinformatics.babraham.ac.uk/projects/fastqc/>

TRIMMOMATIC:<http://www.usadellab.org/cms/?page=trimmomatic/>

TrimmomaticManual_V0.32.pdf

BBDUK: <https://jgi.doe.gov/data-and-tools/bbtools/bb-tools-user-guide/bbdduk-guide/>

7. GENERAL WORKFLOW FOR METAGENOME DATA ANALYSIS

Dr. Muhammed Manzoor A.P.

Metagenome refers the collective microbial genomes and genes in an environment or sample. Metagenomics is a new technique used to study the communities of microorganisms based on their DNA from any environment. Microbiome refers the collection of microbial genomes at a given site. The microbiome analysis involves determining the composition and function of a community of microorganisms in a particular location. The advances in next generation sequencing have made it possible to study bacterial communities as a whole without limiting to small subsections. Some of the well studied microbial niches are the ones that are found on/inside the human body, and this is of great interest in relation to health and disease.

Microbiome projects worldwide have been launched with the goal of understanding the roles of these bacterial associations and their impacts on human health. The beneficial function of microbiota has been well established; it influences the immune system, the metabolism and aids in keeping the general homeostasis of the body. Most metagenomic studies concerning the human microbiota have used targeted methods focusing on the 16S marker gene. Rapidly developing sequencing methods and analytical techniques are enhancing the ability to understand the human microbiome. The microbiome and the human body maintain a balance in the diversity of the microbes within according to the lifestyle and constant supply of nutrition is provided through human diet. In turn, the microbes produce metabolites that are essential for the human body.

On the bioinformatics front, there have been tremendous efforts to develop systems to process and analyze microbiome data, including QIIME and mothur, primarily for 16S rRNA data, while metagenomic data analysis is still often pieced together using various individual bioinformatic methods.

Study design

An obvious stage in establishing a new project involving # microbiome analysis is study design. Some of the parameters might be fixed, based on the hypothesis being tested, such as changes in the microbiome (longitudinal analysis) or whether microbiome differences

correlate with the biological phenotypes (cross-sectional or cohort analysis). Another feature of the study design is whether to perform marker gene analysis or metagenomic sequencing. Marker gene analysis differs from metagenomic shotgun sequencing as it is based on targeting an amplicon of only one gene instead of attempting to sequence all or most genes in a sample. The marker gene most commonly used in prokaryotes is the ubiquitous 16S ribosomal RNA (rRNA) gene, which consists of nine variable regions, each flanked by highly conserved DNA that provides ideal primer sites for amplification. For fungi and other single-cell eukaryotes, the internal transcribed spacer (ITS) region and 18S ribosomal RNA gene are the preferred marker genes.

Sample collection and extraction of nucleic acids

Studying the microbiome can involve sampling from a wide array of sites and so the sampling plan is ultimately dependent on the hypothesis in question. The DNA and/or RNA extracted from collected samples must be of sufficient quantity, of high quality, and must contain a faithful representation of the microbial community present in the sample. Nucleic acid extraction from most human body sites including the gut is a complex procedure owing to the occasionally high proportion of (unwanted) host DNA or RNA, along with other substances including food and cellular metabolites. Unfortunately, no gold-standard method suitable for all sample and cell types exists. DNA yield and apparent the microbiome composition when different extraction kits were compared. The 16S rRNA gene is composed of regions of conserved sequence (near-identical across most bacteria) and of variable sequence or regions, which are phylogenetically distinct for a particular genus and species. Sequencing amplicon pools derived from one or more variable regions reveals 'who is there' in terms of relative abundances of bacterial taxa, along with comparisons of alpha diversity (within-sample diversity; one value per sample) and beta diversity (between-sample diversity; pairwise values for all sample combinations). The Illumina Nextera chemistry and the 16S rRNA V3–V4 region primers are very widely used, with 2×250 paired-end reads.

Sequencing and Bioinformatic analysis

The microbiome data analysis is usually tailored to the question and data type and there are many methods and protocols, even for analysing the same type of data. A large repertoire

of free and open-source software is available for the various analytical steps, from quality-filtering of the raw data to the final stages of visualizing results. The value of a study is ultimately circumscribed by the quality of the data to be analysed so it is essential to quality check and assess all sequencing reads before proceeding with the downstream analysis and interpretation, as low-quality reads will inhibit the assembly of contigs and reduce annotation efficiency. A key starting point is the assessment of the data quality. Software tools such as FastQC can be applied to assess the overall quality of the sequencing runs, while Trimmomatic and tools in the FASTX-Toolkit can be used to filter and trim sequencing reads based on PHRED quality scores, by setting desired lengths or thresholds.

Bioinformatics pipeline

A number of pipelines and software packages are available for analysing 16S rRNA gene amplicons, most notably QIIME, Mothur, and UPARSE, which are all highly flexible and versatile. One study concluded that QIIME, Mothur and MG-RAST produced comparable results when analysing stool samples, and similar microbial composition was found irrespective of the pipeline chosen.

A typical pipeline starts with quality-filtering and de-multiplexing of raw sequencing reads into appropriate sample bins. This process is possible as barcode sequences of short DNA sequences, unique to each sample, have been added to the primers before amplification. A typical approach following this step is to cluster reads into operational taxonomic units (OTUs), which are grouped on the basis of DNA sequence similarity. A 97% similarity level is usually applied as this is considered suitable for binning reads derived from the same species. The main methods for this are *de novo*, closed-reference and open-reference clustering, each with their own set of advantages and disadvantages. *De novo* methods cluster reads without the use of a reference database, whereas the opposite is true for the closed-reference clustering (exclusively reference based). The latter method is much faster because reads with no alignment to the database are discarded. However, some evidence suggests *de novo* clustering is the optimal method. Open-reference clustering combines both *de novo* and closed reference methods, in which any reads discarded during closed reference are subjected to *de novo* clustering, leading to some performance improvements. One comparison of older and newer clustering methods found the newer methods superior

to the more traditional methods. Another important step in the analysis of marker gene sequences is the removal of chimeric sequences, a known artefact of PCR amplification in which hybrid sequences are produced from unrelated parent sequences. The majority of chimaeras are relatively easy to detect using existing reference databases, and software applications for detection (integrated into the commonly used pipelines QIIME and Mothur) include UCHIME and ChimeraSlayer, although all methods might not be 100% effective. QIIME also provides a framework for taxonomic assignment to the created OTUs. The most commonly used classifiers are Mothur and the RDPclassifier. Taxonomic classification requires a reference database of previously classified sequences, such as SILVA91 (16S and 18S rRNA genes), Greengenes (16S rRNA gene), Ribosomal Database Project (16S rRNA gene) and UNITE (ITS region). These classifiers are very accurate to family taxonomic ranks and are often accurate to genus level with some occasional misclassifications for obligate anaerobes (due to lack of reference sequences) and Enterobacteriaceae (due to highly similar reference sequences from different genera), but lose much of the specificity and accuracy at species level. More dedicated species-classifiers such as SPINGO or UTX are better options for higher taxonomic resolution. Sequencing errors can greatly affect OTU clustering performance, and methods removing erroneous reads or alternatives to clustering are desirable (Fig 1).

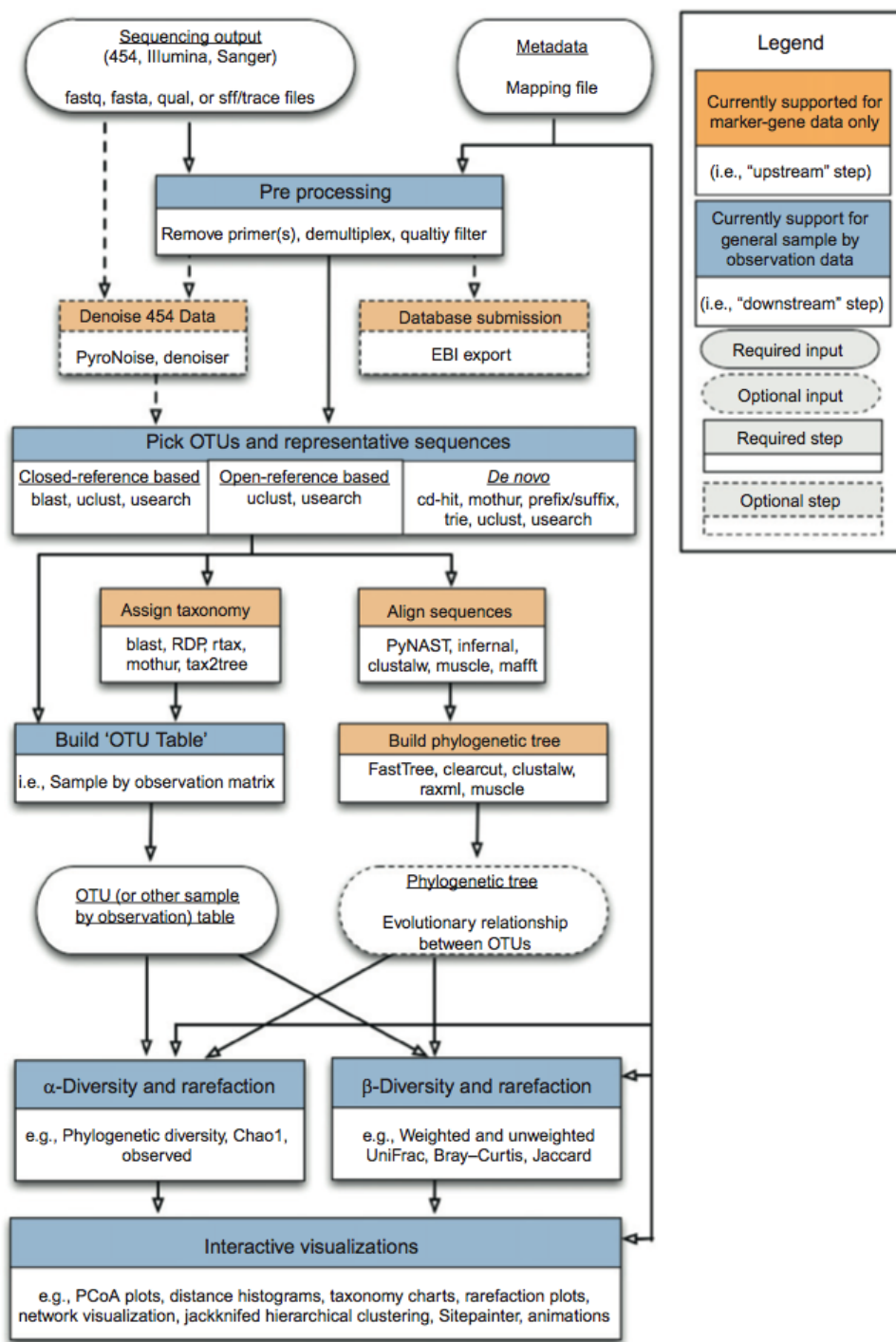


Fig 1: General Workflow for metagenomic data analysis: Figure source: <https://sites.google.com/site/knightslab/wiki/qiime-workflow>

Functional analysis.

Once a metagenome is assembled, genes within the assembled contigs are then identified, which is a prerequisite for deducing the functional potential (annotation) of a microbiome. Although a number of tools exist for this task, such as MetaGeneMark68, FragGeneScan69 and MetaProdigal, the computational challenges they address (with variable success) are greater than for single genomes. As the metagenome assemblies are generally of poorer quality than for single genomes, even when performed at an optimal level, many genes still remain fragmented and incomplete. Following gene identification (gene calling), functional annotation is carried out using computationally demanding homology-based searches (often BLAST71) against databases of orthologues (EggNOG72, COG or KOG73), enzymes (KEGG74) and protein domains and families (Pfam75, TIGRFAMs76, InterPro77).

Reference

1. J Gregory Caporaso, Justin Kuczynski, Jesse Stombaugh, Kyle Bittinger, Frederic D Bushman, Elizabeth K Costello, Noah Fierer, Antonio Gonzalez Pena, Julia K Goodrich, Jeffrey I Gordon, Gavin A Huttenhower, Scott T Kelley, Dan Knights, Jeremy E Koenig, Ruth E Ley, Catherine A Lozupone, Daniel McDonald, Brian D Muegge, Meg Pirrung, Jens Reeder, Joel R Sevinsky, Peter J Turnbaugh, William A Walters, Jeremy Widmann, Tanya Yatsunenko, Jesse Zaneveld and Rob Knight. QIIME allows analysis of high-throughput community sequencing data; *Nature Methods*, 2010; doi:10.1038/nmeth.f.303
2. Schloss PD, Westcott SL, Ryabin T, Hall JR, Hartmann M, Hollister EB, Lesniewski RA, Oakley BB, Parks DH, Robinson CJ, Sahl JW. Introducing mothur: open-source, platform-independent, community-supported software for describing and comparing microbial communities. *Appl. Environ. Microbiol.* 2009 Dec 1;75(23):7537-41.

8. MOTHUR

Dr. Muhammed Manzoor A.P

Dr. Sudeep Ghate

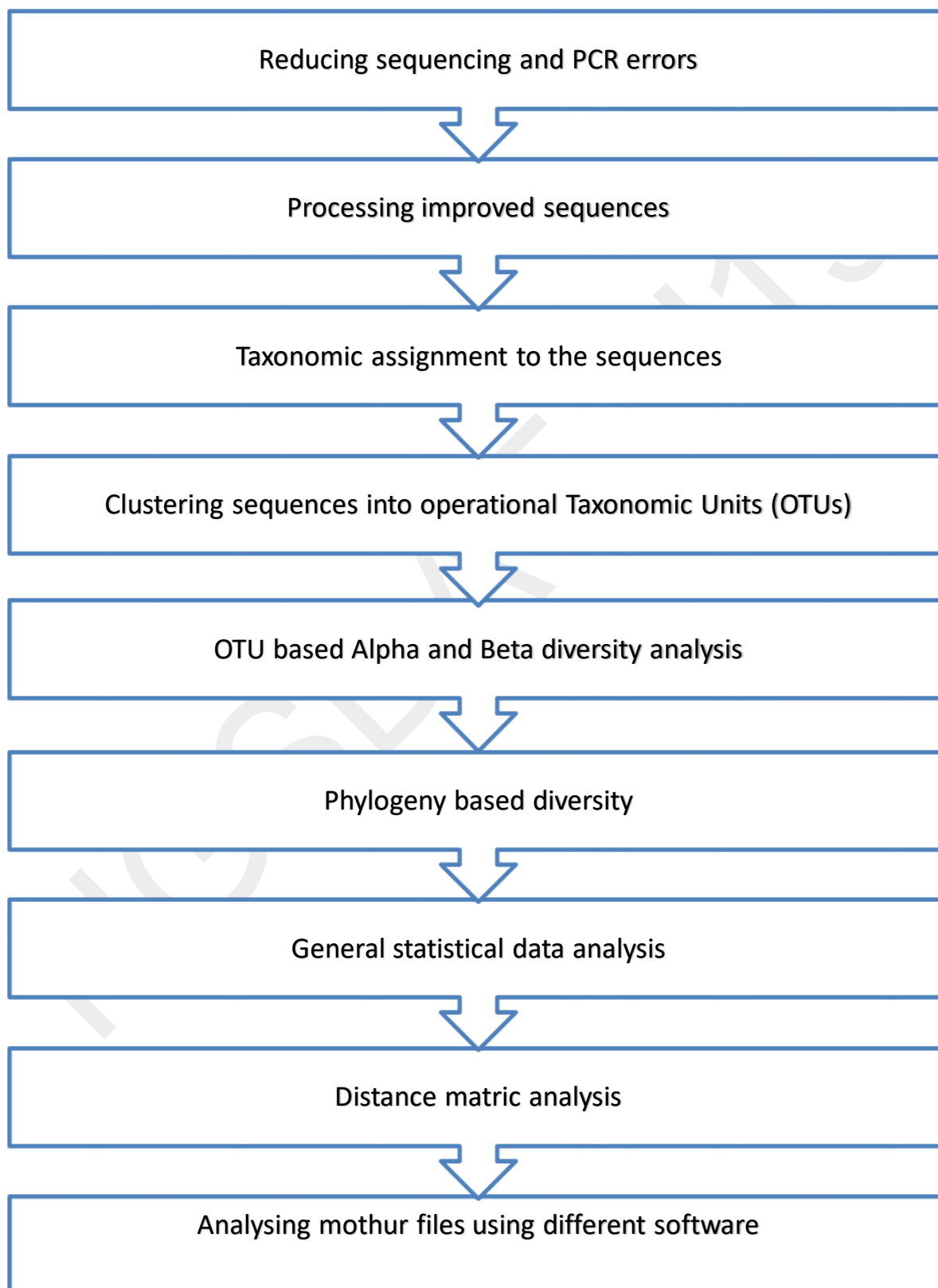
Prerequisite before getting started with mothur software

- The latest version of mother can be downloaded from <https://github.com/mothur/mothur/releases/tag/v1.41.3>
- Getting the data the (.fastq) from the NGS machine or download the data from the sequencing company's server (in this case illuminaMiseq data) or download from the NCBI SRA.
- After downloading: Check the quality of the data. FastQC software (offline/online) which is listed on galaxy server (www.usegalaxy.org).
- Trimming the sequences which fall below quality (poor sequences): Trimmomatic tools (listed under galaxy pipeline).

Here we are using paired end fastq file (both forward and reverse fastq files for individual sample). Fastq files contain both the sequence data and the quality score data.

After obtaining the sequences from the sequencing company's web server, first thing we need to do is to decompress these files because the company gives us the files in the ".GZ" format. Now keep all the ".fastq" files into one folder, in addition, mother software, uchime, silva reference files etc. on the same folder

Before getting started with mothur software. After obtaining our sequences from the sequencing company's web server, first thing we need to do is to decompress those files because the company gives us the files in the ".GZ" format. Now put all the ".fastq" files into one folder, also put mothur software, uchime, silva reference files etc. on the same folder before getting started with mothur software.

WORKFLOW:

Before initiating the analysis we need to know about the quality of the sequences on which we will be working on. For this we will use FastQC software which is a simple and universally acceptable tool for NGS quality analysis.

- Open the FastQC folder and double click on the **run_fastqc**(windows batchfile)
- Once the FastQC dialogue box is open go to file menu and click on open and then select the sequence file which you wish to open
- Click on open and wait for a couple of minutes until the analysis reach to 100% and then view the result

Sequences with low quality score will be trimmed or corrected by using Trimmomatic tool listed under galaxy pipelines (web based server).

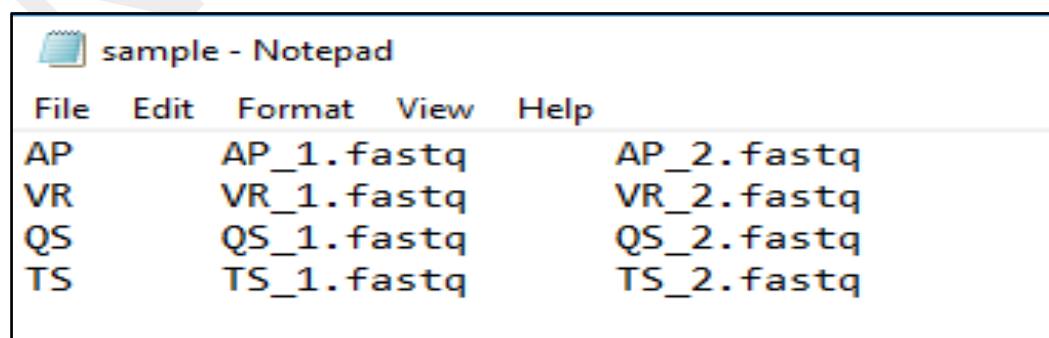
Processing the data using Mothur:

Before starting with mothur few files need to be created by using Notepad++ because mothur needs those files during analysis.

1. Sample.files

Open notepad++ and type the name of your sample as follow

File name Forward fastq file Reverse fastq file


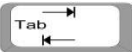



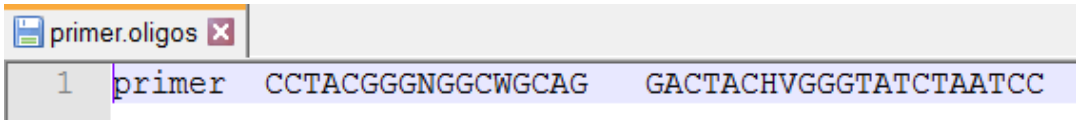
Save the file by the name “sample.files”. You can give any name according to your sample and choice, but the file extension should be

“.files”.

2. Primer.oligos file

This file contains the primers which were used during sequencing. To make this file open Notepad++ and type forward and reverse primer sequences in the format given below

primer fo  re  



```
1 primer CCTACGGGNGGCWGCAG GACTACHVGGGTATCTAATCC
```

Save the file by the name “primer.oligos”.

forward  forward primersequences

3. pcrTest.oligos file

This file also contains the forward and reverse primer sequences which were used for sequencing our samples. To make this file open Notepad++ and type the forward and reverse sequences in the same format given below

forward  forward primersequences

Double click on mothur application for getting started:

If you are putting all the files and sequences in the same folder then you need not to set the directory otherwise the directory can be set by using “set.dir” command. Once you set the directory then mothur will take the files from that directory and also it will save the output file in the same directory, otherwise mothur will take input files

from the same directory in which it is placed and also it will save the output file in the same directory.

```
mothur>set.dir
```

It will generate

1. Current RAM usage: TotalRam:
2. Current input directory saved by mothur:
3. Current default directory saved by mothur:
4. Current working directory:

```
mothur>set.dir(input=input directory name, output=output directory name)
```

Reducing sequencing and pcr errors:

Making contigs:

We are having paired end Illumina MiSeq sequences in the form of “.fastq” files. The first thing we need to do is to combine the two sets of reads for each sample and then to combine the data from all the samples. This is done by using “make.contigs” command, which requires “data.files” file (which was created earlier) as input. This command extracts the sequence and quality score data from the fastq files; it also creates the reverse complement of the reverse fastq reads and then joins the reads into contigs. Here we are using processor 4 to reduce the time (we can also use 8 or above processors according to your computer configuration).

```
mothur>make.contigs(file=sample.files,processors=4) 
```

The first thing you'll see is that it processes the fastq files to generate the individual fasta and qual files. Then it will go through each set of files and make the contigs.

This command will generate fasta and group files which will be used in further steps.

Now let's see how these sequences look like using the "summary.seqs" command:

```
mothur>summary.seqs(fasta=sample.trim.contigs.fasta) or
```

```
mothur>summary.seqs(fasta=current)
```

mothur remembers the latest files as well as no of processors we have used. So if we type fasta=current then mothur will take the latest fasta file and if we type group=current, mothur will take the recent or current group file and soon.

We can run this summary.seqs command whenever we need to know the result of any step during the sequence processing; it is recommended to run this command after each step.

Trimming primers:

Our sequences contain the primers also and therefore those need to be removed from the sequences. This can be done by using "trim.seqs" command, this command needs "primer.oligos" file:

```
mothur>trim.seqs(fasta=current, oligos=primer.oligos, pdiffs=2, flip=T)
```

pdiffs is maximum number of differences to the primer sequence, default=0. Flip option is used to calculate the reverse complement of the sequence. We can also use "reverse.seqs" command for the same purpose.

```
mothur>summary.seqs(fasta=current)
```

Making groupfile

The group file is needed in so many steps and therefore the group file needs to be

created. We do the same by using list.seqs command which writes out the names of the sequences found within a fasta file. The list.seqs command will write out the names of the sequences found within a fastq, fasta, name, group, count, list, or align.report file. The output group file will be useful in further steps:

```
mothur>list.seqs(fasta=current) 
```

```
mothur>get.seqs(group=current,accnos=current) 
```

Screening thesequences

Our file contains a number of sequences, some are very long while some are very small. We need to get rid of those sequences and therefore we use “screen.seqs” command. The screen.seqs command enables you to keep sequences that fulfil certain user defined criteria. This command also removes any sequences with ambiguous bases.

```
mothur>screen.seqs(fasta=current, group=current,  
minlength=...) mothur >summary.seqs(fasta=current)
```

```
mothur>screen.seqs(fasta=current, group=current,  
maxlength=...) mothur >summary.seqs(fasta=current)
```

```
mothur>screen.seqs(fasta=current, group=current,  
maxambig=0) mothur >summary.seqs(fasta=current)
```



```
mothur>screen.seqs(fasta=current, group=current,  
maxhomop=8) mothur >summary.seqs(fasta=current)
```

Processing improved sequences:

Extracting uniquesequences

Now we have sequences which does not contain the primers as well as any longer or shorter reads. We anticipate that many of our sequences are duplicates of each other. Because it's computationally wasteful to align the same thing a zillion times, we'll unique our sequences using the unique.seqs command Iftwo sequences have the same identical

sequences then they are considered duplicates and get merged.

```
mothur>unique.seqs(fasta=current
)  mothur
>summary.seqs(fasta=current)

```

Now we will run `count.seqs` command to generate a table where the rows are the names of the unique sequences and the columns are the names of the groups. The table is then filled with the number of times each unique sequence shows up in each group. We will need a name and a group file to run this command:

```
mothur>count.seqs(name=current,
group=current)          mothur
>summary.seqs(count=current)
```

Aligning sequences with referencedatabase

We can see that now our file contains only unique sequences and therefore we need to align those sequences with reference alignment. Before aligning our sequences we need to get the reference database customized to our region of interest. This will save a lot of time during alignment processing. To run this command we need the reference database (`silva.bacteria.fasta`) file and know where in the alignment our sequences start and end. We use `pcr.seqs` command for the same; this command also needs “.oligos” file (`pcrTest.oligos`):

```
mothur>pcr.seqs(fasta=silva.bacteria.fasta,
oligos=pcrTest.oligos) mothur
>summary.seqs(fasta=current)
```

This `summary.seqs` command will give us the start and end position of our sequences against the reference database.

We now need to cut the reference file to our region of interest and this is also done by using `pcr.seqs` command:

```
mothur>pcr.seqs(fasta=current,start=6428,end=23440)
```



```
mothur >summary.seqs(fasta=current)
```



Now rename the fasta file according to the primers used for the sequencing. This is simply done by rename.file command ([this is optional](#)).

```
mothur>rename.file(input=silva.bacteria.pcr.fasta,new=silva.v3.fasta)
```



```
mothur>summary.seqs(fasta=current)
```



Now the reference database has been customized to align our sequences. We will do the alignment by using align.seqs command:

```
mothur>align.seqs(fasta=sample.trim.contigs.trim.good.good.good.good
.unique.fasta, reference=silva.bacteria.pcr.pcr.fasta, flip=T)
```

```
mothur>summary.seqs(fasta=current, count=current)
```

since our sequence is aligned with the reference database we will run screen.seqs command again to make sure that everything overlaps the same region.

```
mothur > screen.seqs(fasta=current, count=current, summary=current,
start=6430,
end=23440, maxhomop=8)
```

```
mothur>summary.seqs(fasta=current, count=current)
```

Now we will filter the sequences by using filter.seqs command to remove the overhangs at both ends. In this way we can also filter sequences which does not overlap the same alignment coordinates. Also terminal and vertical gaps can be removed by using “trump=.” and “vertical=T” parameter respectively.

```
mothur>filter.seqs(fasta=current, vertical=T, trump=.)
```

By trimming the ends of the sequences some redundancy may have created. To overcome this we will re-run unique.seqs command:

```
mothur>unique.seqs(fasta=current,  
count=current) mothur  
>summary.seqs(fasta=current, count=current)
```

Next we need to de-noise the sequences and this is done by using the pre.cluster command. Here we will allow up to 4 differences between sequences (1 difference per 100 bp and we have 400bp sequence).

```
mothur>pre.cluster(fasta=current, count=current,diffs=4) 
```

Chimeraremoval

We started with lacks of sequences but now we have only few thousand sequences which are good and unique. Now we need to remove the chimera from our sequences and this is done either by chimera.uchime or chimera.vsearch command. Chimera is defined as the single cDNA sequence which originates from two transcripts. It is usually considered as contamination.

```
mothur>chimera.uchime(fasta=current, count=current,dereplicate=t) 
```

Or

```
mothur>chimera.vsearch(fasta=current, count=current, dereplicate=t)
```

If the sequence is found to be chimeric in one sample then “dereplicate=t” will not allow the same sequence to be removed from all samples.

Here we removed the chimeric sequences from the count file but we also need to remove the same from the fasta file. We do this by using remove.seqs command:

```
mothur>remove.seqs(fasta=current,  
accnos=current)          mothur  
>summary.seqs(fasta=current, count=current)
```


Taxonomic assignment

Next we will use `classify.seqs` command to classify the sequences. Once the sequences are classified we will use `remove.lineage` command to remove the undesirables from our dataset.


```
mothur>classify.seqs(fasta=current, count=current, reference=silva.bacteria.fasta,  
taxonomy=silva.bacteria.silva.tax, cutoff=80)
```

```
mothur>remove.lineage(fasta=current, count=current, taxonomy=current, taxon=  
Chloroplast-Mitochondria-unknown-Archaea-Eukaryota)
```

Now we need to create the summary file with the undesirables removed. For this we use `summary.tax` command:

```
mothur>summary.tax(taxonomy=current,count=current) 
```

Or

```
mothur>summary.seqs(fasta=current,count=current) 
```

Clustering the sequences into OTUs

Till now we have removed almost all the bad data from our datasets. It is time to cluster all the sequences into OTUs. For this work we will use `dist.seqs` and `cluster` command. While `dist.seqs` calculates the distances between the sequences, `cluster` command will cluster those distance matrix file into OTUs.

```
mothur>dist.seqs(fasta=current,  
cutoff=0.03) mothur >  
cluster(column=current,  
count=current)
```

Distance matrix file can be generated by two means. First is column formatted dist file and another is phylip formatted dist file.

Phylip formatted dist file can be generated by the following command:

```
mothur>dist.seqs(fasta=current, output=lt)
```

Our dataset may contain the singletons or rare OTUs. To remove these we will now use `remove.rare` command:

```
mothur>remove.rare(list=current, count=current, nseqs=1, label=0.03)
```


Further we want to know that what is the number of sequences each OTU contains from each group. This can be done by using `make.shared` command:

```
mothur>make.shared(list=current, count=current, label=0.03)
```


We also want to know about the taxonomy for each OTUs and we can do the same by using the `classify.otu` command. This command will give us consensus taxonomy file for each OTU:

```
mothur>classify.otu(list=current, count=current, taxonomy=current, label=0.03)
```

Now we need to find the number of sequences in each sample and this will be done by using `count.groups` command:

```
mothur>count.groups(shared=current) 
```

This is time to subsample and rarefying our dataset. We will create a subsampled file in which each group contains the same number of sample. This will be done by using `sub.sample` command:

```
mothur>sub.sample(shared=current,size=37274) 
```

OTU-based analysis:

(a) Alpha diversity

The mean species diversity in sites or habitat at a local scale is known as alpha diversity.

First of all we will generate the rarefaction curve which describes the number of OTUs observed as a function of sampling effort. We will use `rarefaction.single` command to

do the same:

```
mothur>rarefaction.single(shared=current, calc=sobs, freq=100)
```

Note- we can create rarefaction file from both the shared files (normal shared file and subsampled shared file)

```
mothur>summary.single(shared=data.trim.contigs.trim.good.good.good.unique.good.filter.unique.precluster.pick.pick.opti_mcc.0.03.pick.shared, calc=nseqs-coverage-sobs-chao-shannon-simpson, subsample=37274)
```


Now we sequences from the entire dataset will be extracted for building a phylogenetic tree:

```
mothur>get.oturep(list=current, fasta=current, count=current, label=0.03, method=abundance)
```

(B) Beta diversity

The ratio between regional and local species diversity is called beta diversity.

Now we will prepare a heatmap of the relative abundance of each OTU across the samples. This will be done by using heatmap.bin command and log2 scaling the relative abundance values. There are thousands of OTUs, here we will just look at the top 50 or 100 OTUs.

```
mothur>heatmap.bin(shared=data.trim.contigs.trim.good.good.good.unique.good.filter.unique.precluster.pick.pick.opti_mcc.0.03.pick.0.03.subsample.shared,scale=log2,numotu=50) 
```

This command generates an SVG-formatted file which can be visualized by using any of the web browser softwares like chrome, internet explorer etc..

Some other commands to generate heatmap is:

```

mothur>dist.shared(shared=data.trim.contigs.trim.good.good.good.unique.good.filter.
unique.precluster.pick.pick.opti_mcc.0.03.pick.shared, calc=thetayc-jclass,
subsample=37274)

```

```

mothur>heatmap.sim(phylip=data.trim.contigs.trim.good.good.good.unique.good.filter
.unique.precluster.pick.pick.opti_mcc.0.03.pick.thetayc.0.03.lt.ave.dist)

```

```

mothur>heatmap.sim(phylip=data.trim.contigs.trim.good.good.good.unique.good.filter
.unique.precluster.pick.pick.opti_mcc.0.03.pick.jclass.0.03.lt.ave)

```

The above mentioned command will generate two SVG-formatted files. In this heatmap the intensity of red color indicate communities that are more similar than those with black colors.

Lets now create the venn diagram for our samples. The venn diagram shows shared and non shared out between the sequences. We will do this by venn command:

```

mothur>venn(shared=subsample.shared file, group=1-2-3-4)

```

Now we will generate a dendrogram to describe the similarity of the samples to each other. This will be done by using the jclass and thetayc calculators within the tree.shared command:

```

mothur>tree.shared(phylip=data.trim.contigs.trim.good.good.good.unique.good.filter.
unique.precluster.pick.pick.opti_mcc.0.03.pick.thetayc.0.03.lt.ave.dist)

```

Some other statistical analysis using mothur:

```

mothur>pcoa(phylip=data.trim.contigs.trim.good.good.good.unique.good.filter.unique.
precluster.pick.pick.opti_mcc.0.03.pick.thetayc.0.03.lt.ave.dist) 

```

```

mothur>nmds(phylip=data.trim.contigs.trim.good.good.good.unique.good.filter.unique.
precluster.pick.pick.opti_mcc.0.03.pick.thetayc.0.03.lt.ave.dist) 

```


```

mothur>corr.axes(axes=data.trim.contigs.trim.good.good.good.unique.good.filter.
unique.precluster.pick.pick.opti_mcc.0.03.pick.thetayc.0.03.lt.ave.pcoa,

```

```
shared=data.trim.contigs.trim.good.good.good.unique.good.filter.unique.precluster.pick
```

```
.pick.opti_mcc.0.03.pick.0.03.subsample.shared,method=spearman,numaxes=3)
```

```
 mothur>quit()
```

Mothur generated files (shared file, taxonomy file, metadata file etc.) will be further used for analysis using different software packages.

References:

- Edgar, R. C., Haas, B. J., Clemente, J. C., Quince, C., and Knight, R. (2011) UCHIME improves sensitivity and speed of chimera detection. *Bioinformatics* 27: 2194-2200.
- Rognes, T., Flouri, T., Nichols, B., Quince, C., & Mahé, F. (2016). VSEARCH: a versatile open source tool for metagenomics. *PeerJ*, 4, e2584..
- Kozich, J. J., Westcott, S. L., Baxter, N. T., Highlander, S. K., and Schloss, P. (2013) Development of a dual-index sequencing strategy and curation pipeline for analyzing amplicon sequence data on the MiSeq Illumina sequencing platform. *Applied and Environmental Microbiology* 79:5112-5120.
- Schloss, P. D., Westcott, S. L., Ryabin, T., Hall, J. R., Hartmann, M., Hollister, B., Lesniewski, R. A., Oakley, B. B., Parks, D. H., and Robinson, C. J. (2009) Introducing mothur: open-source, platform-independent, community-supported software for describing and comparing microbial communities. *Applied and Environmental Microbiology* 75:7537-7541.

8. QIIME

Mr. Kumar Arvind

QIIME (canonically pronounced "chime") stands for Quantitative Insights into Microbial Ecology, is an open-source bioinformatics pipeline, used for performing microbiome analysis from raw DNA sequencing data. In fact, it is designed to take users from raw sequencing data generated on the Illumina or other platforms through publication quality graphics and statistics.

QIIME is extremely flexible and can accommodate various sequencing technologies and methods of data analysis and allow for robust analysis of millions of sequences. The QIIME pipeline allows you input files directly from the sequencing instrument, demultiplex barcoded samples, generate an OTU table, and perform downstream diversity and statistical analyses.

The basic workflow starts with filtering the reads to ensure good quality data and splitting the libraries to match barcoded samples to the appropriate metadata followed with the generation of an OTU table, which is often done by clustering the reads against the Greengenes reference database; a process which greatly speeds computation. After the OTU table is generated and assigned taxonomies, various downstream analyses can be implemented. QIIME offers support for a number of alpha and beta diversity metrics, data visualization, and multivariate statistics. Furthermore, files generated in QIIME can be used with several other software packages, including Microsoft Excel, PC-ORD, Primer E, and **R. Getting started:-**

Raw data processing

- ⊗ Quality Filtering -- *filter out low base quality and rename samples*
- ⊗ Joined, condensed and split paired files.

Taxonomy/Diversity Analysis

- ⊗ Get OTU cluster of similar sequences

- ⌘ Select one representative sequences for each OTU
- ⌘ Annotate OTU cluster with known taxonomy (consider OTU representative sequences without database hits as "unknown")
- ⌘ Create table (biom format) containing OTU abundance of all samples, and taxonomy information for further analysis

Taxonomy/Diversity Analysis

- ⌘ UniFrac beta diversity analysis
- ⌘ Alpha diversity analysis
- ⌘ Testing for statistical difference

NGSDAT

TOOL LINKS

MOTHUR <https://www.mothur.org/>

QIIME <https://qiime2.org/>

MicrobiomeAnalyst <https://www.microbiomeanalyst.ca/>

Krona <https://github.com/marbl/Krona/wiki>

NGSDAT '19

NOTES

NGS DAT '19

NGSDAT '19
