DBT Sponsored Training Program on

Next Generation Sequencing: Data Analysis & Annotation

March 17 -20, 2014

TRAINING MANUAL

INDIAN INSTITUTE OF SPICES RESEARCH
(Indian Council of Agricultural Research)
Kozhikode – 673012, KERALA

Published by

M. Anandaraj
Director

Organized by

Bioinformatics Centre, Indian Institute of Spices Research, Kozhikode

Compiled & Edited by

Santhosh J Eapen
Rosana O B
Navish Kumar Batchu
Habeeba N T
Jithin Subash

Preface

Next Generation Sequencing: Data Analysis & Annotation

The possibilities offered by next generation sequencing (NGS) platforms are revolutionizing biotechnological laboratories. Recent advances in NGS technologies have moved whole genome sequencing from large centres to small research groups and even individual scientists. Their low cost, simplicity of library generation and instrument operation, and quantity of data generated have made the NGS technologies, alone or in combination, an attractive choice for genome sequencing projects. But most NGS methods produce relatively short sequence fragments and hence data analysis is a major issue to be overcome in order to fulfill their true potential. Besides, the massive data produced by NGS also presents a significant challenge for data storage, and management solutions. These short sequences need to be assembled, a computational process in which they are compiled into whole genome sequences followed by gene prediction/annotation. Though genome sequencing is growing rapidly, quality of the assemblies made on many occasions is lower than the Sanger standards, reflecting the quality of the sequencing technology used, but also of the analysis software employed for assembly and annotation. Further adding to the bioinformatics problems, there are differences among the various NGS platforms in term of data format, length of reads, etc., which results in the need for diversity in bioinformatics including sequence quality scoring, alignment, assembly, and data processing. Though there are a number of service providers, a true researcher should invariably do a close scrutiny of the sequence data generated to verify its quality as well as accuracy.

The short-term training 'Next Generation Sequencing: Data Analysis and Annotation' is the third in this seriesand the twenty seventh program that Bioinformatics Centre is organizing. The main objectives of this training program is provide an exposition of various steps involved and software available for NGS data analysis and to provide hands on training on NGS data analysis and their annotation. The experience earned while assembling and annotating the eukaryotic plant pathogen *Phytophthoracapsici* will be shared. The feedback from two earlier programs on the same topic has helped us to refine the contents and structure of this short course. Hence, we strongly feel that this short-term training will provide vital information on the latest research and technology developments in the domain area and will offer clean competitive benefits to your research. We hope that this compendium will be useful for the participants during the course of this training program as well as in their future endeavours.

Santhosh J. Eapen
Course Coordinator

DBT Sponsored Training Program

Next Generation Sequencing: Data Analysis and Annotation

March 17 -20, 2014

Schedule

	Monday, 17 March 2014
09.30 am	Registration
10.15 am	Inauguration and address
11.15 am	Tea/Coffee & Group Photo
11.30 am	Pre-evaluation *by P. Rajeev, Senior Scientist, IISR, Calicut*
12.00 pm	Next Generation Sequencing (L) *by A. I. Bhat, Principal Scientist, IISR, Calicut*
01.30 pm	Basic Linux Commands (L&P) *by Jithin Subash, Research Scholar, IISR, Calicut*
03.30 pm	NGS Data Formats (L) *by Rosana O. B., Research Scholar, IISR, Calicut*
	Tuesday, 18 March 2014
09.30 am	Quality Control (L&P) *by* Anantharamanan R., Scientist A, Rubber Research Institute of India, Kottayam
01.30 pm	NGS Data Analysis – Genome Alignment (L&P) *by* Anantharamanan R., Scientist A, Rubber Research Institute of India, Kottayam
	Wednesday, 19 March 2014
09.30 am	Genome Assembly *by* Anantharamanan R., Scientist A, Rubber Research Institute of India, Kottayam
01.30 pm	Variant Annotation (L&P) *by* Anantharamanan R., Scientist A, Rubber Research Institute of India, Kottayam
	Thursday, 20 March 2014
09.30 am	Plant Transcriptome Mapping (L) *by D. Prasath, Senior Scientist, IISR, Calicut*
10.30 am	Genome Annotation *by Rosana O. B., Research Scholar, IISR, Calicut*
01.30 pm	Comparative genomics (L & P) *by Navish Kumar B. Research Scholar, IISR, Calicut*
02.30 pm	Post Evaluation & Feedback *by P. Rajeev, Senior Scientist, IISR, Calicut*
03.00 pm	Transcriptome Data Analysis Using CLC bio Genomic Workbench (L & P) *by Uday Deshpande, Field Application Scientist, CLC Bio India Pvt. Ltd,. Visakhapatnam*
04.00 pm	Valedictory Chief guest: Dr. M. Anandaraj, Director, IISR, Calicut

CONTENTS

Chapter 1

Next Generation Sequencing Technologies

Dr. A. I. Bhat

Introduction

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. As evidenced throughout this review, 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 human genome in terms of genetic variations, some of which may be confined to susceptible loci for some common human conditions. This chapter reviews 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 and bioinformatics.

Since the time DNA was discovered as the code to all biological life on earth, man has sought to unravel its mysteries. If the genetic code could be sequenced or "read", the origins of life itself may be revealed. Although this thought might not be entirely true, the efforts to date made have certainly revolutionized the biological field. The "original" sequencing methodology, known as Sanger chemistry, uses specifically labeled 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 methods 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 basepairs (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 CYP2D6 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". NGS is a rapidly evolving technology that is changing on an almost daily basis. The purpose of this review is to highlight these advances and bring the reader up to date on the latest technological achievements in DNA sequencing technologies, particularly as related to genomics. Following completion of the HGP, a new approach, genome-wide association study (GWAS), was widely applied to genomics. Although several early GWAS studies reported potentially promising results, the majority of GWAS studies were disappointing because of inadequate sample size, limitation of arrays for certain genetic variations, and/or heterogeneity in phenotype. These obstacles may be overcome by new genomic technology, i.e., next-generation sequencing (NGS), also known as massively parallel sequencing or multiplex cyclic sequencing. Since many genetic variants which contribute to many human conditions are still unknown, unbiased whole-genome sequencing will help to identify these genetic variants, including single nucleotide variants (SNVs) or single nucleotide polymorphisms (SNPs), small insertions and deletions (indels, 1–1000 bp), and structural and genomic variants (>1000 bp).

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, Polonator G.007 and the Helicos HeliScope 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. We also describes 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 labeled 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 labeled 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 NIH-PA Author Manuscript NIH-PA Author Manuscript NIH-PA Author Manuscript new and exciting technologies, faster sequencing speed, and a more affordable price. Methodologies used by each of the current available NGS systems are 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 labeled 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) ABI SOLiD platform

The ABI SOLiD platform uses a unique sequencing-byligation approach in which it uses an emulsion PCR approach with small magnetic beads to amplify the DNA fragments for parallel sequencing. During SOLiD sequencing, DNA ligation is carried out to link specific fluorescent labeled 8-mer oligonucleotides for "dinucleotide-encoding", whose 4^{th} and 5th bases are encoded by specific fluorescence. Each fluorescent marker on a 8-mer identifies a two-base combination, which can be further distinguished with a universal primer offsetting scheme. The primer offsetting scheme allows a universal primer that is offset by one base from the adapter-fragment position to hybridize to DNA templates in five cycle sets permitting the entire fragment to be sequenced and each base position sequenced twice during each cycle. Each ligation step is followed by fluorescence detection and another round of ligation. SOLiD4 analyzer has a read length of up to 50 bp and can produce 80–100 Gbp of mappable sequences per run. The latest model, 5500×l solid system (previously known as SOLiD4hq) can generate over 2.4 billion reads per run with a raw base accuracy of 99.94% due to its 2-base encoding mechanism. This instrument is unique in that it can process two slides at a time; one slide is receiving reagents while the other is being imaged. The SOLiD4 platform probably provides the best data quality as a result of its sequencing-by-ligation approach but the DNA library preparation procedures prior to sequencing can be tedious and time consuming. The newly marketed EZ-Bead system may provide some resolution to this problem.

(iv) Danaher/Dover/Azco Polonator G.007

The Danaher/Dover/Azco Polonator G.007 is a new platform on the market with emphasis on competitive pricing. The Polonator platform employs a sequencing-by-ligation approach using a randomly arrayed, bead-based, emulsion PCR to amplify DNA fragments for parallel sequencing. The short-read length is 26 bp, and 8–10 Gbp of sequence reads are generated per run, with 92% of the reads mappable. The random bead-based array will likely be replaced with their patented rolonies technology (rolling circle colonies) on an ordered array to increase accuracy and improve read length.

(v) Helicos HeliScope

The Helicos HeliScope platform is the first single molecular sequencing technology available that uses a highly sensitive fluorescence detection system to directly detect each nucleotide as it is synthesized. The distinct characteristic of this technology is its ability to sequence single DNA

molecules without amplification, defined as Single-Molecule Real Time (SMRT) DNA sequencing. The short-read length ranges from 30 bp to 35 bp at present time, with a raw base accuracy greater than 99%, and 20–28 Gbp of potential sequence reads per run in the near future.

The advantage of single-molecule DNA sequencing technology is its potential to read extremely long sequences and fast sequencing speed, which could translate into a dramatic reduction in overall sequencing cost. As such, advanced single DNA molecule sequencing technology has been defined as the next-NGS technology. However, the basic philosophy of massive parallel sequencing is still the same and the term next- generation sequencing (NGS) will only be used in this review. More detailed technical description of these platforms is available elsewhere.

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, where sequencing the human genome at high fold coverage for less than $100 is now a reasonable goal for the community.

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. These technologies are at varying stages of development, making the writing of a review on TGS difficult given there is still much to prove regarding the utility of many of the TGS technologies.

However, 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 resequence genomes as needed for any application. Here we discuss many of the emerging TGS technologies that have the potential to make such stunning advances possible.

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) SMS sequencing 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 DNAsequencing-by-synthesis. Nano-technologies have long been considered a cutting-edge technology for single-molecule DNA sequencing and several nanopore sequencing concepts and technologies are currently under development. One concept is based on the observation that when a DNA strand is pulled through a nanopore by an electrical current, each nucleotide base (A, T, C, G) creates a unique pattern in the electrical current. This unique nanopore electrical current fingerprint can be used for nanopore sequencing.

(iii) Fluorescence-based single-molecule sequencing

Pacific BioSciences is developing a single-molecule real time (SMRT) DNA sequencing technology. This approach performs single-molecule sequencing by identifying nucleotides which are phospholinked with distinctive colors. During the synthesis process, fluorescence emitted as the phosphate chain is cleaved and the nucleotide is incorporated by a polymerase into a single DNA strand.

(iv) 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.

(v) 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 tunneling microscopy (STM) can reach atomic resolution, STM for single-molecule sequencing is being explored. LightSpeed 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

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 byproduct. 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.

Focusing on resequencing specific sections of the human genome combined with genome- region enrichment, Genizon BioSciences is developing a sequencing-by-hybridization technology based

on known reference sequences. Avantome (acquired by Illumina) is also exploring the single-molecule sequencing technologies.

Road to the personal genome project

Since the initiation of 1000 genome project the cost of sequencing an individual genome has been rapidly decreasing and will likely reach $1000 per person within a short period of time, making personalized medicine become a possible reality. In genomics, the personal genome era made available by NGS technologies will mark a significant milestone in entire genomic research field in the foreseeable future. It is not clear which NGS technology will eventually dominate the genomic research field, but it is almost certain that further reductions in cost, rapid increases in sequencing speed with improved accuracy, and the advantages conferred by these new technologies will assure that NGS will become an essential molecular tool affecting all aspects of the biological sciences. 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
Currently available				
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/Heliscope	No	35 (25–55) bp	21–37 Gbp/run	True single-molecule sequencing (tSMS)
In development				
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
Genovoxx	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

Current strategies for the NGS project

To ensure the correct identification of genetic variants, short-read coverage must be sufficient to ensure the complete and accurate sequence assembly. Currently, at least 30× coverage is recommended in whole-genome scans for rare genetic variants in human genomes, which is a burden on computer resources and cost management. Although the cost of whole-genome sequencing has dropped substantially, the cost remains a major obstacle; whole-genome sequencing of a single individual currently costs approximately $100,000.00. By targeting specific regions of interest, selective DNA enrichment techniques improve the overall cost and efficiency of NGS; however, targeted enrichment must maintain uniform coverage, high reproducibility, and no allele bias for any genomic region. Targeted sequencing generally focuses on all protein-coding subsequences (the functional exome), which only requires ~5% as much sequencing compared to that required for the entire human genome. This strategy currently reduces the overall cost to around $10,000 or less for the sequencing of a single individual. An important consideration to the cost of such experiments is the depth of sequence coverage required to achieve a desired sensitivity and specificity of at least 25-fold nominal sequence coverage. The most common techniques for targeted sequence enrichment are either microarray-based or solution hybrid- based. Several targeted selection technologies have been marketed and successfully applied in different NGS projects with variable success and may become the tools of choice to lower the burden of time and cost. For example, using targeted selection strategy, the mutations in DHODH from four individuals from three unrelated families with Miller syndrome have been successfully identified, illustrating that selective DNA enrichment techniques will dramatically reduce overall cost and accelerate discovery of genetic variants that cause rare and yet to be discovered genetic disorders. Other genetic loci for rare diseases have also been successfully identified through exome sequencing, further validating this strategy. Commercially available products for targeted sequence-enrichment include Agilent's SureSelect and NimbleGen's SeqCap/EZ Exome (both array- and solution-based technologies), RainDance and Illumina's TruSeq (solution-based technology), Febit's HybSelect and LC Sciences (microarray-based strategy), Qiagen and Fluidigm (PCR-based method) (Table 3).

Table 3. Targeted sequence- enrichment technologies for NGS.

Technology	Approach	Platform	Website
Agilent/SureSelect	Array- and solution-based	Illumina/Roche/ABI	http://www.chem.agilent.com/
RainDance	Microdroplet-based	Illumina/Roche/ABI	http://www.raindancetechnologies.com/
NimbleGen/SeqCap/EZ Exome	Array- and solution-based	Illumina/Roche/ABI	http://www.nimblegen.com/products/seqcap/index.html
Febit/HybSelect	Microarray-based	Illumina/Roche/ABI	http://www.febit.com/microarray-sequencing/index.cfm
Fluidigm	PCR-based	Illumina/Roche/ABI	http://www.fluidigm.com/targeted-resequencing.html
Mycroarray/Myselect	Solution-based	Illumina/Roche/ABI	http://www.mycroarray.com/products/myselect.html
LC Sciences	Microarray-based	Illumina/Roche/ABI	http://www.lcsciences.com/applications/genomics/
Qiagen/SeqTarget	Long-range PCR-based	Illumina/Roche/ABI	http://www.qiagen.com/products/seqtargetsystem.aspx
Illumina/TruSeq	Solution-based	Illumina/Roche/ABI	http://www.illumina.com/applications.ilmn

Bioinformatics for NGS data

The parallel short-read strategy of NGS opens many challenges for bioinformatics to interpret the short reads and the genetic variations in human genomes. The full benefit of NGS will not be achieved until bioinformatics are able to maximally interpret and utilize these short-read sequences, including alignment, assembly, etc. Typically, tens or hundreds of Gbp short reads can be generated during each run in any given NGS platform. As a result, the average NGS experiment generates terabytes of raw data, making data analysis and management of data problematic. Given the vast amount of data produced by NGS, developing a massive data storage and management solution and creating informatics tools to effectively analyze data will be essential to the successful application of NGS technology. Further adding to the bioinformatics problems, there are differences among the various NGS platforms in term of data format, length of reads, etc., which results in the need for diversity in bioinformatics including sequence quality scoring, alignment, assembly, and data processing.

The benefits of NGS sequencing will not be fully appreciated until extremely high- performance computing and intensive bioinformatics support is available. The information accrued by NGS may lead to a paradigm shift in the way that genetics and bioinformatics converge. Since NGS technology is in an early stage of development, a variety of software tools are under development and many are available online for NGS data analysis. Their functions fit into several general categories: (1) alignment of reads to a reference sequence; (2) de novo assembly, (3) reference-based assembly; (4) base-calling and/or genetic variation detection (such as SNV, Indel); (5) genome annotation, and (6) utilities for data analysis.

Alignment and assembly

Despite the sequencing power of NGS, the short-read length strategy creates serious limitations in many biological applications. Efforts to date have focused on overcoming the limitation of short reads for genome-wide analysis, but unfortunately, current available bioinformatics ability and computing power is lagging far behind the needs for NGS sequencing data analysis. In genomics, reference-based assembly is often performed to map the number of short reads to a human reference genome which creates challenges for the algorithms and computing of alignment. Since repetitive sequences are widely distributed across the entire human genome, some short reads will align equally to multiple chromosomal locations. This is one of the reasons multiple-fold coverage of a given region is required for NGS and why further resequencing with Sanger methodology is often needed to ascertain the genetic variant detected in short reads.

The most important step in NGS data analysis is successful alignment or assembly of short reads to a reference genome. It is a challenge to efficiently align short reads to a reference genome, especially when developing new algorithms to handle ambiguities or lack of accuracy during the alignment. Based on the mapping quality concept, MAQ (Mapping and Assembly with Quality), a very popular NGS software program, was developed that can efficiently map short reads to a

reference genome and derive genotype calls to the consensus sequence with quality scores. MAQ is one of the first reference guided assembly programs. It is accurate, efficient, versatile, and user-friendly, and has been successfully applied to several NGS projects. ELAND (Efficient Large-Scale Alignment of Nucleotide Databases), another NGS program designed to search DNA files for short DNA reads allowing up to 2 errors per match, has also been successfully used in several NGS projects. Benchmarks comparing ELAND with other popular NGS software, such as MAQ, BLAST (Basic Local Alignment Search Tool), SOAP (Short Oligonucleotide Alignment Program), and SeqMap etc. (Table 4), generally place ELAND as one of the fastest available programs. Compared to reference-based assembly with very short-read length sequences, de novo assembly is even more challenging. Currently de novo assembly with NGS data is generally limited to microbial genome projects (Metagenomics) due to the small bacterial genome size. The primary goal of current algorithms and computing for short-read assembly with NGS technologies is to increase read length. This goal will likely be achieved by the development of single-molecule sequencing technologies. Certain improvements in existing NGS technologies, such as mate-paired short reads, may also make this goal attainable. Individual human genomes (one Asian and one African) have been successfully sequenced and assembled using the Illumina Genome Analyzer (read lengths ranged from 35 to 75 basepairs) with a modified SOAP program, SOAP denovo. Available bioinformatic tools for short-read alignment, de novo and reference-based assembly for NGS are listed in Table 4. Since many of the programs are open source, additional programming may be needed to modify the program to the needs of a specific NGS project. Some online utility programs, such as EagleView or LookSeq also provide some additional assistance on NGS data analysis and interpretation (Table 4). In bioinformatics, sequence assembly refers to aligning and merging fragments of a much longer DNA sequence in order to reconstruct the original order of the sequence. Genome assembly is simply putting sequenced fragments of DNA into their correct chromosomal positions. This is needed as DNA sequencing technology cannot read whole genomes in one go, but rather reads small pieces of between 20 and 1000 bases, depending on the technology used. Typically the short fragments, called reads, result from sequencing genomic DNA, or gene transcript (ESTs).

The first sequence assemblers began to appear in the late 1980s and early 1990s as variants of simpler sequence alignment programs to piece together vast quantities of fragments generated by automated sequencing instruments called DNA sequencers. As the sequenced organisms grew in size and complexity (from small viruses over plasmids to bacteria and finally eukaryotes), the assembly programs used in these genome projects needed to increasingly employ more and more sophisticated strategies to handle: Terabytes of sequencing data which need processing on computing clusters; Identical and nearly identical sequences (known as *repeats*) which can, in the worst case, increase the time and space complexity of algorithms exponentially; Errors in the fragments from the sequencing instruments, which can confound assembly.

Table 4. Alignment, Assembly and Utility Bioinformatics tools

Program	Function	Platform	Website
De novo assembly			
Abyss	Alignment/assembly	Illumina	http://www.bcgsc.ca/platform/bioinfo/software/abyss
ALLPATHS	Alignment/assembly	Illumina	http://www.broadinstitute.org/science/programs/genome-biology/crd
AMOScmp	Alignment/assembly	Roche	http://sourceforge.net/projects/amos/files/
ARACHNE	Alignment/assembly	Roche	http://www.broadinstitute.org/science/programs/genome-biology/crd
CAP3	Alignment/assembly	Roche	http://pbil.univ-lyon1.fr/cap3.php
consensus/Seq-Cons	Alignment/assembly	Roche	http://www.seqan.de/downloads/projects.html
Curtain	Alignment/assembly	Illumina/Roche/ABI	http://code.google.com/p/curtain/
Edena	Alignment/assembly	Illumina	http://www.genomic.ch/edena
Euler-SR	Alignment/assembly	Illumina/Roche	http://euler-assembler.ucsd.edu/portal/?q=team
FuzzyPath	Alignment/assembly	Illumina/Roche	ftp://ftp.sanger.ac.uk/pub/zn1/fuzzypath/fuzzypath_v3.0.tgz
IDBA	Alignment/assembly	Illumina	http://www.cs.hku.hk/~alse/idba/
MIRA/MIRA3	Alignment/assembly	Illumina/Roche	http://chevreux.org/projects_mira.html
Newbler	Alignment/assembly	Roche	roche-applied-science.com/
Phrap	Alignment/assembly	Illumina/Roche	http://www.phrap.org/consed/consed.html#howToGet
RGA	Alignment/assembly	Illumina	http://rga.cgrb.oregonstate.edu/
QSRA	Alignment/assembly	Illumina	http://qsra.cgrb.oregonstate.edu/
SHARCGS	Alignment/assembly	Illumina	http://sharcgs.molgen.mpg.de/
SHORTY	Alignment/assembly	ABI	http://www.cs.sunysb.edu/~skiena/shorty/
SHRAP	Alignment/assembly	Roche	By request
SOAPdenovo	Alignment/assembly	Illumina	http://soap.genomics.org.cn
SOPRA	Alignment/assembly	Illumina/ABI	http://www.physics.rutgers.edu/%7Eanirvans/SOPRA/
SR-ASM	Alignment/assembly	Roche	http://bioserver.cs.put.poznan.pl/sr-asm-short-reads-assembly-algorithm
SSAKE	Alignment/assembly	Illumina/Roche	http://www.bcgsc.ca/platform/bioinfo/software/ssake
Taipan	Alignment/assembly	Illumina	http://sourceforge.net/projects/taipan/files/
VCAKE	Alignment/assembly	Illumina/Roche	http://sourceforge.net/projects/vcake
Velvet	Alignment/assembly	Illumina/Roche/ABI	http://www.ebi.ac.uk/%7Ezerbino/velvet
Reference-based assembly			
BFAST	Alignment/assembly	Illumina/ABI	http://sourceforge.net/apps/mediawiki/bfast/index.php?title=Main_Page
Bowtie	Alignment/assembly	Illumina/Roche/ABI	http://bowtie-bio.sourceforge.net
BWA	Alignment/assembly	Illumina/ABI	http://bio-bwa.sourceforge.net/bwa.shtml
CoronaLite	Alignment/assembly	ABI	http://solidsoftwaretools.com/gf/project/corona/
CABOG	Alignment/assembly	Roche/ABI	http://wgs-assembler.sf.net
ELAND/ELAND2	Alignment/assembly	Illumina/ABI	http://www.illumina.com/
EULER	Alignment/assembly	Illumina	http://euler-assembler.ucsd.edu/portal/
Exonerate	Alignment/assembly	Roche	http://www.ebi.ac.uk/~guy/exonerate
EMBF	Alignment/assembly	Illumina	http://www.biomedcentral.com/1471-2105/10?issue=S1
GenomeMapper	Alignment/assembly	Illumina	http://1001genomes.org/downloads/genomemapper.html
GMAP	Alignment/assembly	Illumina	http://www.gene.com/share/gmap

Program	Function	Platform	Website
gnumap	Alignment/assembly	Illumina	http://dna.cs.byu.edu/gnumap/
ICON	Alignment/assembly	Illumina	http://icorn.sourceforge.net/
Karma	Alignment/assembly	Illumina/ABI	http://www.sph.umich.edu/csg/pha/karma/
LAST	Alignment/assembly	Illumina	http://last.cbrc.jp/
LOCAS	Alignment/assembly	Illumina	http://www-ab.informatik.uni-tuebingen.de/software/locas
Mapreads	Alignment/assembly	ABI	http://solidsoftwaretools.com/gf/project/mapreads/
MAQ	Alignment/assembly	Illumina/ABI	http://maq.sourceforge.net
MOM	Alignment/assembly	Illumina	http://mom.csbc.vcu.edu/
Mosaik	Alignment/assembly	Illumina/Roche/ABI	http://bioinformatics.bc.edu/marthlab/Mosaik
mrFAST/mrsFAST	Alignment/assembly	Illumina	http://mrfast.sourceforge.net/
MUMer	Alignment/assembly	ABI	http://mummer.sourceforge.net/
nexalign	Alignment/assembly	Illumina	http://genome.gsc.riken.jp/osc/english/dataresource/
Novocraft	Alignment/assembly	Illumina	http://www.novocraft.com/
PerM	Alignment/assembly	Illumina/ABI	http://code.google.com/p/perm/
RazerS	Alignment/assembly	Illumina/ABI	http://www.seqan.de/projects/razers.html
RMAP	Alignment/assembly	Illumina	http://rulai.cshl.edu/rmap
segemehl	Alignment/assembly	Illumina/Roche	http://www.bioinf.uni-leipzig.de/Software/segemehl/
SeqCons	Alignment/assembly	Roche	http://www.seqan.de/projects/seqcons.html
SeqMap	Alignment/assembly	Illumina	http://biogibbs.stanford.edu/*jiangh/SeqMap/
SHRiMP	Alignment/assembly	Illumina/Roche/ABI	http://compbio.cs.toronto.edu/shrimp
Slider/SliderII	Alignment/assembly	Illumina	http://www.bcgsc.ca/platform/bioinfo/software/slider
SOCS	Alignment/assembly	ABI	http://solidsoftwaretools.com/gf/project/socs/
SOAP/SOAP2	Alignment/assembly	Illumina/ABI	http://soap.genomics.org.cn
SSAHA/SSAHA2	Alignment/assembly	Illumina/Roche	http://www.sanger.ac.uk/Software/analysis/SSAHA2
Stampy	Alignment/assembly	Illumina	http://www.well.ox.ac.uk/~marting/
SXOligoSearch	Alignment/assembly	Illumina	http://synasite.mgrc.com.my:8080/sxog/NewSXOligoSearch.php
SHORE	Alignment/assembly	Illumina	http://1001genomes.org/downloads/shore.html
Vmatch	Alignment/assembly	Illumina	http://www.vmatch.de/
Diagnostics/utilities			
Artemis/ACT	Visualization tool	Illumina/Roche	http://www.sanger.ac.uk/resources/software/artemis/
CASHX	Pipeline	Illumina	http://seqanswers.com/wiki/CASHX
Consed	Visualization tool	Illumina/Roche	http://www.genome.washington.edu/consed/consed.html
EagleView	Visualization tool	Illumina/Roche	http://bioinformatics.bc.edu/marthlab/EagleView
FastQC	Quality assessment	Illumina/ABI	http://www.bioinformatics.bbsrc.ac.uk/projects/fastqc/
Gambit	Visualization tool	Illumina/Roche	http://bioinformatics.bc.edu/marthlab/Gambit
Goby	Data management	Illumina/Roche/ABI	http://campagnelab.org/software/goby/
G-SQZ	Data management	Illumina/ABI	http://public.tgen.org/sqz
Hawkeye	Visualization tool	Illumina/Roche	http://amos.sourceforge.net/hawkeye
Hybrid-SHREC	Error Correction	Illumina/Roche/ABI	http://www.cs.helsinki.fi/u/lmsalmel/hybrid-shrec/
IGV	Visualization tool	Illumina	http://www.broadinstitute.org/igv/?q=home
LookSeq	Visualization tool	Illumina/Roche	http://lookseq.sourceforge.net
MagicViewer	Visualization tool	Illumina	http://bioinformatics.zj.cn/magicviewer/

Program	Function	Platform	Website
MapView	Visualization tool	Illumina	http://evolution.sysu.edu.cn/mapview/
NGSView	Visualization tool	Illumina/ABI	http://ngsview.sourceforge.net
PIQA	Quality assessment	Illumina	http://bioinfo.uh.edu/PIQA
Reconciliation	Assembly pipeline	Illumina	http://www.genome.umd.edu/software.htm
RefCov	Sequence coverage	Illumina/Roche	http://genome.wustl.edu/tools/cancer-genomics
SAM Tools	Utilities	Illumina/Roche	http://sourceforge.net/projects/samtools/files/
Savant	Visualization tool	Illumina/Roche	http://compbio.cs.toronto.edu/savant/
ShortRead	Quality assessment	Illumina/Roche	http://bioconductor.org/packages/2.6/bioc/html/ShortRead.html
SHREC	Error Correction	Illumina/Roche	http://www.informatik.uni-kiel.de/jasc/Shrec/
Staden Tools (GAP5)	Pipeline	Illumina/Roche	http://sourceforge.net/projects/staden/files/
Tablet	Visualization tool	Illumina/Roche	http://bioinf.scri.ac.uk/tablet
TagDust	Data cleaning	Illumina	http://genome.gsc.riken.jp/osc/english/software/.
TileQC	Quality assessment	Illumina	http://www.science.oregonstate.edu/~dolanp/tileqc
XMatchView	Visualization tool	Illumina/Roche	http://www.bcgsc.ca/platform/bioinfo/software/xmatchview
Yenta	Visualization tool	Illumina	http://genome.wustl.edu/tools/cancer-genomics
Geneus	Data management	Illumina/ABI	http://www.genologics.com/solutions/research-informatics/

Annotation and functional prediction

After successful alignment and assembly of NGS data, the next challenge is to interpret the large number of apparently novel genetic variants (or mutations) present by chance in any single human genome, making it difficult to identify which variants are causal, even when considering only non-synonymous variants. Many novel genetic variants/variations have been discovered for each sequenced genome, resulting in approximately 400 function- altering variants for protein-coding sequences per individual genome. Recognition of functional variants is at the center of the NGS data analysis and bioinformatics. It is challenging to develop software with the ability to distinguish low-frequency alleles descendent from ancient ancestors from de novo or extremely rare mutations recently introduced into the population. Available bioinformatics tools for annotation and functional prediction of NGS data are listed in Table 5. SIFT (Sorting Intolerant From Tolerant) is used to predict whether an amino acid substitution affects protein function based on sequence homology and the physical properties of the amino acid can be applied to find non-synonymous polymorphisms within NGS data. By considering the physiochemical variations presented in protein sequence alignment and the property of variations, Multivariate Analysis of Protein Polymorphism (MAPP) can predict the impact of all possible amino acid substitutions on the function of the protein. Predictor of human Deleterious Single Nucleotide Polymorphisms (PhD-SNP) is an optimized program to predict if a given single point protein mutation can be classified as disease-related or as neutral polymorphism based on protein sequence and profile information. Polymorphism Phenotyping (PolyPhen) and updated PolyPhen-2 are tools which predict the possible impact of an amino acid substitution on the structure and function of a specific protein using straightforward comparative physical

methods. Variation detection software, which includes screening genomes for structural and single nucleotide variants and the differences between genomes are generally integrated with alignment and assembly processes and are listed in Table 5.

Table 5. Genetic Variant Prediction and detection bioinformatics programs for NGS data analysis

Variant prediction/detection	Platform	Website
Functional variant prediction		
B-SIFT		http://research-pub.gene.com/bsift/
MAPP		http://mendel.stanford.edu/supplementarydata/stone_MAPP_2005
PhD-SNP		http://gpcr.biocomp.unibo.it/~emidio/PhD-SNP/PhD-SNP
PolyPhen-2/PolyPhen		http://genetics.bwh.harvard.edu/pph2/
SIFT		http://blocks.fhcrc.org/sift/SIFT.html
SNAP		http://www.rostlab.org/services/SNAP
SNAPper/Pedant		http://pedant.gsf.de/snapper
Variant detection		
Structural/genomic variant		
BreakDancer	Roche/Illumina/ABI	http://genome.wustl.edu/tools/cancer-genomics/
BreakDancer/BD- Mini	Roche/Illumina/ABI	http://seqanswers.com/wiki/BreakDancer
Breakway	Roche/Illumina/ABI	http://sourceforge.net/projects/breakway/files/
CNVSeq	Roche	http://tiger.dbs.nus.edu.sg/CNV-seq/
cnvHMM	Illumina	http://genome.wustl.edu/pub/software/cancer-genomics/cnvHMM/
cnD	Illumina	http://www.sanger.ac.uk/resources/software/cnd.html
GASV/GSV	Illumina	http://cs.brown.edu/people/braphael/software.html
Hydra	Illumina	http://code.google.com/p/hydra-sv/
MoDIL	Illumina	http://compbio.cs.toronto.edu/modil/
mrFAST	Illumina	http://mrfast.sourceforge.net/
NovelSeq	Roche/Illumina/ABI	http://compbio.cs.sfu.ca/strvar.htm
PEMer	Roche/Illumina/ABI	http://sv.gersteinlab.org/pemer/
Pindel	Illumina	http://www.ebi.ac.uk/~kye/pindel/
SegSeq	Illumina/ABI	http://www.broadinstitute.org/
SOAPsv	Roche/Illumina/ABI	http://soap.genomics.org.cn
Solid large Indel tool	ABI	http://solidsoftwaretools.com/gf/project/large_indel/
Solid CNV tool	ABI	http://solidsoftwaretools.com/gf/project/cnv/
SWT	Illumina	http://genome.wustl.edu/pub/software/cancer-genomics/GSTAT/
VariationHunter/VH-CR	Illumina	http://compbio.cs.sfu.ca/strvar.html
VARiD	ABI	http://compbio.cs.utoronto.ca/varid

Variant prediction/detection	Platform	Website
Single nucleotide variant		
Atlas-SNP2	Roche/Illumina	http://www.hgsc.bcm.tmc.edu/cascade-tech-software-ti.hgsc
BOAT	Illumina	http://boat.cbi.pku.edu.cn/
DNA Baser	Roche	http://www.dnabaser.com/help/manual.html
DNAA	Roche/Illumina/ABI	http://sourceforge.net/projects/dnaa/
Galign	Illumina	http://shahamlab.rockefeller.edu/galign/galign.htm
GigaBayes/PbShort	Roche/Illumina	http://bioinformatics.bc.edu/marthlab/GigaBayes
GSNAP	Roche/Illumina	http://share.gene.com/gmap.
inGAP	Roche/Illumina	http://sites.google.com/site/nextgengenomics/ingap
ngs_backbone	Roche/Illumina	http://bioinf.comav.upv.es/ngs_backbone/index.html
Omixon Variant	ABI	http://www.omixon.com/omixon/index.html
PyroBayes	Roche	http://bioinformatics.bc.edu/marthlab/PyroBayes
ssahaSNP	Illumina/Roche	http://www.sanger.ac.uk/Software/analysis/ssahaSNP
Slider	Illumina	http://www.bcgsc.ca/platform/bioinfo/software/slider
SNP-o-matic	Illumina	http://snpomatic.sourceforge.net
SNPSeeker	Illumina	http://www.genetics.wustl.edu/rmlab/
SNVMix	Illumina	http://compbio.bccrc.ca
SOAPsnp	Roche/Illumina/ABI	http://soap.genomics.org.cn
SWA454	Roche	http://www.broadinstitute.org/science/programs/genome-biology/crd
SVA	Illumina	http://www.svaproject.org/
VAAL	Illumina	http://www.broadinstitute.org/science/programs/genome-biology/crd
VarScan	Roche/Illumina	http://genome.wustl.edu/tools/cancer-genomics
VARiD	Roche/Illumina/ABI	http://compbio.cs.utoronto.ca/varid
Differences between genomes		
DIAL	Illumina	http://www.bx.psu.edu/miller_lab/
SomaticCall	Illumina	http://www.broadinstitute.org/science/programs/genome-biology/crd
SWAP454	Roche	http://www.broadinstitute.org/science/programs/genome-biology/crd
VAAL	Illumina	http://www.broadinstitute.org/science/programs/genome-biology/crd

End-user packages

End-user software packages which provide a user-friendly interface, easy to use data input and output formats, and integrates multiple computing programs into one software package, may be the best solution for most biomedical researchers. Based on our experience, among available end-user packages, Genomic Workbench from CLC Bio appears to be the most widely used. NextGENe from SoftGenetics is excellent for candidate-gene resequencing projects, but it cannot handle very large datasets and may not be suitable for large genome sequencing projects. SeqMan Ngen from DNASTAR is under development but currently unavailable. Although commercial end-user packages tend to carry a hefty price, some are available free online as detailed in Table 6.

Table 6. End-User software packages and cloud computing software for NGS data analysis

Software packages	Function	Website
End-user software packages		
Genomic workbench/CLCbio	Multi-task	http://www.clcbio.com/index.php?id=1331
NextGENe/SoftGenetics	Multi-task	http://softgenetics.com/NextGENe.html
Genomatix Genome Analyzer	Multi-task	http://www.genomatix.de/genome_analyzer.html
Zoom	Multi-task	http://www.bioinformaticssolutions.com/products/zoom/index.php
SeqMan Ngen/DNASTAR	Multi-task	http://www.dnastar.com/t-products-seqman-ngen.aspx
JMP Genomics	Multi-task	http://www.jmp.com/software/genomics/index.shtml
RTG/Real Time Genomics	Multi-task	http://www.realtimegenomics.com/RTG-Software
PASS	Multi-task	http://pass.cribi.unipd.it/cgi-bin/pass.pl?action=Download
CASAVA	Multi-task	http://www.illumina.com/software/
Geneus/GenoLogics	Multi-task	http://www.genologics.com/solutions/research-informatics/
Roche Analysis tools	Multi-task	http://454.com/products-solutions/analysis-tools/index.asp
VSRAP	Multi-task	http://sourceforge.net/apps/mediawiki/vancouvershortr/
BING	Multi-task	http://www.dinulab.org/bing
PaCGeE/PGI	Multi-task	http://personalgenomicsinstitute.org/index.php/
GATK	Multi-task	http://www.broadinstitute.org/gsa/wiki/index.php/
Geneious Pro	Multi-task	http://www.geneious.com/default,1246,NGS%20Assembly.sm
Partek GS/Partek	Multi-task	http://www.partek.com/partekgs
Bioscope	Multi-task	https://products.appliedbiosystems.com/ab/en/US/adirect/
Cloud computing		
Crossbow	Mapping and SNP calling	http://bowtie-bio.sf.net
CloudBurst	Reference-based mapping	http://sourceforge.net/apps/mediawiki/cloudburst-bio/
Contrail	De novo assembly	http://sourceforge.net/apps/mediawiki/contrail-bio/
Cloud-MAQ	Modified-Maq for cloud	http://geschickten.com/download.html
Bioscope	Reference-based mapping	https://products.appliedbiosystems.com/ab/en/US/adirect/
Cycle Computing	Cloud computing	http://www.cyclecomputing.com/
GenomeQuest	Customer service	http://www.genomequest.com/
Geospiza/GeneSifter	Customer service	http://www.geospiza.com/Contact/genesiftertrial_ng.shtml

As previously mentioned, the fact that high-performance computing and intensive bioinformatic support is needed for NGS, it is difficult for many research laboratories to successfully conduct NGS projects due to the high level of information technology support required. A possible solution is cloud computing. In cloud computing, a user can use a virtual operating system (or "cloud") to process data on a computer cluster for high parallel tasks. CrossBow is the first cloud computing software capable of performing alignment and single nucleotide polymorphism analysis on multiple whole- human datasets. CloudBurst is another new parallel read-mapping cloud algorithm optimized for mapping NGS data to a human reference genome, SNP discovery, genotyping and personal genomics. Data generated on Applied Bio-systems' SoLiD platform uses a two colored system which makes it unsuitable for analysis by many available software packages. The Bioscope package, developed by ABI, is devoted to their SoLiD data and can be used as a single software package or for cloud computing, likewise, CASAVA package developed by Illumina is utilized for Genome Analyzer data. Available NGS cloud computing technologies are listed in Table 6. GenomeQuest, Complete Genomics and Geospiza/GeneSifter

provide online customer oriented NGS data analysis services, which is a little different from cloud computing by definition.

Conclusion

NGS technologies will certainly enable us to identify all the causative variants including "rare variants" within individual human subjects. It is anticipated that whole- genome sequencing (or exome sequencing) will make significant contributions to our understanding of the genetic etiologies that contribute to complex human disease, as well as the genetic basis of genomics.

Further reading

1. *Ansorge WJ. Next-generation DNA sequencing techniques. Nat. Biotechnol. 2009; 25:195–203.*
2. *Bau S, Schracke N, Kranzle M, Wu H, Stahler PF, Hoheisel JD, Beier M, Summerer D. Targeted next- generation sequencing by specific capture of multiple genomic loci using low-volume microfluidic DNA arrays. Anal. Bioanal. Chem. 2009; 393:171–175. [PubMed: 18958448]*
3. *Chaisson MJ, Brinza D, Pevzner PA. De novo fragment assembly with short mate-paired reads: does the read length matter? Genome Res. 2009; 19:336–346. [PubMed: 19056694]*
4. *Chistoserdova L. Recent progress and new challenges in metagenomics for biotechnology. Biotechnol. Lett. 2010; 32:1351–1359. [PubMed: 20495950]*
5. *Daly AK. Genome-wide association studies in pharmacogenomics. Nat. Rev. Genet. 2010a; 11:241– 246. [PubMed: 20300088]*
6. *Daly AK. Pharmacogenetics and human genetic polymorphisms. Biochem. J. 2010b; 429:435–449. [PubMed: 20626352]*
7. *Flicek P, Birney E. Sense from sequence reads: methods for alignment and assembly. Nat. Methods. 2009; 6:S6–S12. [PubMed: 19844229]*
8. *Frazer KA, Murray SS, Schork NJ, Topol EJ. Human genetic variation and its contribution to complex traits. Nat. Rev. Genet. 2009; 10:241–251. [PubMed: 19293820]*
9. *Li H, Ruan J, Durbin R. Mapping short DNA sequencing reads and calling variants using mapping quality scores. Genome Res. 2008a; 18:1851–1858. [PubMed: 18714091]*
10. *Li R, Li Y, Kristiansen K, Wang J. SOAP: short oligonucleotide alignment program. Bioinformatics. 2008b; 24:713–714. [PubMed: 18227114]*
11. *Mardis ER. Anticipating the 1,000 dollar genome. Genome Biol. 2006; 7:112. [PubMed: 17224040]*
12. *Mardis ER. The impact of next-generation sequencing technology on genetics. Trends Genet. 2008a; 24:133–141. [PubMed: 18262675]*

Chapter 2

Basic Linux Commands

Navish Kumar Batchu & Jithin Subash

Command	Description	Usage/Example	Comments
mkdir	Create a Directory	mkdir Linux_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 cd ..	Changes current directory location to Linux_commands 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	rm text1	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 mv text1 ./ Linux_commands	It will rename text1 to text2 i,e the name is changed. 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 grep –c 'hello' text1	Searches for 'hello' in 'text1' Displays number of times 'hello' is present in text1
wc	Prints word count of newlines, words, etc	wc –l text1	Prints number of lines in 'text1'
tar	To compress and uncompress folders	tar –cvf new text1 text2 tar –xvf new	Compresses text1 and text2 and create new.tar file Uncompress folder new
gunzip	To uncompress zipped file	gunzip filename.gz	Uncompress the folder

Chapter 3

NGS Data Formats

Rosana O. Babu

Introduction- NGS Data Formats

Sequence Sources & Read Data Formats

NGS data sets are very large, but as the cost per MB is now so low we expect to see them generated for many large-scale experiments where the mapping and analysis of the short read sets will be the key to interpretation of results.

For further analysis of NGS assembled data, whether a de novo assembly or using a set of NGS reads mapped to a reference sequence. Our initial aim is to rapidly produce a set of utility functions that can be used to read and interpret NGS assemblies, for example MAQ .map files, and to calculate statistics based on position, gene an

notations, etc. These will then be used to build new applications and to demonstrate to end users, and to other developers, the ease of programming and the added value of the many interfaces already available.

NGS data volumes require new storage structures. As well as reading MAQ .map files and other formats for 454 sequencing and for new instruments) either by reading their formats or by using existing utilities to convert them. The instrument manufacturers show encouraging signs of using open data format and we will keep a close watch on emerging formats for a brief understanding

(i) NGS Data Sources

- NCBI
- EMBL
- DDBJ
- MINSEQE Minimum Information about a high-throughput Sequencing Experiment

(ii) Sequences

Sequences can be read and written in a variety of formats. These can be very confusing for users, but EMBOSS aims to make life easier by automatically recognizing the sequence format on input. That means that if you are converting from using another sequencing package to EMBOSS and you have your existing sequences in a format that is specific for that package, for example GCG format, you will have no problem reading them in. If you don't hold your sequence in a

recognized standard format, you will not be able to analyze your sequence easily. Sequencers; read aligners; genome assemblers; scientific results from sequencing data; databases of reads; databases of genomes, variation, or other results. When all of these output data and require inputs in their own formats, we can be sure that there will be enormous amount of annoying, boring, and disappointing work while dealing with sequencing data.

(iii) What is sequence format?

Sequence formats are ASCII TEXT. They are the required arrangement of characters, symbols and keywords that specify what things such as the sequence, ID name, comments, etc. look like in the sequence entry and where in the entry the program should look to find them. There are generally no hidden, unprintable 'control' characters in any sequence format (there are none in those that EMBOSS supports). All standard sequence formats can be printed out or viewed simply by displaying their file.

(iv) Why so many formats?

There are at least a couple of dozen sequence formats in existence at the moment. Some are much more common than others. Formats were designed so as to be able to hold the sequence data and other information about the sequence. Nearly every sequence analysis package written since programs were first used to read and write sequences has invented its own format. Nearly every collection of sequences that dares call itself a database has stored its data in its own format. Most formats allow you to hold other description, annotation and comments like Fasta format. Other formats have specific fields for holding information such as references, keywords, associated entries in other databases and feature tables.

(v) Identification

Most sequence formats include at least one form of ID name, usually placed somewhere at the top of the sequence format. The simple format **FASTA** has the ID name as the first word on its title line. For example the ID name 'xyz':

>**xyz**
ttcctctttctcgactccatcttcgcggtagctgggaccgccgttcagtcgccaatatgcagctctttgtccgcgcccaggagctaca
caccttcgaggtgaccggccaggaaacggtcgcccagatcaaggctcatgtagcctcactggagggcatt

(vi) Sequence ID's and Accessions

An entry in a database must have some way of being uniquely identified in that database. Most sequence databases have two such identifiers for each sequence - an ID name and an Accession number.

Why are there two such identifiers? The ID name was originally intended to be a human-readable name that had some indication of the function of its sequence. In EMBL and GenBank the first two (or three) letters indicated the species and the rest indicated the function, for example 'hsfau' is the 'Homo Sapiens FAU pseudogene'. This naming scheme started to be a problem when the number of entries added each day was so vast that people could not make up the ID names fast enough. Instead, the Accession numbers were used as the ID name. Therefore you will now find ID names like 'AF061303', the same as the Accession number for that sequence in EMBL. ID names are not guaranteed to remain the same between different versions of a database (although in practice they usually do).

Accession numbers are unique alphanumeric identifiers that are guaranteed to remain with that sequence through the rest of the life of the database. If two sequences are merged into one, then the new sequence will get a new Accession number and the Accession numbers of the merged sequences will be retained as 'secondary' Accession numbers. EMBL, GenBank and SwissProt share an Accession numbering scheme - an Accession number uniquely identifies a sequence within these three databases.

(vii) The Sequence

Nucleotide (DNA or RNA) sequences are usually stored in the IUBMB standard codes (International Union of Biochemistry and Molecular Biology*).*

Similarly, protein sequences are usually stored in the IUPAC standard one-letter codes (*International Union of Pure and Applied Chemistry*).

```
A = adenine
C = cytosine
G = guanine
T = thymine
U = uracil
R = G A (purine)
Y = T C (pyrimidine)
K = G T (keto)
M = A C (amino)
S = G C
W = A T
B = G T C
D = G A T
H = A C T
V = G C A
N = A G C T (any)
```

Sequence Database Formats

Some of the most widespread sequence formats apart from **fasta** are those used by the major sequence databases.

- EMBL
- GenBank
- SwissProt
- PIR

(i) Sequence Files

Files can hold sequences in standard recognized formats. Files can also hold sequences in non-standard unrecognizable ways. Do not save your sequences in a word-processor format file.

NGS Data Formats

To date, the following sequence formats are accepted as input for NGS tools.

(i) Read Formats (sample are given below)

- **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

Sample Sequence and Read formats

Sequence formats

(i) Plain sequence format
A sequence in plain format may contain only IUPAC characters and spaces (no numbers!).

Note: A file in plain sequence format may only contain one sequence, while most other formats accept several sequences in one file.

An example sequence in plain format is
acaagatgccattgtcccccggcctcctgctgctgctctccggggccacggccaccgctgccctgcccctggagggtggccc
cggaataaggaaaagca

(ii) EMBL format
A sequence file in EMBL format can contain several sequences. One sequence entry starts with an identifier line ("ID"), followed by further annotation lines. The start of the sequence is marked by a line starting with "SQ" and the end of the sequence is marked by two slashes ("//").

An example sequence in EMBL format is:
ID AB000263 standard; RNA; PRI; 368 BP.
XX
AC AB000263;
XX
DE Homo sapiens mRNA for preprocortistatin like peptide, complete cds.
XX
SQ Sequence 368 BP;
acaagatgccattgtcccccggcctcctgctgctgctctccggggccacggccaccg 60
ctgccctgcccctggagggtggccccaccggccgagacagcgagcatatgcaggaagcgg 120
caggaataaggaaaagcagcctcctgactttcctcgcttggtggtttgagtggacctccc 180
aggccagtgccgggcccctcataggagaggaagctcgggaggtggccaggcggcaggaag 240
gcgcacccccccagcaatccgcgcgccgggacagaatgccctgcaggaacttcttctgga 300
agaccttctcctcctgcaaataaaacctcacccatgaatgctcacgcaagtttaattaca 360
gacctgaa 368
//

(iii) FASTA format

A sequence file in FASTA format can contain several sequences. Each sequence in FASTA format begins with a single-line description, followed by lines of sequence data. The description line must begin with a greater-than (">") symbol in the first column.

An example sequence in FASTA format is:

>AB000263 |acc=AB000263|descr=Homo sapiens mRNA for preprocortistatin like peptide, complete cds.|len=368
ACAAGATGCCATTGTCCCCCGGCCTCCTGCTGCTGCTGCTCTCCGGGGCCACGG
CCACCGCTGCCCTGCCCCTGGAGGGTGGC

(iv) GCG format

A sequence file in GCG format contains exactly one sequence, begins with annotation lines and the start of the sequence is marked by a line ending with two dot ("..") characters. This line also contains the sequence identifier, the sequence length and a checksum. This format should only be used if the file was created with the GCG package.

An example sequence in GCG format is:

ID AB000263 standard; RNA; PRI; 368 BP.
XX
AC AB000263;
XX
DE Homo sapiens mRNA for preprocortistatin like peptide, complete cds.
XX
SQ Sequence 368 BP;
AB000263 Length: 368 Check: 4514 ..
1 acaagatgccattgtcccccggcctcctgctgctgctgctctccggggccacggccaccg
61ctgccctgcccctggagggtggccccaccggccgagacagcgagcatatgcaggaagcgg

(v) GCG-RSF (rich sequence format)

The new GCG-RSF can contain several sequences in one file. This format should only be used if the file was created with the GCG package.

(vi) GenBank format

A sequence file in GenBank format can contain several sequences. One sequence in GenBank format starts with a line containing the word LOCUS and a number of annotation lines. The start of the sequence is marked by a line containing "ORIGIN" and the end of the sequence is marked by two slashes ("//").

An example sequence in GenBank format is:

```
LOCUS       AB000263              368 bp    mRNA    linear   PRI 05-FEB-1999
DEFINITION  Homo sapiens mRNA for preprocortistatin like peptide, complete
cds.
ACCESSION   AB000263
ORIGIN
    1  acaagatgccattgtcccccggcctcctgctgctgctctccggggccacggccaccg
   61 tgccctgcccctggagggtggccccaccggccgagacagcgagcatatgcaggaagcgg
  121ggaataaggaaaagcagcctcctgactttcctcgcttggtggtttgagtggacctccc
  181gccagtgccgggcccctcataggagaggaagctcgggaggtggccaggcggcaggaag
  241gcaccccccccagcaatccgcgcgccgggacagaatgccctgcaggaacttcttctgga
  301accttctcctcctgcaaataaaacctcacccatgaatgctcacgcaagtttaattaca
  361cctgaa
//
```

(vii) Genomatix annotation syntax

Some Genomatix tools, e.g. Gene2Promoter or GPD allow the extraction of sequences. Genomatix uses the following syntax to annotate sequence information: each information item is denoted by a keyword, followed by a "=" and the value. These information items are separated by a pipe symbol "|".

The keywords are the following:

loc	The Genomatix Locus Id, consisting of the string "GXL_" followed by a number.
sym	The gene symbol. This can be a (comma-separated) list.
geneid	The NCBI Gene Id. This can be a (comma-separated) list.
acc	A unique identifier for the sequence. E.g. for Genomatix promoter regions, the Genomatix Promoter Id is listed in this field.
taxid	The organism's Taxon Id
spec	The organism name
chr	The chromosome within the organism.
ctg	The NCBI contig within the chromosome.
str	Strand, (+) for sense, (-) for antisense strand.
start	Start position of the sequence (relative to the contig).
end	End position of the sequence (relative to the contig).
len	Length of the sequence in basepairs.
tss	A (comma-separated list of) UTR-start/TSS position(s). If there are several TSS/UTR-starts, this means that several transcripts share the same promoter (e.g. when they are splice variants). The positions are relative to the promoter region.
probe	A (comma-separated list of) Affymetrix Probe Id(s).
unigene	A (comma-separated list of) UniGene Cluster Id(s).
homgroup	An identifier (a number) for the homology group (available for promoter sequences

	only). Orthologously related sequences have the same value in this field.
promset	If the sequence is a promoter region, the promoter set is denoted here.
descr	The gene description. If several genes (i.e. NCBI gene ids) are associated with the sequence, the descriptions for all of the genes are note, separated by ";"
comm	A comment field, used for additional annotation. For promoter sequences, this field contains information about the transcripts associated with the promoter. For each transcript the Genomatix Transcript Id, accession number, TSS position and quality is listed, separated by "/". For GenomatixCompGen promoters no transcripts are assigned, in this case the string "CompGen promoter" is denoted.

(viii) IG format

A sequence file in IG format can contain several sequences, each consisting of a number of comment lines that must begin with a semicolon (";"), a line with the sequence name (it may not contain spaces!) and the sequence itself terminated with the termination character '1' for linear or '2' for circular sequences.

An example sequence in IG format is:
; comment
; comment
AB000263
ACAAGATGCCATTGTCCCCCGGCCTCCTGCTGCTGCTGCTCTCCGGGGCCACGG
CCACCGCTGCCCTGCC

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 (see the black example).

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

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

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     TATTGCAACTTGTTTAATTTTTTCATGCCATTATCA
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 ord ('h') – 64 = 104 – 64 = 40.

```
@ILMN-GA001_3_208HWAAXX_1_1_110_812
ATACAAGCAAGTATAAGTTCGTATGCCGTCTT
+ILMN-GA001_3_208HWAAXX_1_1_110_812
hhhYhh]NYhhhhhhYIhhaZT[hYHNSPKXR
@ILMN-GA001_3_208HWAAXX_1_1_111_879
GGAGGCTGGAGTTGGGGACGTATGCGGCATAG
+ILMN-GA001_3_208HWAAXX_1_1_111_879
hSWhRNJ\hFhLdhVOhAIB@NFKD@PAB?N?
```

Read with .fastq format.

See http://www.asciitable.com/ for ASCII codes (Dec is the 'ord' in decimal numbers).

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 ^ (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
        .GGCAGCGGGCAGGGCGAGCCAATGCGTGTGGGGGGGGGGGGGCTCGCAGTGGGGGGGGAACGGCGAGTGCGGGGGG
        BBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBB      0
HWUSI-EAS521    2       1       26      0       562     0       1
        .GGGAAATAGCTTTCACGCCTTTAGATAATTTCATAAAAATCATAGCGCCAAATGGGGAGCAAACTACCATACACC
        BBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBB      0
HWUSI-EAS521    2       1       26      0       1113    0       1
        .ATCTTTAACAGACCAAGACTGGGCCACAAGCCTCCAGACTGTAACCACTGCTTCAAAGAGGCTTAGGCAGGCAGG
        BBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBB      0
HWUSI-EAS521    2       1       26      0       244     0       1
        .GGCTGGGTATGAGTCAGGGGGCTCCAGAGAGACAGAACCAGTCGGACATCGACAGATAGATGCGGGGGGAGTTAT
        BBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBB      0
```

Read with .Qseq format.

Illumina "**single line format**" and **SCARF**: base-called sequences and quality scores in a bit more handy format.

```
>1-1-137-689     AACATAATGTTCACTGAGAACACATTGCACTCAA   U0
>1-1-87-649      TATTGCAACTTGTTTAATTTTTCATGCCATTATCA  U1
>1-1-121-642     TACATGATTTGCATTTGGTAAATAGCTACTTTTAT  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
T3302323020310210322333002030023300 1
+
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, what is good news.

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

```
@HD    VN:1.0
@SQ    SN:chr20 LN:62435964
@RG    ID:L1 PU:SC_1_10 LB:SC_1 SM:NA12891
@RG    ID:L2 PU:SC_2_12 LB:SC_2 SM:NA12891
read_28833_29006_6945 99  chr20 28833 20 10M1D25M = 28993 195 \
       AGCTTAGCTAGCTACCTATATCTTGGTCTTGGCCG <<<<<<<<<<<<<<<<<<<<:<9/,&,22;;<<< \
       NM:i:1 RG:Z:L1
read_28701_28881_323b 147 chr20 28834 30 35M      = 28701 -168 \
       ACCTATATCTTGGCCTTGGCCGATGCGGCCTTGCA <<<<<;<<<<7;:<<<6;<<<<<<<<<<<7<<<< \
       MF:i:18 RG:Z:L2
```

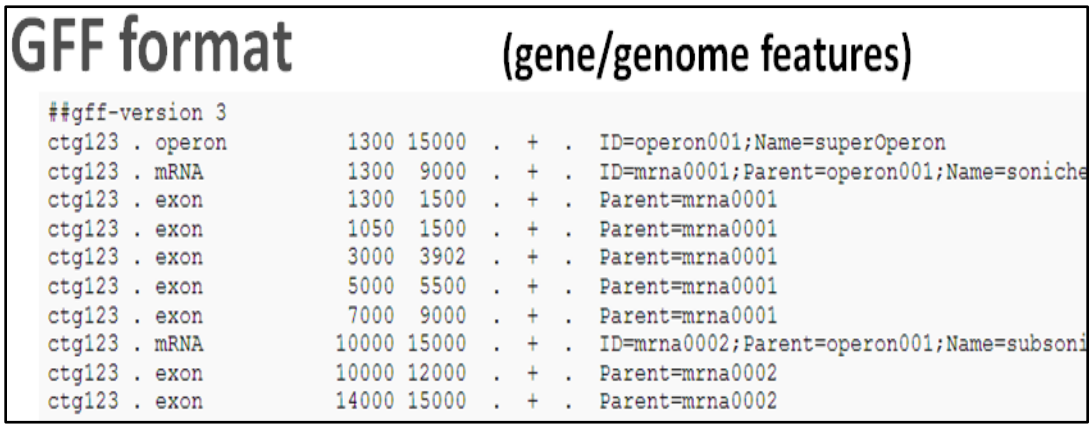
Figure 10: Read with SAM format

Genome & sequence annotation formats

Genome & sequence annotation formats can store some of the diverse kinds of data resulting from a sequencing study.BED format is a plain-text (tabular) format that can directly be displayed for example in the UCSC Genome Browser or similar.

GFF format, latest version GFF3. It's a tabular plain-text format for genome or sequence annotation, can contain also the sequences, alignments, dependencies between features. Is extensible. Currently most recommended & quite "standard" format.

BioXSD is a new set of structured, "object-oriented" formats for exchange of sequence data, any kind of sequence/genome annotation, and related. It is under development and optimised for use with (but not limited to) Web services, and aims to becoming a standard format in future. Very generic and extensible thanks to the use of XML Schema (XSD).



Read with SAM format

Metadata Storage

Nucleotide sequence databases (GenBank, EMBL, 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 Minimimum 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.)

The two main public read archives and their two main formats

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. 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 anycases 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.

Chapter 4

Quality control *(QC)*

Anantharamanan R.

Introduction- Quality control *(QC)*

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. Modern high throughput sequencers can generate tens of millions of sequences in a single run. Before analyzing this sequence to draw biological conclusions you should always perform some simple quality control checks to ensure that the raw data looks good and there are no problems or biases in your data which may affect how you can usefully use it.

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.
- QC and primary analysis of the huge sequencing data using ordinary methods / machines is close to impossible.

Q C analysis using Tool: FastQC

FastQC aims to provide a QC report which can spot problems which originate either in the sequencer or in the starting library material. FasQC 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.

Basic Operations

(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. In this mode the program will merge the files in a sample group and present a single report for each sample. Also Casavafastq 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. If you want to override this detection and specify the file format manually then you can use the drop down file filter in the file chooser to select the type of file you're going to load. You need to use the drop down selector to make the program use the Mapped BAM or Casava file modes as these won't be selected automatically.

(ii) Evaluating Results

The analysis in FastQC is performed by a series of analysis modules. The left hand side of the main interactive display or the top of the HTML report show a summary of the modules which were run, and a quick evaluation of whether the results of the module seem entirely normal (green tick), slightly abnormal (orange triangle) or very unusual (red cross).

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. Some experiments may be expected to produce libraries which are biased in particular ways. You should treat the summary evaluations therefore as pointers to where you should concentrate your attention and understand why your library may not look random and diverse.

(iii) Saving a Report

In addition to providing an interactive report FastQC also has the option to create an HTML version of this report for a more permanent record. This HTML report can also be generated directly by running FastQC in non-interactive mode. 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. When uncompressed this will create a folder with the same name as the report file. Inside this will be a series of files, but the one to look at is the fastqc_report.html file which links to the other files in the report.

Analysis modules

(i) Basic Statistics

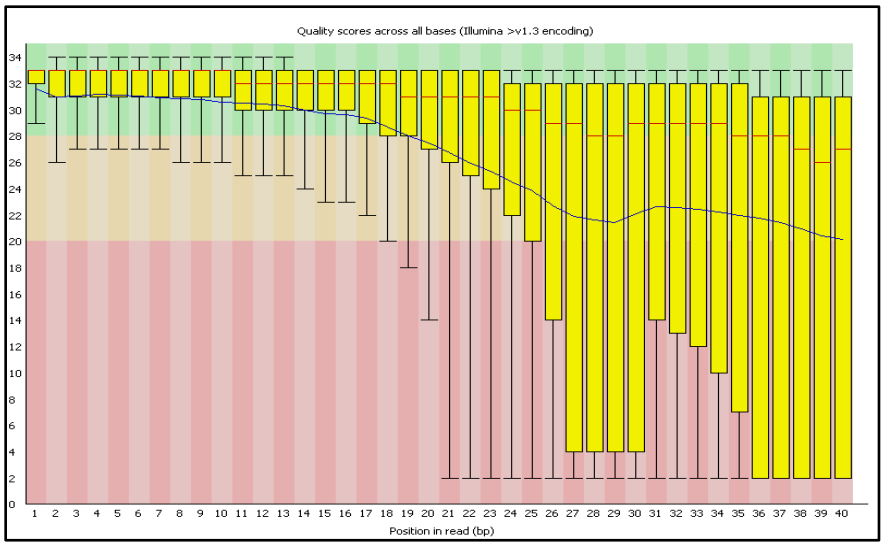
The Basic Statistics module generates some simple composition statistics for the file analyzed.

- Filename: The original filename of the file which was analyzed
- 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. At the moment these will always be the same. In the future it may be possible to analyze just a subset of sequences and estimate the total number, to speed up the analysis, but since we have found that problematic sequences are not evenly distributed through a file we have disabled this for now.
- 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 the number of sequences actually used for the rest of the analysis.

- 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



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:

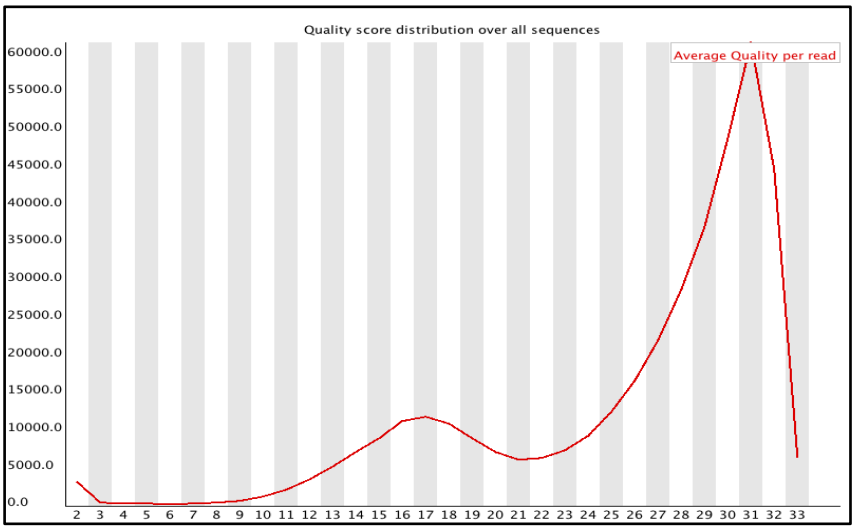
- The central red line is the median value
- The yellow box represents the inter-quartile range (25-75%)
- The upper and lower whiskers represent the 10% and 90% points
- The blue line represents the mean quality

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

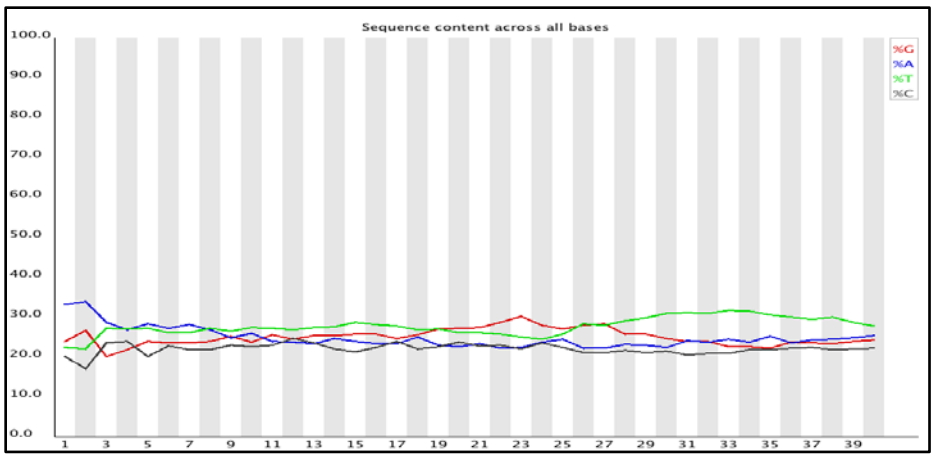
The per sequence quality score report allows you to see if a subset of your sequences have universally low quality values. It is often the case that a subset of sequences will have universally poor quality, often because they are poorly imaged (on the edge of the field of view etc), however these should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

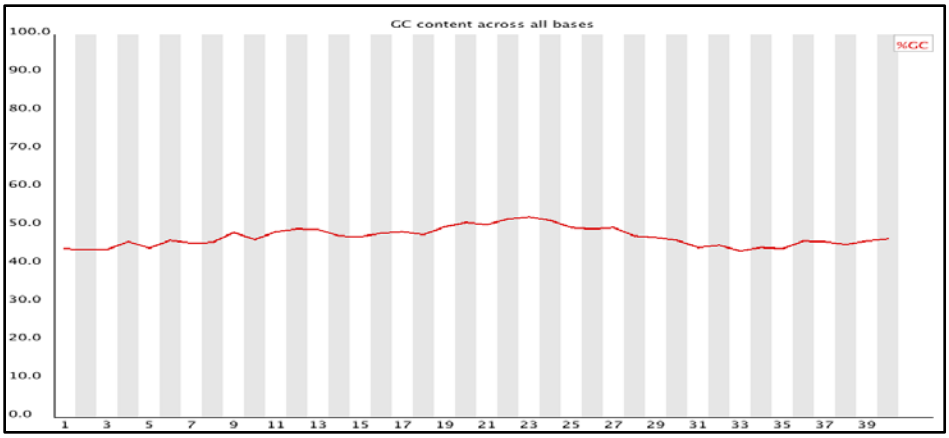
Per Base Sequence Content plots out the proportion of each base position in a file for which each of the four normal DNA bases has been called.



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

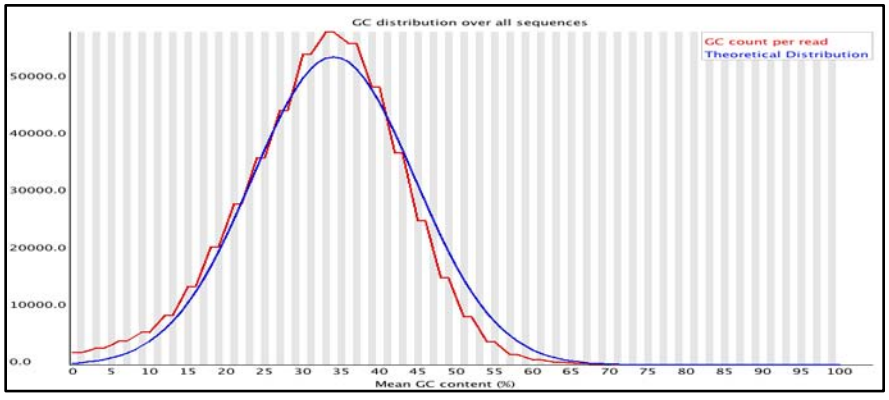
Per Base GC Content plots out the GC content of each base position in a file. In a random library you would expect that there would be little to no difference between the different bases of a sequence run, so the line in this plot should run horizontally across the graph. The overall GC content should 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 it 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 modeled normal distribution of GC content.

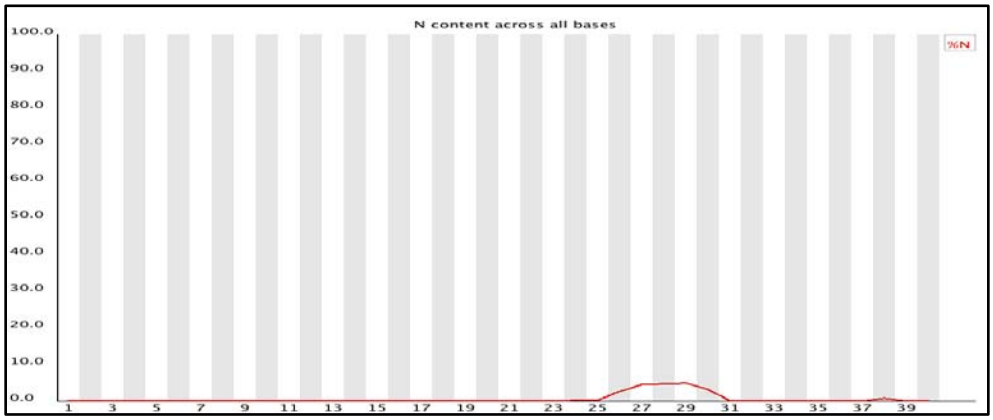


In a normal random library you would expect to see a roughly normal distribution of GC content where the central peak corresponds to the overall GC content of the underlying genome. Since we don't know the the GC content of the genome the modal GC content is calculated from the observed data and used 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.

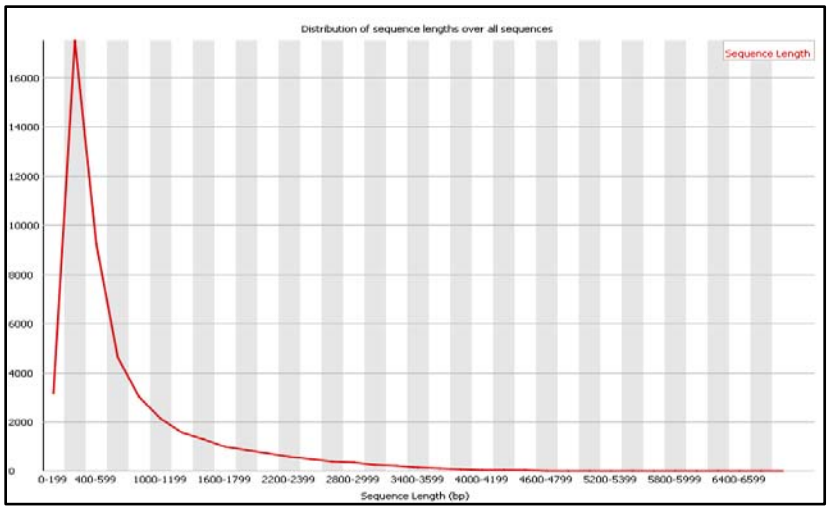


It's not unusual to see a very low proportion of Ns appearing in a sequence, especially nearer the end of a sequence. However, if this proportion rises above a few percent it suggests that the analysis pipeline was unable to interpret the data well enough to make valid base calls. This module raises 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

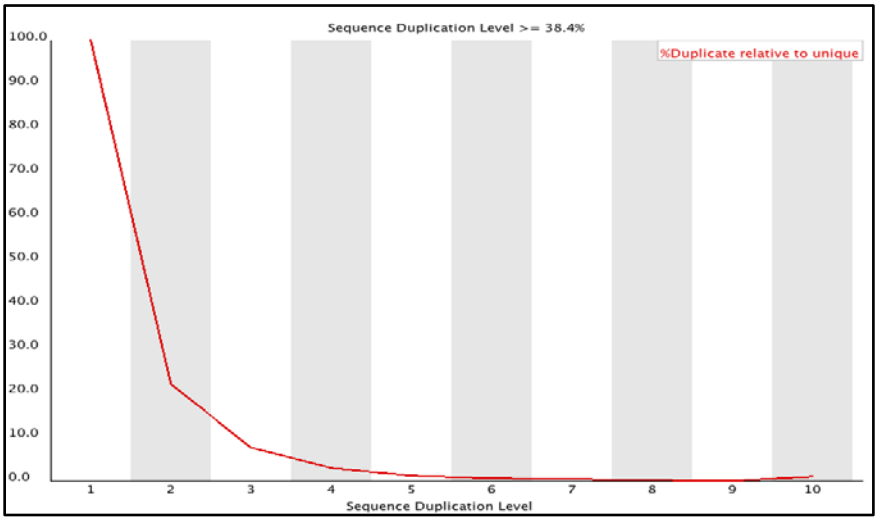
Some high throughput sequencers generate sequence fragments of uniform length, but others can contain reads of wildly varying lengths. Even within uniform length libraries some pipelines will trim sequences to remove poor quality base calls from the end. This module generates a graph showing the distribution 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.



To cut down on the memory requirements for this module only sequences which occur in the first 200,000 sequences in each file are analyzed, but this should be enough to get a good impression for the duplication levels in the whole file. 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 - so it's not unusual to see a small rise in this final 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

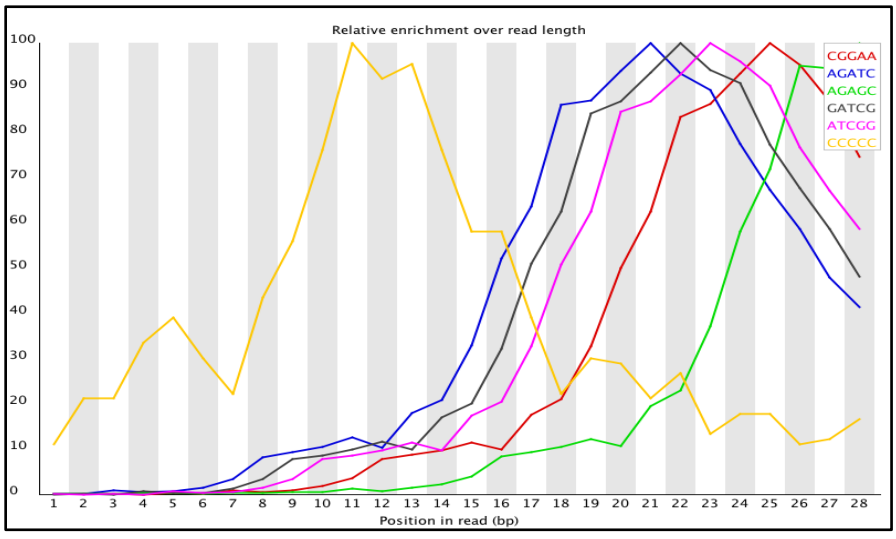
A normal high-throughput library will contain a diverse set of sequences, with no individual sequence making up a tiny fraction of the whole. Finding that a single sequence is very overrepresented in the set either means that it is highly biologically significant, or indicates that the library is contaminated, 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.

(xi) Over represented K-mers

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

TOOL 1: FastQC Task 1: Open FastQC /FastQC/fastqc **Task 2:** Upload read files File > Open = Upload read file in fastq format **Task 3:** Do the quality checks

TOOL 2: FASTX TOOLKIT Task 1: Generate quality score statistics

```
fastx_quality_stats -Q33 -i <r1.fq> -o <r1.stats>
fastx_quality_stats -Q33 -i <r2.fq> -o <r2.stats>
```

Task 2: Generate quality score boxplot

```
fastq_quality_boxplot_graph.sh -i <r1.stats> -o <r1.png>
fastq_quality_boxplot_graph.sh -i <r2.stats> -o <r2.png>
```

Task 3: Trimming reads

```
fastx_trimmer -Q33 -f <int> -l <int> -i <read.fq> -o <t_read.fq> or fastx_trimmer
-Q33 -t <int> -i <read.fq> -o <t_read.fq>
```

Task 4: Quality filtering

```
fastq_quality_filter -Q33 -v -q <int> -p <int> -i <r1.fq> -o <f_r1.fq>
fastq_quality_filter -Q33 -v -q <int> -p <int> -i <r2.fq>  -o <f_r2.fq>
```

TOOL 3: Fastq Screen

Task 1: Configure database Edit the "fastq_screen.conf" in order to configure database required to run fastq screen.

Task 2: Create bowtie 2 index bowtie2-build ref.fa ref_bowtie2

Example format:

```
############################## DATABASE Bacteria_Bowtie2
/fastq_screen/ref_bowtie2 BOWTIE2 ##############################
```

Task 3: Download the PhiX data Download the PhiX data and configure it in "fastq_screen.conf"

Task 4: Run the screening Single-end reads fastq_screen --aligner bowtie2 reads.fastq Paired-end reads fastq_screen --aligner bowtie2 --paired r1.fq r2.fq

TOOL 4: Cutadapt Task 1: Discard sequencing adapters cutadapt -b <adapter sequence> --discard-trimmed r1.fq > adp_r1.fq cutadapt -b <adapter sequence> --discard-trimmed r2.fq > adp_r2.fq

Further Reading

1. *Patel RK, Jain M (2012). NGS QC Toolkit: A toolkit for quality control of next generation sequencing data. PLoS ONE, 7(2): e30619.*

Chapter 5

Genome Alignment & Assembly

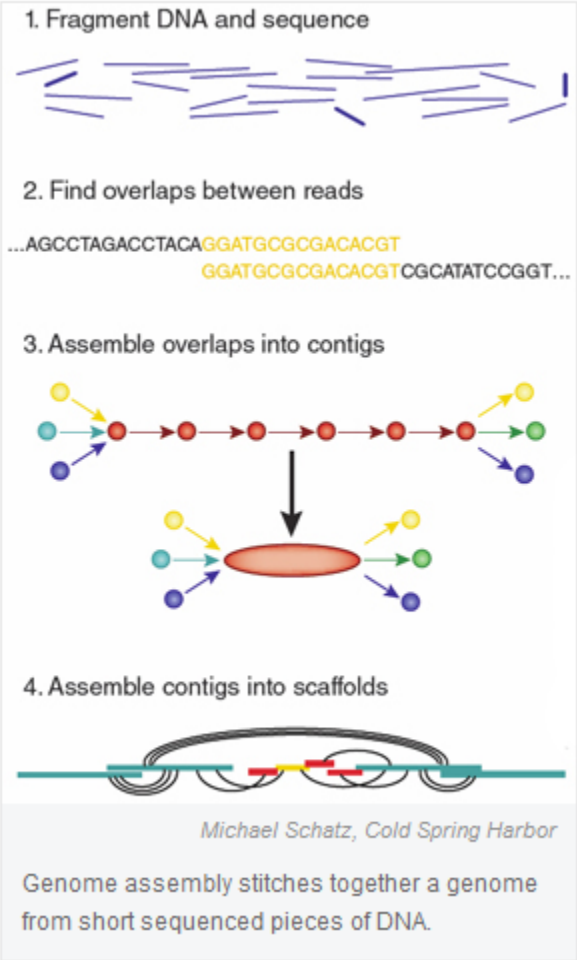
Anantharamanan R.

Introduction

In bioinformatics, sequence assembly refers to aligning and merging fragments of a much longer DNA sequence in order to reconstruct the original order of the sequence. Genome assembly is simply putting sequenced fragments of DNA into their correct chromosomal positions. This is needed as DNA sequencing technology cannot read whole genomes in one go, but rather reads small pieces of between 20 and 1000 bases, depending on the technology used. Typically the short fragments, called reads, result from sequencing genomic DNA, or gene transcript (ESTs). It is the process of taking a large number of short DNA sequences and putting them back together to create a representation of the original chromosomes from which the DNA originated. In a shotgun sequencing project, the entire DNA from a source (usually a single organism, anything from a bacterium to a mammal) is first fractured into millions of small pieces. These pieces are then "read" by automated sequencing machines, which can read up to 1000 nucleotides or bases at a time. (The four bases are adenine, guanine, cytosine, and thymine, represented as AGCT.) A genome assembly algorithm works by taking all the pieces and aligning them to one another, and detecting all places where two of the short sequences, or *reads*, overlap. These overlapping reads can be merged, and the process continues.

Genome assembly is a very difficult computational problem, made more difficult because many genomes contain large numbers of identical sequences, known as repeats. These repeats can be thousands of nucleotides long, and some occur in thousands of different locations, especially in the large genomes of plants and animals. The resulting (draft) genome sequence is produced by combining the information sequenced contigs and then employing linking information to create scaffolds. Scaffolds are positioned along the physical map of the chromosomes creating a "golden path".

To assemble a genome, computer programs typically use data consisting of single and paired reads. Single reads are simply the short sequenced fragments themselves; they can be joined up through overlapping regions into a continuous sequence known as a 'contig'. Repetitive sequences, polymorphisms, missing data and mistakes eventually limit the length of the contigs that assemblers can build.



Given a set of sequence fragments the object is to find the shortest common super sequence.

1. calculate pairwise alignments of all fragments
2. choose two fragments with the largest overlap
3. merge chosen fragments
4. repeat step 2. and 3. until only one fragment is left

5. Paired reads typically are about the same length as single reads, but they come from either end of DNA fragments that are too long to be sequenced straight through. Depending on the preparation technique, that distance can be as short as 200 base pairs or as large as several tens of kilobases. Knowing that paired reads were generated from the same piece of DNA can help link contigs into 'scaffolds', ordered assemblies of contigs with gaps in between. Paired-read data can also indicate the size of repetitive regions and how far apart contigs are.

Genome Assemblers

The first sequence assemblers began to appear in the late 1980s and early 1990s as variants of simpler sequence alignment programs to piece together vast quantities of fragments generated by automated sequencing instruments called DNA sequencers. Originally, most large-scale DNA sequencing centers developed their own software for assembling the sequences that they produced. However, this has changed as the software has grown more complex and as the number of sequencing centers has increased. An example of such assembler *Short Oligonucleotide Analysis Package* developed by BGI for de novo assembly of human-sized genomes, alignment, SNP detection, resequencing, indel finding, and structural variation analysis. As the sequenced organisms grew in size and complexity from small viruses over plasmids to bacteria and finally eukaryotes, the assembly programs used in these genome projects needed to increasingly employ more and more sophisticated strategies to handle:

- Terabytes of sequencing data which need processing on computing clusters;
- Identical and nearly identical sequences (known as *repeats*) which can, in the worst case, increase the time and space complexity of algorithms exponentially;
- Errors in the fragments from the sequencing instruments, which can confound assembly.

De-novo vs. mapping (reference) assembly

In sequence assembly, two different types can be distinguished:

- **De-novo**: assembling short reads to create full-length (sometimes novel) sequences. Overlapping reads are presumed to be from the same area of the genome, for *de-novo* assembly, read depth should be more.
- **Mapping or Reference:** assembling reads against an existing backbone sequence taken as reference, building a sequence that is similar but not necessarily identical to the backbone sequence. Combines all sequencing reads into contigs based on sequence similarity between reads and reference.

In terms of complexity and time requirements, de-novo assemblies are orders of magnitude slower and more memory intensive than mapping assemblies. This is mostly due to the fact that the assembly algorithm needs to compare every read with every other read (an operation that has a complexity of O (n^2) but can be reduced to O(n log(n)). Referring to the comparison drawn to shredded books in the introduction: while for mapping assemblies one would have a very similar book as template (perhaps with the names of the main characters and a few locations changed), the de-novo assemblies are more hardcore in a sense as one would not know beforehand whether this would become a science book, or a novel, or a catalogue etc.

Reference Assembly

One of the most complex and computationally intensive tasks of genome sequence analysis is genome assembly. Even today, few centers have the resources, in both software and hardware, to assemble a genome from the thousands or millions of individual sequences generated in a whole-genome shotgun sequencing project. With the rapid growth in the number of sequenced genomes has come an increase in the number of organisms for which two or more closely related species have been sequenced. This has created the possibility of building a comparative genome assembly algorithm, which can assemble a newly sequenced genome by mapping it onto a reference genome. Reference assisted assembly requires the use of a reference sequence, as a model, to assist in the assembly of the novel genome. The standard method for identifying the best reference sequence for the assembly of a novel genome aims at counting the number of reads that align to the reference sequence, and then choosing the reference sequence which has the highest number of reads aligning to it.

De novo assembly

A new generation of sequencing technologies is revolutionizing molecular biology. Illumina's Solexa and Applied Biosystems' SOLiD generate gigabases of nucleotide sequence per week. However, a perceived limitation of these ultra-high-throughput technologies is their short read-lengths. De novo assembly of sequence reads generated by classical Sanger capillary sequencing is a mature field of research. Unfortunately, the existing sequence assembly programs were not effective for short sequence reads generated by Illumina and SOLiD platforms. Early studies suggested that, in principle, sequence reads as short as 20-30 nucleotides could be used to generate useful assemblies of both prokaryotic and eukaryotic genome sequences, albeit containing many gaps. The early feasibility studies and proofs of principle inspired several bioinformatics research groups to implement new algorithms as freely available software tools specifically aimed at assembling reads of 30-50 nucleotides in length. This has led to the generation of several draft genome sequences based exclusively on short sequence Illumina sequence reads, recently culminating in the assembly of the 2.25-Gb genome of the giant panda from Illumina sequence reads with an average length of just 52 nucleotides.

Velvet is a novel set of de Bruijn graph-based sequence assembly methods for very short reads that can both remove errors and, in the presence of read pair information, resolve a large number of repeats. With unpaired reads, the assembly is broken when there is a repeat longer than the *k*-mer length. With the addition of short reads in read pair format, many of these repeats can be resolved, leading to assemblies similar to draft status in bacteria and reasonably long (~5 kb) SCSCs in eukaryotic genomes. Velvet can convert high-coverage very short reads into reasonably sized contigs with no additional information. With additional paired read information to resolve small repeats, almost complete genomes can be assembled. We believe the Velvet framework will provide a rich set of different algorithmic options tailored to different tasks and thus provide a platform for cheap de novo sequence assemblies, eventually for all genomes.

Assessing Assembly Quality

Common measures of quality are:

- number and sizes of contigs
- Assumption: few large contigs is better than many small contigs.
- True because there are fewer gaps in the former, but, does not account for the possibility of misassembling.

Tools/ Software's for assembly

- TIGR Assembler
- Velvet (Denovo)
- Maq (Reference)
- Reference assembly& Alignment using BWA tool and Visualization of alignment using SAM

Applications of Genome assembly

- Generating and interpreting alignment status and reports
- Genome variation calling (finding SNP's, *indels*)
- Variation annotation and Viewing

Practical

TOOL 5: BOWTIE 2

Task 1: Creating reference index

Build bowtie index from a set of DNA sequences. Bowtie2-build outputs a set of 6 files with suffixes .1.bt2, .2.bt2, .3.bt2, .4.bt2, .rev.1.bt2, and .rev.2.bt2. These files together constitute the index: they are all that is needed to align reads to that reference. The original sequence FASTA files are no longer used by Bowtie 2 once the index is built.

Command:

bowtie2-build [options] <reference_sequence> <index_name>*

Example:

bowtie2-build ref.fa ref_bowtie2

Task 2:

Aligning reads against reference sequence

Command:

bowtie2 -x <reference index> -p <number of processors to be used> -S (generate output in SAM format) -1 <read_1> -2 <read_2> <output.sam>

Example:

bowtie2 -x ref_bowtie2 -p 10 -S -1 r1.fq -2 r2.fq aln.sam

TOOL 6: SAMTOOL

Samtools is a set of utilities that manipulate alignments in the BAM format. It imports from and exports to the SAM (Sequence Alignment/Map) format, does sorting, merging and indexing.

Task 3:

Index reference sequence in the FASTA format. faidx will index the file and create <ref.fasta>.fai on the disk.

Command:

samtools faidx ref.fa

Task 4:

Import SAM to BAM

Command:

samtools view -bS aln.sam > aln.bam

-b: Output in the BAM format. -s: Input in the SAM format.

Task 5:

Sort alignments by leftmost coordinates. File <out.prefix>.bam will be created. This command may also create temporary files <out.prefix>.%d.bam when the whole alignment cannot be fitted into memory

Command:

samtools sort aln.bam aln.sorted

Task 6:

Index sorted alignment for fast random access. Index file <aln.bam>.bai will be created.

Command:

samtools index aln.sorted.bam

Task 7:

Remove potential PCR duplicates

Command:

samtools rmdup -S aln.sorted.bam rmdup.bam

-S : Treat paired-end reads and single-end reads

Task 8:

Sort alignments by leftmost coordinates. File <out.prefix>.bam will be created. This command may also create temporary files <out.prefix>.%d.bam when the whole alignment cannot be fitted into memory

Command:

samtools sort rmdup.bam rmdup_ aln.sorted

Task 9:

Index sorted alignment for fast random access. Index file <aln.bam>.bai will be created.

Command:

samtools index rmdup_aln.sorted.bam

Further Reading

1. *Michael C. Schatz, Arthur L. Delcher, and Steven L. Salzberg 2010. Assembly of large genomes using second-generation sequencing. Genome Res. 20: 1165-1173*

2. *Butler J, MacCallum I, Kleber M, Shlyakhter IA, Belmonte MK, Lander ES, Nusbaum C, Jaffe DB. 2008. ALLPATHS: De novo assembly of whole-genome shotgun microreads. Genome Res 18: 810–820.*

3. *Jaffe DB, Butler J, Gnerre S, Mauceli E, Lindblad-Toh K, Mesirov JP, Zody MC, Lander ES. 2003. Whole-genome sequence assembly for mammalian genomes: Arachne 2. Genome Res 13: 91–96.*

Chapter 6

Variant Calling

Anantharamanan R.

Practicals

TOOL 7: BCFTOOLS

Task 10:

Generate BCF or pileup for one or multiple BAM files.

Command:

samtools mpileup -ugf ref.fa rmdup_aln.sorted.bam | bcftools view -bvcg - > var.raw.bcf

-u Generate uncompress BCF output
-g Generate BCF output
-f Faidx indexed reference sequence file

Task 11:

Call SNPs and short INDELs from BCF files

Command:

bcftools view var.raw.bcf | vcfutils.pl varFilter -D 100 > 100_var.flt.vcf

Chapter 7

Plant Transcriptome Mapping

Dr. D. Prasath

Introduction

The transcriptome encompasses the set of transcripts from a cell or a population of cells, which include protein-coding mRNAs and non-coding small RNAs (e.g. ribosomal, tRNA, miRNA). Traditionally, transcriptome profiling, or transcriptomics, has focused on quantifying gene expression. With the advent of Ultra high-throughput sequencing (UHTS) technologies, it is now possible to obtain highly resolved structural information of RNA populations on a high-throughput platform. This includes mapping transcript initiation and termination sites, splice junctions and post-transcriptional modifications. Such information will lead to a better understanding of the functional elements within the genome and the discovery of novel developmental or environmental regulatory networks.

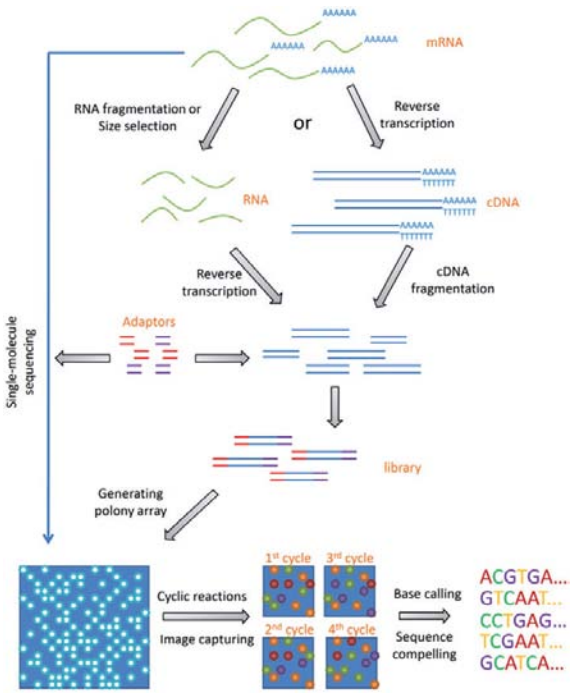
EST sequencing has traditionally been the core technology used for the discovery of reference transcripts. However, it has some inherent limitations, such as low throughput, high cost and a long experimental cycle. In recent years, researchers have developed a high-throughput sequencing technology called Next Generation Sequencing (NGS). Various platforms utilize NGS, such as the Illumina Genome Analyzer, the Roche/454 Genome Sequencer FLX Instrument, and the ABI SOLiD System; these have proven to be powerful and cost-effective tools for advanced research in many areas, including genome sequencing, genome resequencing, miRNA expression profiling, DNA methylation analysis, and especially the de novo transcriptome sequencing of non-model organisms. This method of transcriptome analysis is fast and simple because it does not require bacterial cloning of the cDNAs. Instead, direct cDNA sequencing generates an extraordinary depth of short reads. It is a more comprehensive and efficient way to measure transcriptome composition, obtain RNA expression patterns, and discover new genes. In addition, this approach is very sensitive, and thus allows the detection of low-abundance transcripts. Illumina genome analyzer based sequencing technology (Illumina, USA) yields huge amount of short reads with high coverage. Assembling such short reads is a challenging task, more so in the absence of reference sequences. A few bioinformatics tools have been developed for de novo assembly using short-read sequence data which vary in their success and application, and depends upon data specific strategies.

RNA sequencing

RNA-seq refers to the deep-sequencing of RNA pools. While UHTS often refers to deep sequencing of mRNAs, any RNA population can be analyzed. Two methods are typically used to capture and sequence RNA pools. In both methods mRNA pools are enriched by capturing the

molecules through the polyadenylated tails, and a ribosomal RNA removal step is often added before or after the mRNA purification. In one method, mRNA enriched pools are then fragmented into roughly equal lengths and then reverse-transcribed using random hexamers to generate a cDNA library. Alternatively, RNA is reverse transcribed using an oligo-dT adapter and the resulting cDNA is fractionated. The former method has the advantage of more uniform representation across the coding region of the transcript, but may result in the under-representation 50 and 30 sequences. The later method provides good coverage of 30 sequences but biases against the body of the transcript. The cDNAs are then fitted with adaptors at one or both ends through a ligation step(s). It is desirable to add these adaptors during the single-strand stage (RNA or cDNA) synthesis step in order to retain strand specificity in the final sequence reads. The tagged cDNA library is subsequently amplified through PCR before being sequenced.

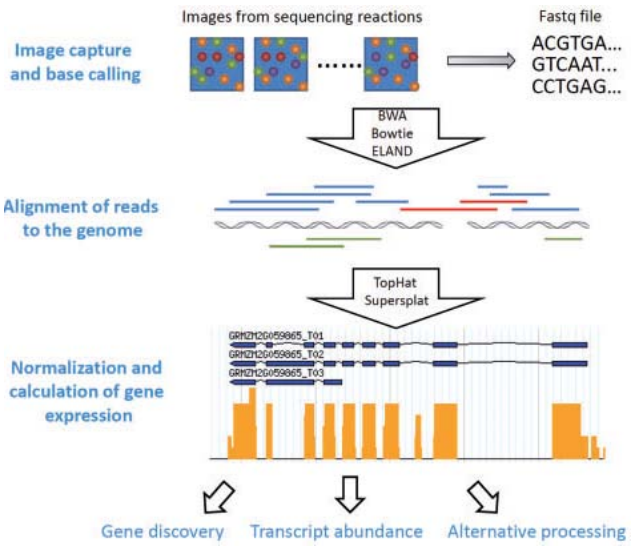
For a typical RNA-seq experiment, mRNA is isolated and reverse-transcribed (RT) into cDNA libraries with homogeneous lengths. This is achieved by either RNA or cDNA fragmentation. Recently, single-molecule capture methods have been developed (left) that obviate the need for a RT step. In the case of smRNA studies, total RNA is fractionated on acrylamide gels and smRNAs excised. Adaptors at one or both ends of the RNA are added prior to cDNA amplification and library construction. For the Solexa platform, cDNA molecules are anchored onto a polony array (flow cell) surface, which are then subjected to PCR amplification. Images are taken after each cycle for base calling and sequence generation. Currently for the Illumina platform, ~120 million single or paired-end reads of 32 - 80 nt are generated on a single flow cell (8 lanes/flow cell) which is then processed further depending on the research goals.



Overview of RNA-seq experimental procedures.

RNA sequencing analysis

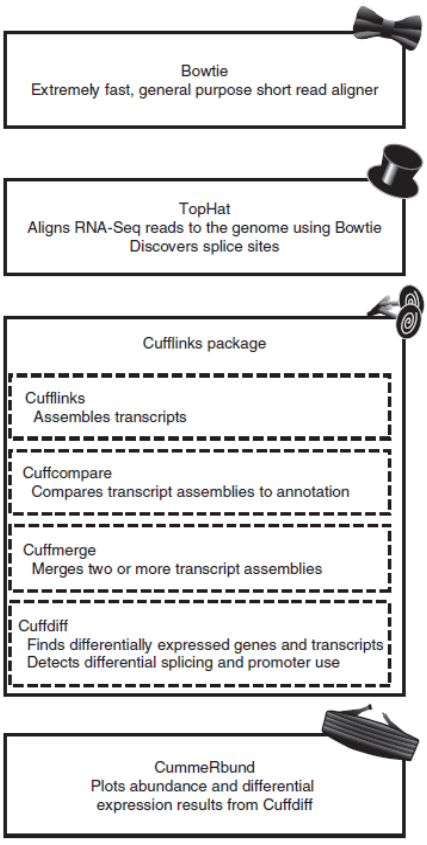
Overview of RNA-seq data analysis



Although the volume of data from RNA-seq experiments is often burdensome, it can provide enormous insight. Just as cDNA sequencing with Sanger sequencers drastically expanded our catalog of known human genes, RNA-seq reveals the full repertoire of alternative splice isoforms in our transcriptome and sheds light on the rarest and most cell- and context-specific transcripts6. Furthermore, because the number of reads produced from an RNA transcript is a function of that transcript's abundance, read density can be used to measure transcript and gene expression with comparable or superior accuracy to expression microarrays. RNA-seq experiments must be analyzed with robust, efficient and statistically principled algorithms. Fortunately, the bioinformatics community has been hard at work developing mathematics, statistics and computer science for RNA-seq and building these ideas into software tools.

RNA-seq analysis tools generally fall into three categories:

(i) those for read alignment;
(ii) those for transcript assembly or genome annotation; and
(iii) those for transcript and gene quantification.



Differential gene and transcript expression analysis

The volume and complexity of data from RNA-seq experiments necessitate scalable, fast and mathematically principled analysis software. TopHat and Cufflinks are free, open-source software tools for gene discovery and comprehensive expression analysis of high-throughput mRNAsequencing (RNA-seq) data. Together, they allow biologists to identify new genes and new splice variants of known ones, as well as compare gene and transcript expression under two or more conditions. TopHat and Cufflinks performs such analyses, also covers several

accessory tools and utilities that aid in managing data, including CummeRbund, a tool for visualizing RNA-seq analysis results.

Ginger transcriptome analysis: A case study

Genomic tools are now being developed to accelerate the identification of resistance genes and the development of bacterial wilt resistant ginger. The present study describes the first global analysis of ginger and mango ginger *R. solanacearum* challenge inoculated regimes, which would serve as a blueprint of gene expression profile.

Ginger (*Zingiber officinale*) is affected by many diseases. Among them, bacterial wilt (*Ralstonia solanacearum)* is one of the most important production constraints in tropical, sub tropical and warm temperature regions of the world. In spite of extensive search, no resistance source could be located in ginger. However, mango ginger *(Curcuma amada)*, which is resistant to *R. solanacearum*, is a potential donor, if the exact mechanism of resistance is understood. Although ESTs and other cDNA sequences are among the most reliable evidences for the identification of gene-rich regions in a genome, gene identification and genome annotation, very less effort has been made for ginger-pathogen interaction. Our objective is deep sequencing of transcriptomes of both genotypes followed by *de novo* assembly and annotation.

The transcriptome has been sequenced from rhizome tissue samples after challenge inoculation with *R. solanacearum* using next generation technologies for short read (Illumina platform). It involves analysis of GC content, repeat content, putative functions, gene families, transcription factor encoding genes and its comparative analysis with and *C. amada*. The identification of many defense related genes differentially expressed provides many insights to resistance mechanism to *R. solanacearum* and for studying pathways involved in responses to pathogen. We also identified several candidate genes that may underline the difference in resistance to *R. solanacearum* between ginger and mango ginger.

Conclusion

Despite these challenges, UHTS-based transcriptomics approaches promise ‘never-before’ opportunities to explore plant transcriptomes. As improvements to the sequencing chemistry, sequencing hardware and software and statistical methods of analysis continue to progress, the expectations for transcriptomics studies will continue to increase. It has been speculated that the cost of sequencing a complete genome or transcriptome will not be a limiting factor in the foreseeable future. This may allow experiments that were deemed as economically unfeasible in the past to be routinely performed. In summary, UHTS-based approaches have clearly demonstrated their advantages over previously developed methods and are becoming the new standard for transcriptomics studies.

Further Reading

1. *Annadurai RS, Neethiraj R, Jayakumar V, Damodaran AC, Rao SN, et al. (2013) De Novo Transcriptome Assembly (NGS) of Curcuma longa L. Rhizome Reveals Novel Transcripts Related to Anticancer and Antimalarial Terpenoids. PLoS ONE 8(2): e56217. doi:10.1371/journal.pone.0056217*

2. *Barakat A, DiLoreto D, Zhang Y, Smith C, Baier K, Powell W, Wheeler N, Sederoff R, Carlson J. 2009. Comparison of the transcriptomes of American chestnut (Castanea dentata) and Chinese chestnut (Castanea mollissima) in response to the chestnut blight infection. BMC Plant Biology, 9(1): 51.*

3. *Garg R, Patel RK, Jhanwar S, Priya P, Bhattacharjee A, Yadav G, Bhatia S, Chattopadhyay D, Tyagi AK, Jain M (2011). Gene discovery and tissue-specific transcriptome analysis in chickpea with massively parallel pyrosequencing and web resource development. Plant Physiol., 156: 1661-1678.*

4. *Grabherr MG, Haas BJ, Yassour M, Levin JZ, Thompson DA, Amit I, Adiconis X, Fan L, Raychowdhury R, Zeng Q, Chen Z, Mauceli E, Hacohen N, Gnirke A, Rhind N, di Palma F, Birren BW, Nusbaum C, Lindblad-Toh K, Friedman N, Regev A. 2011. Full-length transcriptome assembly from RNA-Seq data without a reference genome. Nat. Biotechnol. 15: 29(7): 644-52.*

5. *Jhanwar S, Priya P, Garg R, Parida SK, Tyagi AK, Jain M (2012). Transcriptome sequencing of wild chickpea as a rich resource for marker development. Plant Biotechnology J., 10: 690-702*

6. *Li, B. and Dewey, C. N. (2011). RSEM: accurate transcript quantification from RNA-Seq data with or without a reference genome.* BMC Bioinformatics *12: 323.*

7. *Trapnell, C. et al. 2012. Differential gene and transcript expression analysis of RNA-seqexperiments with TopHatand Cufflinks. Nature Protocols 7: 562–578.*

8. *Wang L, Li P, Brutnell TP (2010) Exploring plant transcriptomes using ultra high-throughput sequencing. Briefings in Functional Genomics 9: 118–128.*

9. *Tophat Manual: http://tophat.cbcb.umd.edu/manual.html*

10. *Cufflinks Manual: http://cufflinks.cbcb.umd.edu/manual.html*

Chapter 8

Functional Annotation

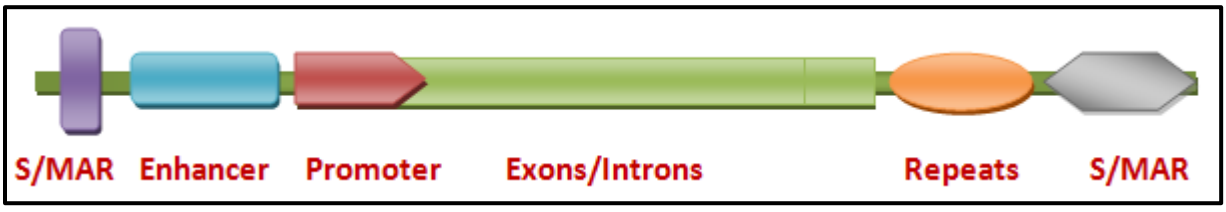
Rosana O. Babu

Introduction

Obtaining the sequence from whole genome sequencing and assembling is not the end of a genome project, however finding and attaching the structural elements and its related function are the next major steps, which are called "Genome Annotation". The process of identifying the locations of genes and all of the coding regions in a genome and determining what those genes do. An annotation (irrespective of the context) is a note added by way of explanation or commentary. Once a genome is sequenced, it needs to be annotated to make sense of it. Annotation is the process of adding pertinent information about the raw DNA sequences to the genome or process of attaching biological information to DNA or Protein sequences by describing different regions of the code and identifying which regions can be called genes and thereby its products and functions. This include spotting locations of genes, total number of genes, coding regions, intron-exon structure, start and stop codons, intron lengths, alternative splicing, SNP's, InDels and untranslated regions (UTRs) as well as and determining what those genes do along with the gene product and functional information. Once a genome is sequenced, it needs to be annotated to make sense of it. With many genomes now sequenced, computational annotation methods to characterize genes and proteins from their sequence are increasingly important. It consists of two main steps:

- Identifying elements on the genome- gene structure prediction
- Attaching biological information to these elements- gene function prediction

There are various parts within the gene with different functions, some may code for protein, others may contain regulatory information, some may form introns and will not be translated and their function is still unclear. The diagram shown below represents fragment of DNA, with single hypothetical gene. Each region has to be annotated from DNA sequences based on similarity searches or literature reviews.



Obviously computer programs are essential to this process; however, human brains are often required to evaluate computer-generated gene models. Several Automatic annotation tools are available that are highly accurate. Annotation tools can perform all this by computer analysis, as opposed to manual annotation which involves human expertise. Ideally, these approaches co-exist and complement each other in the same annotation pipeline.

The basic level of annotation uses BLAST for finding similarities, and then annotating genomes based on that. However, much additional information is available to annotation platform nowadays. Some databases use genome context information, similarity scores, experimental data, and integrations of other resources to provide genome annotations through their Subsystems approach. Other databases such as Ensemble rely on both curated data sources as well as a range of different software tools in their automated genome annotation pipeline.

Annotation Methods

- *Ab-Initio* - Statistics-based methods
- Homology based methods

The first hurdle for any functional annotation process is to define 'function'. In general however, the problem is multi-dimensional: a protein can have a molecular function, a cellular role, and be part of a functional complex or pathway (these are the distinctions used in the Gene Ontology. Furthermore, certain aspects of molecular function can be illustrated by multiple descriptive levels (for example, the coarse 'enzyme' category versus a more specific 'protease' assignment). Even the more detailed definition would not reveal the cellular role of the protein (apoptosis, metabolism, blood coagulation, and so on). Most function-prediction methods, both sequence and structure based, rely on inferring relationships between proteins that permit the transfer of functional annotations and binding specificities from one to the other. A notable challenge here is deciphering the connection between the detected similarities (structural or in sequence) and the actual level of functional relatedness. Function is often associated with domains, and another problem is the identification of functional domains from sequence alone. The accuracy of current methods for predicting domain boundaries is not yet completely satisfactory. Several methods provide reliable predictions if a structural template for the protein is available, but when this is not the case, one is left with the problem of whether the experimental annotation used for the inference refers to the same domain for which the sequence similarity/motif is established.

The function of a protein can also be inferred from its evolutionary relationship with proteins of known function, provided that the relationship is properly inspected. Orthologous proteins in different species most often share function, but paralogy (that is, divergence following duplication of the original gene) does not guarantee common function. Distinguishing between orthology and paralogy can be attempted on the basis of observed sequence-similarity patterns,

by analyzing the specific conservation pattern of residues responsible for function in the family, or on the basis of the protein structure (either experimentally determined or modeled). In all cases, this requires the clustering of proteins into evolutionary families, which can be achieved using similarity-detection tools such as BLAST or profiling tools based on multiple sequence alignments, for example, PSI-BLAST. Several available resources provide pre-compiled family assignments for proteins on a genomic scale, based only on their sequence. Resources can be subdivided into those that consider full-length sequences and those based on domains or motifs that map to certain sub-sequences. In both cases, the degree of granularity of the classification is important, as this is related to the level of functional features that a group of proteins is expected to share.

The accuracy of sequence-based methods is affected by the type and amount of information on the specific protein family but, overall, they seem to be reasonably accurate. Their success rate has been shown to be greater than 70% when tested on a limited dataset (all structures solved by the Midwest Center for Structural Genomics during the first five years of the Protein Structure Initiative)

Structural & Functional Genome Annotation

Structural annotation consists of the identification of genomic elements.

- Open reading frame and their localization
- Gene structure
- Coding regions
- Location of regulatory motifs

Functional annotation consists of attaching biological information to genomic elements.

- Biochemical function
- Biological function
- Involved regulation and interactions
- Expression

These steps may involve both biological experiments and *in-silico* analysis. A variety of software tools have been developed to permit scientists to view and share genome annotations.

- Structural annotation for prokaryotes & eukaryotes – PRODIGAL & AUGUSTUS
- Functional annotation (based gene level homology and protein level homology)

Practical

Structural Annotation

1. Eukaryotes- AUGUSTUS (Gene structural prediction with a genome model)

AUGUSTUS is used in many genome annotation projects. Accuracy measure use *sensitivity* (Sn) and *specificity* (Sp) for a feature (coding base, exon, transcript, gene); the sensitivity is defined as the number of correctly predicted features divided by the number of annotated features. The specificity is the number of correctly predicted features divided by the number of predicted features. A predicted exon is considered correct if both splice sites are at the annotated position of an exon. A predicted transcript is considered correct if all exons are correctly predicted and no additional exons not in the annotation. A predicted gene is considered correct if any of its transcripts are correct, i.e. if at least one isoform of the gene is exactly as annotated in the reference annotation.

Usage:

```
~/Programs/augustus.2.5.5/bin/augustus --strand=both --genemodel=partial --singlestrand=true --alternatives-from-evidence=true --alternatives-from-sampling=true --progress=true --gff3=on --uniqueGeneId=true --species=magnaporthe_grisea Input_genome.fasta > structural_annotation.gff
```

2. Prokaryotes – PRODIGAL

Prodigal (Prokaryotic Dynamic Programming **G**enefinding **Al**gorithm) is a microbial (bacterial and archaeal) gene finding program. Prodigal is an extremely fast gene recognition tool (written in very vanilla C). It can analyze an entire microbial genome in 30 seconds or less. Prodigal is a highly accurate gene finder. It correctly locates the 3' end of every gene in the experimentally verified Ecogene data set (except those containing introns). It possesses a very sophisticated ribosomal binding site scoring system that enables it to locate the translation initiation site with great accuracy (96% of the 5' ends in the Ecogene data set are located correctly). Prodigal's false positive rate compares favorably with other gene identification programs, and usually falls under 5%. Prodigal performs well even in high GC genomes, with over a 90% perfect match (5'+3') to the *Pseudomonas aeruginosa* curated annotations. Prodigal can run in metagenomic mode and

analyze sequences even when the organism is unknown. Prodigal can be run in one step on a single genomic sequence or on a draft genome containing many sequences. It does not need to be supplied with any knowledge of the organism, as it learns all the properties it needs to on its own. Prodigal source code is freely available under the General Public License.

Usage:

```
~/Programs/augustus.2.5.5/bin/augustus --strand=both --genemodel=partial --singlestrand=true --alternatives-from-evidence=true --alternatives-from-sampling=true --progress=true --gff3=on --uniqueGeneId=true --species=magnaporthe_grisea Input_genome.fasta > structural_annotation.gff
```

Functional Annotation

Steps for functional annotation

Functional annotation requires creating database for homology searches. First download complete swissprot database and Complete NCBI and setup database for homology search using following commands.

Run separately each command for nucleotide and protein

1. setup blast database for nucleotide

```
formatdb -i ncbi.fasta -p F -o F -n database1
```

2. setup blast database for protein

```
formatdb -i uniprot_sprot.fasta -p T -o F -n database2
```

3. Blasting the annotations

(i) For nucleotide

```
blastall -p blastn -i sequence.fasta -d database1 -a 2 -m 8 -o blast_result_nucleotide.txt
```

(ii) For Protein

```
blastall -p blastx -i contigs.fa -d database2 -a 2 -m 8 -o blast_result_protein.txt
```

4. Sorting for blast min E-value

(i) For nucleotide

```
sort -k 1,1 -k 11,11g blast_result.txt > sorted_blast_result_nucleotide.txt
```

(ii) For Protein

```
sort -k 1,1 -k 11,11g blast_resul_prot.txt > sorted_blast_result_protein.txt
```

5. Further filtering for best blast hit

(i) For nucleotide

```
awk '{if($1!=id){print};id=$1}' sorted_blast_result_nucleotide.txt>best_blast_result1.txt
```

(ii) For Protein

```
awk '{if($1!=id){print};id=$1}' sorted_blast_result_protein.txt>best_blast_result2.txt
```

6. Removing Positive strand blast hits

```
awk '{if($9>$10){print}}' best_blast_result1.txt> blast_result_neg.txt
```

7. Removing negative strand blast hits

```
awk '{if($10>$9){print}}' best_blast_result1.txt > blast result_pos.txt
```

Blast2GO Interface and its basic functions

- Annotation generation
- Annotation modulation
- Additional tools
- GO graph visualization

1. Load data: Go to \File" ->\Load FASTA File" and select your .fasta _le containing the set of sequences in FASTA format. Several example _le can be downloaded from the Blast2GO site. Alternatively you can load 10 example sequences into Blast2GO choosing \Load 10 Example Sequences" in the \File" menu.

2. BLAST: Go to \Blast" ->\Make BLAST". At the BLAST Con_guration Dialog- select the type of BLAST mode which is appropriate for your sequence type (blastx for nucleotide and blastp for protein data) and click on the top arrow to start the BLAST search against NCBI's non redundant NR database.

_ Once your BLAST analysis is _nished visualize your results at \Statistics" {>\BLAST Statistics".

_ On the Main Sequence Table, right-click on a sequence to open the Single Sequence Menu (Figure 15). Select Show BLAST Result to open the BLAST Browser for that sequence.

3. Mapping: Go to \Mapping" ->\Run GO-Mapping Step" and click on the top arrow to start mapping GO terms. Mapped sequences will turn green. Once Mapping is completed visualize your results at \Statistics" ->\Mapping Statistics".

4. Annotation: Go to \Annotation" ->\Run Annotation Step" and click on the top arrow to start the annotation. Annotated sequences will turn blue.

_ Once Annotation is completed visualize your results at \Statistics" ->\Annotation Statistics".

_ On the Main Sequence Table, right-click on a sequence to open the Single Sequence Menu. Select \Draw Graph of Annotations" to visualize the annotation on the GO DAG for that sequence.

_ If desired, modify the annotation by clicking with the left mouse button and select Change Annotation or change the extent of annotation by adding implicit terms (\Annotation"
->\Run ANNEX") or reducing to a GO-Slim representation (\Analysis"
->\Run GO-Slim").

During the annotation process, Enzyme Codes will be also given when a GO-term/EC

number equivalence is available.

_ Optionally, \InterProScan" ->\Run InterProScan" to obtain InterPro annotations is available and highly recommended to improve the annotation. Once InterProScan results are retrieved use \Merge InterProScan GOs to Annotation" to add GO terms obtained through motifs/domains to the current annotations.

5. Enrichment Analysis: Blast2GO provides tools for the statistical Analysis of GO term frequency di_erences between two sets of sequences. Go to \Analysis" ->\Enrichment Analysis" ->\Make Fisher's Exact Test". A new Dialog window is opend (Figure 25). Select a .txt _le containing a sequence IDs list for a subset of sequences. A test-set example _le can be downloaded form the Blast2GO web site. Select a second set of sequences as reference/background set if disired or skip this step and the whole actual loaded set of annotations present in Blast2GO will be used as reference. Click on run button to start the analysis. A table containing the results of this test will appear on the \Statistics Tab".
_ Click on \Make Enriched Graph" to visualize the results of the Fisher's Exact Test on the GO DAG.
_ Click on \Bar Chart" to obtain a bar chart representation of GO term frequencies.

6. Combined Graph: Blast2GO can visualize the combined annotation for a group of sequences on the GO DAG. Select a group of sequences to generate their combined graph at \Tools" ->\Select Sequences by Names". You can use the Demo Test Set used previously for this. Alternatively, you can select sequences manually using the sequence check boxes of the Main Sequence Table. Go to \Analysis" ->\Make Combined Graph". Set on the Seq Filter of the Combined Graph Dialog the minimal number of sequences collected at a GO term for its node to be shown in the generated graph. If you are using the Blast2GO Test Set example, set this value to 100. Click on the top arrow to generate the graph.

7. Save Results:

_ \File" ->\Save B2G-Project" saves the current Blast2GO project as .dat _le.
_ \File\ ->\Export allows to export the generated data in many di_erent formats. \Export Annotation" exports the actual annotation results as .annot _le.
_ \Enrichment Analysis" ->Export Results: exports results of the Fisher's Exact Test as a tabulator seperated text _le.
_ To save graph information, use the little icons/buttons on the corresponding graph windows. Graphs can be saved/exported as .pdf, .png, .svg and .txt.

Further Reading

1. *Ashburner M, Ball CA, Blake JA, Botstein D, Butler H, Cherry JM, Davis AP, Dolinski K, Dwight SS, Eppig JT, Harris MA, Hill DP, Issel-Tarver L, Kasarskis A, Lewis S, Matese JC, Richardson JE, Ringwald M, Rubin GM, Sherlock G: Gene ontology: tool for the unification of biology. The Gene Ontology Consortium. Nat Genet 2000, 25:25-29.*

2. *Tress M, Cheng J, Baldi P, Joo K, Lee J, Seo JH, Lee J, Baker D, Chivian D, Kim D, Ezkurdia I: Assessment of predictions submitted for the CASP7 domain prediction category.Proteins 2007, 69(Suppl 8):137-151.*

3. *Altschul SF, Gish W, Miller W, Myers EW, Lipman DJ: Basic local alignment search tool. J MolBiol 1990, 215:403-410.*

4. *Altschul SF, Madden TL, Schäffer AA, Zhang J, Zhang Z, Miller W, Lipman DJ: Gapped BLAST and PSI-BLAST: a new generation of protein database search programs. Nucleic Acids Res 1997, 25:3389-3402.*

5. *Watson JD, Sanderson S, Ezersky A, Savchenko A, Edwards A, Orengo C, Joachimiak A, Laskowski RA, Thornton JM: Towards fully automated structure-based function prediction in structural genomics: a case study. J MolBiol 2007, 367:1511-1522.*

Chapter 9

Comparative Genomics

Rosana O. Babu & Navish K Batchu

What is comparative genomics?

Comparative genomics is an exciting new field of biological research in which the genome sequences of different species - human, mouse and a wide variety of other organisms from yeast to chimpanzees - are compared. Comparative genomics is the study of the relationship of genome structure and function across different biological species or strains. Comparative genomics is an attempt to take advantage of the information provided by the signatures of selection to understand the function and evolutionary processes that act on genomes. While it is still a young field, it holds great promise to yield insights into many aspects of the evolution of modern species. The sheer amount of information contained in modern genomes (3.2 gigabases in the case of humans) necessitates that the methods of comparative genomics are automated. Gene finding is an important application of comparative genomics, as is discovery of new, non-coding functional elements of the genome.

By comparing the finished reference sequence of the human genome with genomes of other organisms, researchers can identify regions of similarity and difference. This information can help scientists better understand the structure and function of human genes and thereby develop new strategies to combat human disease. Comparative genomics also provides a powerful tool for studying evolutionary changes among organisms, helping to identify genes that are conserved among species, as well as genes that give each organism its unique characteristics.

Human FOXP2 gene and evolutionary conservation is shown in and multiple alignment (at bottom of figure) in this image from the UCSC Genome Browser. Note that conservation tends to cluster around coding regions (exons).

Comparative genomics exploits both similarities and differences in the proteins, RNA, and regulatory regions of different organisms to infer how selection has acted upon these elements. Those elements that are responsible for similarities between different species should be conserved through time (stabilizing selection), while those elements responsible for differences among species should be divergent (positive selection). Finally, those elements that are unimportant to the evolutionary success of the organism will be unconserved (selection is neutral).

One of the important goals of the field is the identification of the mechanisms of eukaryotic genome evolution. It is however often complicated by the multiplicity of events that have taken place throughout the history of individual lineages, leaving only distorted and superimposed traces in the genome of each living organism. For this reason comparative genomics studies of

small model organisms (for example the model *Caenorhabditiselegans* and closely related *Caenorhabditisbriggsae*) are of great importance to advance our understanding of general mechanisms of evolution.

Having come a long way from its initial use of finding functional proteins, comparative genomics is now concentrating on finding regulatory regions and siRNA molecules. Recently, it has been discovered that distantly related species often share long conserved stretches of DNA that do not appear to code for any protein (see conserved non-coding sequence). One such ultra-conserved region, that was stable from chicken to chimp has undergone a sudden burst of change in the human lineage, and is found to be active in the developing brain of the human embryo.

Computational approaches to genome comparison have recently become a common research topic in computer science. A public collection of case studies and demonstrations is growing, ranging from whole genome comparisons to gene expression analysis. This has increased the introduction of different ideas, including concepts from systems and control, information theory, strings analysis and data mining. It is anticipated that computational approaches will become and remain a standard topic for research and teaching, while multiple courses will begin training students to be fluent in both topics.

Benefits of comparative genomics

Dramatic results have emerged from the rapidly developing field of comparative genomics. Comparison of the fruit fly genome with the human genome reveals that about sixty percent of genes are conserved (Adams *et al.* 2000). That is, the two organisms appear to share a core set of genes. Researchers have also found that two-thirds of human genes known to be involved in cancer have counterparts in the fruit fly.

Using computer-based analysis to zero in on the genomic features that have been preserved in multiple organisms over millions of years, researchers will be able to pinpoint the signals that control gene function, which in turn should translate into innovative approaches for treating human disease and improving human health. In addition to its implications for human health, comparative genomics may benefit the broader animal world and ecological studies as well. As sequencing technology grows easier and less expensive, it will find wide applications in agriculture, biotechnology, and zoology as a tool to tease apart the often-subtle differences among animal and plant species. Such efforts might also lead to the rearrangement of our understanding of some branches of the evolutionary "tree of life," as well as point to new strategies for conserving rare and endangered species.

Why is there an increased interest in genomics?

Although living creatures look and behave in many different ways, all of their genomes consist of DNA, the chemical chain that makes up the genes that code for thousands of different kinds of proteins. Precisely which protein is produced by a given gene is determined by the sequence in which four chemical building blocks - adenine (A), thymine (T), cytosine (C) and guanine (G) - are laid out along DNA's double-helix structure.

In order for researchers to use an organism's genome most efficiently in comparative studies, data about its DNA must be in large, contiguous segments, anchored to chromosomes and, ideally, fully sequenced. Furthermore, the data needs to be organized to allow easy access for researchers using sophisticated computer software to conduct high-speed analyses.

The successful completion of the Human Genome Project in April 2003 has demonstrated that large-scale sequencing projects can generate high-quality data at a reasonable cost. As a result, the interest in sequencing the genomes of many other organisms has risen dramatically.

What other genomes have been sequenced?

In addition to sequencing the 3 billion letters in the human genetic instruction book, researchers involved in the Human Genome Project have already sequenced the genomes of a number of important model organisms that are commonly used as surrogates in studying human biology. These are the chimpanzee, the mouse, the rat, two puffer fish, two fruit flies, two sea squirts, two roundworms, baker's yeast and the bacterium *Escherichia coli*. Currently, sequencing centers supported by the National Human Genome Research Institute (NHGRI) of the National Institutes of Health (NIH) are close to completing working drafts of the chicken, the dog, the honey bee, the sea urchin and a set of four fungi. In the summer of 2003, the centers also began sequencing the genome of the rhesus macaque monkey, and many other organisms are in the sequencing pipeline.

Major outcomes of comparative genomics

The rapidly emerging field of comparative genomics has already yielded dramatic results. For example, a March 2000 study comparing the fruit fly genome with the human genome discovered that about 60 percent of genes are conserved between fly and human. Or, to put it simply, the two organisms appear to share a core set of genes.

Researchers have found that two-thirds of human genes known to be involved in cancer have counterparts in the fruit fly. Even more surprisingly, when scientists inserted a human gene associated with early-onset Parkinson's disease into fruit flies, they displayed symptoms similar to those seen in humans with the disorder, raising the possibility the tiny insects could serve as a new model for testing therapies aimed at Parkinson's.

More recently, a comparative genomic analysis of six species of yeast prompted scientists to significantly revise their initial catalog of yeast genes and to predict a new set of functional elements thought to play a role in regulating genome activity.

How Are Genomes Compared?

A simple comparison of the general features of genomes such as genome size, number of genes, and chromosome number presents an entry point into comparative genomic analysis. Data for several fully-sequenced model organisms is shown in Table 1. The comparisons highlight some striking findings. For example, while the tiny flowering plant Arabidopsis thaliana has a smaller genome than that of the fruit fly Drosophila melanogaster (157 million base pairs v. 165 million

base pairs, respectively) it possesses nearly twice as many genes (25,000 v. 13,000). In fact A. thaliana has approximately the same number of genes as humans (~25,000). Thus, a very early lesson learned in the "genomic era" is that genome size does not correlate with evolutionary status, nor is the number of genes proportionate to genome size.

Table 1. Comparative genome sizes of humans and other model organisms

Organism	Estimated size (base pairs)	Chromosome number	Estimated gene number
Human (*Homo sapiens*)	3 billion	46	~25,000
Mouse (*Mus musculus*)	2.9 billion	40	~25,000
Fruit fly (*Drosophila melanogaster*)	165 million	8	13,000
Plant (*Arabidopsis thaliana*)	157 million	10	25,000
Roundworm (*Caenorhabditis elegans*)	97 million	12	19,000
Yeast (*Saccharomyces cerevisiae*)	12 million	32	6,000
Bacteria (*Escherichia coli*)	4.6 million	1	3,200

Table 1 : Comparative genome sizes of humans and other model organisms

Figure 1: Conserved segments in the human and mouse genome

Human chromosomes, with segments containing at least two genes whose order is conserved in the mouse genome as color blocks. Each color corresponds to a particular mouse chromosome. Centromeres, are arranged in similar blocks in different species. The nature and extent of conservation of synteny differs substantially among chromosomes. For example, the X chromosomes are represented as single, reciprocal syntenic blocks. Human chromosome 20 corresponds entirely to a portion of mouse chromosome 2, with nearly perfect conservation of

order along almost the entire length, disrupted only by a small central segment. Human chromosome 17 corresponds entirely to a portion of mouse chromosome 11. Other chromosomes, however, show evidence of more extensive inter chromosomal rearrangement. Results such as these provide an extraordinary glimpse into the chromosomal changes that have shaped the mouse and human genomes since their divergence from a common ancestor 75–80 million years ago.

Comparison of discrete segments of genomes is also possible by aligning homologous DNA from different species. An example of such an alignment is shown in Figure 2, where a human gene (pyruvate kinase: PKLR) and the corresponding PKLR homologs from macaque, dog, mouse, chicken, and zebrafish are aligned. Regions of high DNA sequence similarity with human across a 12-kilobase region of the PKLR gene are plotted for each organism. Notice the high degree of sequence similarity between human and macaque (two primates) in both PKLR exons (blue) as well as introns (red) and untranslated regions (light blue) of the gene. In contrast, the chicken and zebra fish alignments with human only show similarity to sequences in the coding exons; the rest of the sequence has diverged to a point where it can no longer be reliably aligned with the human DNA sequence. Using such computer-based analysis to zero in on the genomic features that have been preserved in multiple organisms over millions of years, researchers are able to locate the signals that represent the location of genes, as well as sequences that may regulate gene expression. Indeed, much of the functional parts of the human genome have been discovered or verified by this type of sequence comparison (Lander *et al.* 2001) and it is now a standard component of the analysis of every new genome sequence.

subcentromeric heterochromatin of chromosomes 1, 9 and 16, and the repetitive short arms of 13, 14, 15, 21 and 22 are in black. (International Human Genome Sequencing Consortium; Lander, E. S. et al. 2001)

Finer-resolution comparisons are possible by direct DNA sequence comparisons between species. Figure 1 depicts a chromosome-level comparison of the human and mouse genomes that shows the level of synteny between these two mammals. Synteny is a situation in which genes

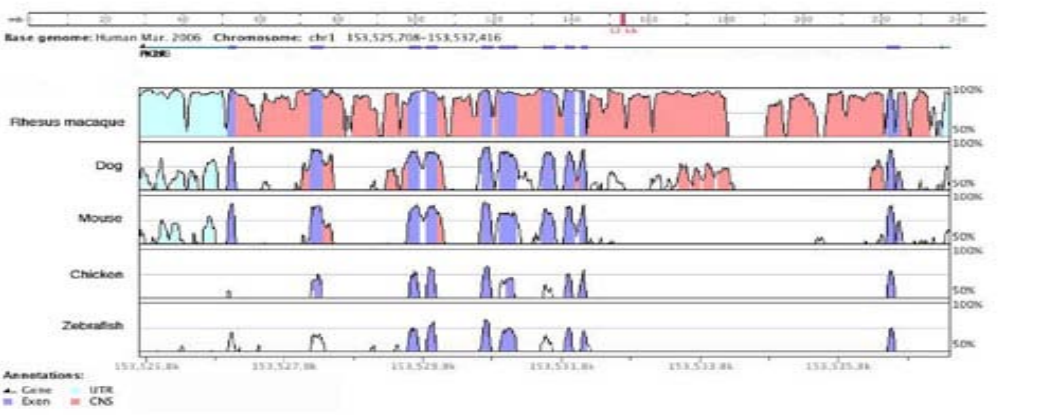


Figure 2: Human PKLR gene region compared to the macaque, dog, mouse, chicken, and zebra fish genomes

Numbers on the vertical axis represent the proportion of identical nucleotides in a 100-bp window for a point on the plot. Numbers on the horizontal axis indicate the nucleotide position from the beginning of the 12-kilobase human genomic sequence. Peaks shaded in blue correspond to the PKLR coding regions. Peaks shaded in light blue correspond to PKLR mRNA untranslated regions. Peaks shaded in red correspond to conserved non-coding regions (CNSs), defined as areas where the average identity is > 75%. Alignment was generated using the sequence comparison tool VISTA (http://pipeline.lbl.gov).

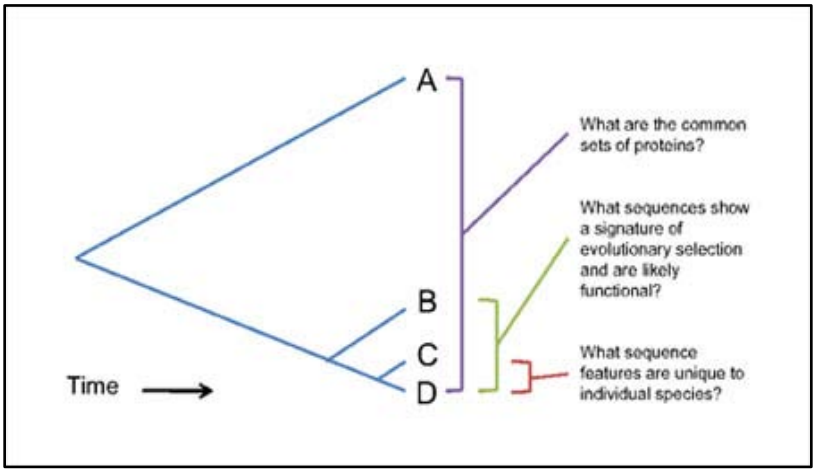


Figure 3: Comparisons of genomes at different phylogenetic distances serve to address specific questions.

We have learned from homologous sequence alignment that the information that can be gained by comparing two genomes together is largely dependent upon the phylogenetic distance between them. Phylogenetic distance is a measure of the degree of separation between two organisms or their genomes on an evolutionary scale, usually expressed as the number of accumulated sequence changes, number of years, or number of generations. The distances are often placed on phylogenetic trees, which show the deduced relationships among the organisms (Figure 3). The more distantly related two organisms are, the less sequence similarity or shared genomic features will be detected between them. Thus, only general insights about classes of shared genes can be gathered by genomic comparisons at very long phylogenetic distances (e.g., over one billion years since their separation). Over such very large distances, the order of genes and the signatures of sequences that regulate their transcription are rarely conserved.

At closer phylogenetic distances (50–200 million years of divergence), both functional and non-functional DNA is found within the conserved segments. In these cases, the functional sequences will show signatures of selection by virtue of their sequences having changed less, or more slowly than, non-functional DNA. Moreover, beyond the ability to discriminate functional from non-functional DNA, comparative genomics is also contributing to the identification of general

classes of important DNA elements, such as coding exons of genes, non-coding RNAs, and some gene regulatory sites. In contrast, very similar genomes separated by about 5 million years of evolution (such as human and chimpanzee) are particularly useful for finding the sequence differences that may account for subtle differences in biological form. These are sequence changes under directional selection, a process whereby natural selection favors a single phenotype and continuously shifts the allele frequency in one direction. Comparative genomics is thus a powerful and promising approach to biological discovery that becomes more and more informative as genomic sequence data accumulate.

Practical- VISTA Tools for comparative genomics

Comparison of DNA sequences from different species is a fundamental method for identifying functional elements in genomes. VISTA is a comprehensive suite of programs and databases for comparative analysis of genomic sequences. There are two ways of using VISTA - you can submit your own sequences and alignments for analysis (VISTA servers) or examine pre-computed whole-genome alignments of different species.

VISTA tools are created to assist biologists in carrying out this task. First VISTA server at http://www-gsd.lbl.gov/vista/ was launched in the summer of 2000 and was designed to align long genomic sequences and visualize these alignments with associated functional annotations. Currently the VISTA site includes multiple comparative genomics tools and provides users with rich capabilities to browse pre-computed whole-genome alignments of large vertebrate genomes and other groups of organisms with VISTA Browser, to submit their own sequences of interest to several VISTA servers for various types of comparative analysis and to obtain detailed comparative analysis results for a set of cardiovascular genes. We illustrate capabilities of the VISTA site by the analysis of a 180 kb interval on human chromosome 5 that encodes for the kinesin family member 3A (*KIF3A*) protein.

Tools for comparative genomics

(i) mVISTA

Align and compare your sequences from multiple species mVISTA. mVISTA is designed to perform pairwise alignments of DNA sequences up to mega bases long from two or more species and to visualize these alignments together with annotations. AVID is the alignment engine behind mVISTA, and it allows the global alignment of DNA sequences of arbitrary length. To use mVISTA for comparative sequence analysis, two or more sequences in FASTA format (plain text only) or GenBank accession numbers together with a gene annotation file are submitted to the Web server.

(ii) rVISTA

rVISTA (regulatory VISTA) combines searching the major transcription factor binding site database TRANSFAC™ Professional from Biobase with a comparative sequence analysis. It can be used directly or through links in mVISTA, GenomeVISTA and VISTA Browser.The visualization program for rVISTA allows the user to look at binding sites for a single transcription factor and/or various combinations of transcription factor binding sites, which allows the user easily to examine the clustering of binding sites for factors that are believed to interact with one another. Both global (AVID) and local (BLASTZ) alignment algorithms are incorporated into rVISTA.

(iii) Genome VISTA

Compare your sequences with several whole genome assemblies. It will automatically find the ortholog, obtain the alignment and VISTA plot.

- **wgVISTA**

  Align sequences up to 10Mb long (finished or draft) including microbial whole-genome assemblies.

- **Phylo-Vista**

  The Phylo-VISTA program with its associated web server presents a novel method for the visualization and analysis of conservation in multiple sequence alignments by providing several significant extensions to VISTA tools. Analyze multiple DNA sequence alignments of sequences from different species while considering their phylogenic relationships.

Installation and Usage

VISTA is an application that allows the visualization of long sequence alignments with annotation information. The VISTA program uses the file or files (to visualize several related alignments) produced by any procedure of global or local alignment (such as BLAST, Gap (GCG), etc.) of two DNA sequences and parsed by the user according to 'Alignment_file' format. In a short time we are planning to release the stand-alone global alignment software to use as the first step before visualization.

The VISTA plot is based on moving a user-specified window over the entire alignment and calculating the percent identity over the window at each base pair. The X-axis represents the base sequence; the Y-axis represents the percent identity. If the user supplies an annotation file, genes and exons are marked above the plot. The direction of genes is indicated by an arrow, while the

coding exons and UTRs are marked with rectangles of different color. Conserved regions are highlighted under the curve, with red indicating a conserved non-coding region and blue indicating a conserved exon. Conserved UTRs are colored turquoise. The colors can be modified by the user. A conserved region is defined with percentage and length cutoffs. Conserved segments with percent identity X and length Y are defined to be regions in which every contiguous sub-segment of length Y was at least X% identical to its paired sequence. These segments are merged to define the conserved regions.

VISTA can be configured for visualizing alignments of various lengths by changing several parameters: the number of pages on which the output appears, the number of frames per page, the window size, and the resolution at which the alignment is plotted. VISTA allows one to easily create figures for various documents. For simplicity it is also possible to specify only a subset of these parameters, with the rest being automatically calculated. VISTA also supports simultaneous visualization of several related alignments.

(i) Installation:

Create a directory and copy Vista.jar and retepPDF2.jar to it. Then change your CLASSPATH environment variable to include references to these two files.

(ii) Example I (Windows):

1. mkdir c:\vista

2. copy source_path\Vista.jar c:\vista

3. copy source_path\retepPDF2.jar c:\vista

4. set CLASSPATH=c:\vista\Vista.jar;c:\vista\retepPDF2.jar

(iii) Example II (UNIX, csh/tcsh):

1. mkdir target_path/vista

2. cp source_path/Vista.jar target_path/vista

3. cp source_path/retepPDF2.jar target_path/vista

4. setenv CLASSPATH "target_path/vista/Vista.jar:target_path/vista/retepPDF2.jar"

(iv) Usage: java Vista [-options] plot_file

where "plot_file" is the name of a file containing plot parameters (for file format see Appendix III),

and options include:

-q turn on quiet mode

-d turn on debug mode

Further Reading

1. Stein LD, *et al.* (2003). "The genome sequence of Caenorhabditisbriggsae: a platform for comparative genomics". *PLoS Biology***1** (2): E45. doi:10.1371/journal.pbio.0000045. PMC 261899.PMID 14624247.
2. "Newly Sequenced Worm a Boon for Worm Biologists". *PLoS Biology***1** (2): e4–e4. 2003. doi:10.1371/journal.pbio.0000044.edit
3. Bejerano, Gill; Michael Pheasant, Igor Makunin, Stuart Stephen, W James Kent, John S Mattick, David Haussler (2004-05-28). "Ultraconserved elements in the human genome".*Science (New York, N.Y.)*304 (5675): 1321-1325. doi:10.1126/science.1098119. ISSN 1095-9203.Retrieved 2012-01-13.
4. Pollard, Katherine S.; Sofie R. Salama, Nelle Lambert, Marie-Alexandra Lambot, Sandra Coppens, Jakob S. Pedersen, Sol Katzman, Bryan King, Courtney Onodera, Adam Siepel, Andrew D. Kern, Colette Dehay, Haller Igel, Manuel Ares, Pierre Vanderhaeghen, David Haussler (2006). "An RNA gene expressed during cortical development evolved rapidly in humans". *Nature* 443 (7108): 167-172. doi:10.1038/nature05113. ISSN 0028-0836. Retrieved 2012-01-13.
5. Cristianini N and Hahn M (2006). *Introduction to Computational Genomics*.Cambridge University Press.ISBN 0-5216-7191-4.
6. Via, Allegra; Javier De Las Rivas, Teresa K. Attwood, David Landsman, Michelle D. Brazas, Jack A. M. Leunissen, Anna Tramontano, Maria Victoria Schneider (2011-10-27). "Ten Simple Rules for Developing a Short Bioinformatics Training Course".*PLoSComputBiol***7** (10): e1002245. doi:10.1371/journal.pcbi.1002245. Retrieved 2011-12-03.
7. Adams, M. D., Celniker, S. E. *et al*. The genome sequence of Drosophila melanogaster.*Science* 287, 2185-2195 (2000).
8. Aparicio, S., Chapman, J. *et al*, Whole-genome shotgun assembly and analysis of the genome of Fugurubripes.*Science* 297, 1301-1310 (2002).
9. Blattner, F. R., Plunkett, G. *et al*.The complete genome sequence of Escherichia coli K-12.*Science* 277, 1453-1462 (1997).
10. Blattner, F. R., Plunkett, G. *et al.* Sequence and comparative analysis of the chicken genome provide unique perspectives on vertebrate evolution. *Nature* 432, 695-716 (2004).

Application Note

Application note on *de novo* assembly of a plant transcriptome using Illumina paired-end sequencing data from *Lactuca serriola* (prickly lettuce) in CLC Genomics Workbench

Uday Deshpande

De novo assembly of paired-end plant transcriptome data

CLC bio's de novo *assembly tool allows for initiation of genomic data analysis in organisms without previously sequenced genomes. This is especially important in plant and animal genome research where the majority of species do not have reference sequences available. The* de novo *assembler was tested over the years by many plant and animal researchers who demonstrated that it successfully assembled short sequencing reads into long contigs. This application note provides some guidelines for de novo assembly of transcriptome high-throughput sequencing reads using CLC Genomics Workbench.*

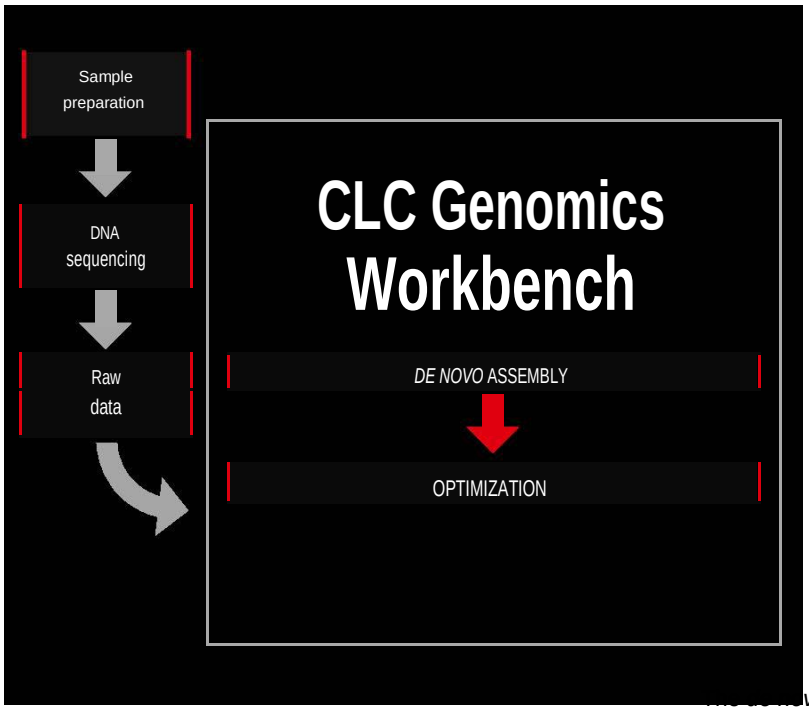


Figure 1: Workflow for *de novo* assembly using CLC Genomics Workbench

Reads dataset

This application note uses the publically available reads from *Lactuca serriola* (prickly lettuce) mRNA library[1,2]. The library was prepared using the Illumina mRNA-Seq library protocol followed by DSN (Duplex Specific Nuclease) nor-malization. The data set consists of around 93 Million (46.5M X 2) paired end reads (85 nt) from Illumina Genome Analyzer IIx.

[1]mRNA library submitted to GenBank SRA (Short Read Archive) by University of California – Davis researchers. Compositae Genomics Project http://compgenomics. ucdavis.edu

[2]Data is available at http://www.ncbi.nlm.nih.gov/sra/SRX098217

Software: CLC Genomics Workbench 4.9

Analysis workflow

De novo Assembly

The *de novo* assembly algorithm of CLC Genomics Workbench supports hybrid data, meaning you can perform assemblies that use both short and long reads, paired reads as well as reads from different sequencing technologies: all these can be entered into the same assembly job.

The *de novo* assembly algorithm is divided into two phas-es. In the first phase, simple contig sequences are created by using all the information that is in the read sequences. This is the actual *de novo* assembly part of the process, however, these contig sequences do not contain any information about which reads were used to create the contigs. In the second phase, all reads are mapped back to the simple contig sequences. This is done to show e.g. coverage levels along the contigs and enables more downstream analysis like variant detection. When starting the assembly you can choose whether you want to output simple contigs only, or have the reads mapped back, too. For reporting, you can choose a summary report on the assembly and a sequence list of the unmapped reads.

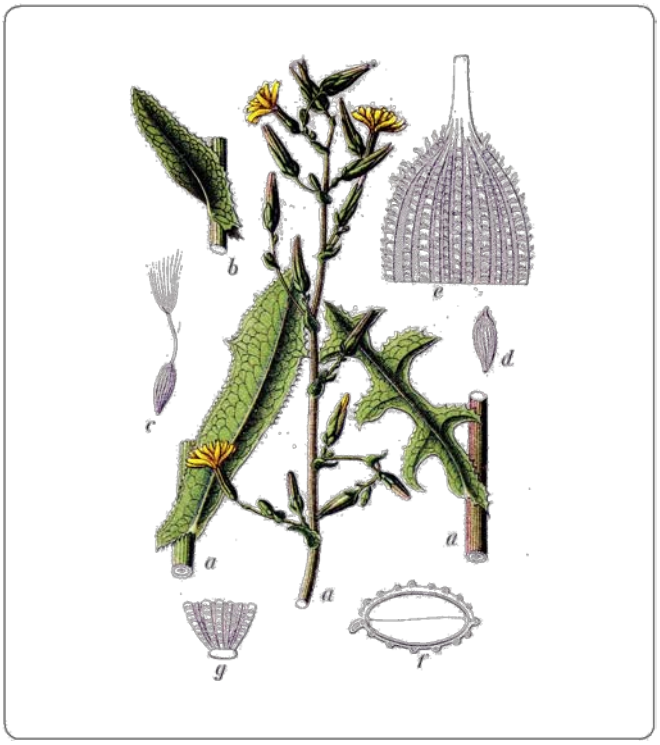


Figure 2: *Lactuca serriola*

Word size	24	42	48	54	60	64
% reads matched	95.4	95.3	94.8	94.1	93.9	92.3
# contigs, x1000	66.9	54.8	50.7	46.5	42.5	39.9
Average contig length	919	954	959	966	959	940
Assembly length in Mb	61.5	52.3	48.7	44.9	40.7	37.5

Table 1: Effect of word size on assembly outcome

Parameters have to be specified for each set of reads. Paired information is used for both the first and the second phase of the assembly but the other parameters relate only to the second phase where the reads are mapped back to the contig sequences.

The parameter settings include penalty scores for mismatch, insertion and deletion costs, length fraction and similarity. For the data set used here, we selected the heaviest penalty for mismatch, insertion, and deletion costs which is 3. Length fraction was set to 0.5 and similarity was set to 0.95. This means that at least 50% of the individual read needs to have at least 95% identity with the contig sequence in order to be included in the mapping.

You can also specify the minimum contig length. We set the minimum contig length to 300 nt so contigs shorter than this will not be reported.

Furthermore, you can specify how to deal with non-specific reads and report-ing of conflicts.

Finally, our *de novo* assembler allows the user to either manually set the word size or it can be automatically chosen by the algorithm. For the first *de novo* assembly run we selected to use the automatic word size.

Results and optimization

The assembler selected the word size of 24. The input was around 81 million reads and the number of resulting contigs was 66.9K with an average contig length of 919 nt.

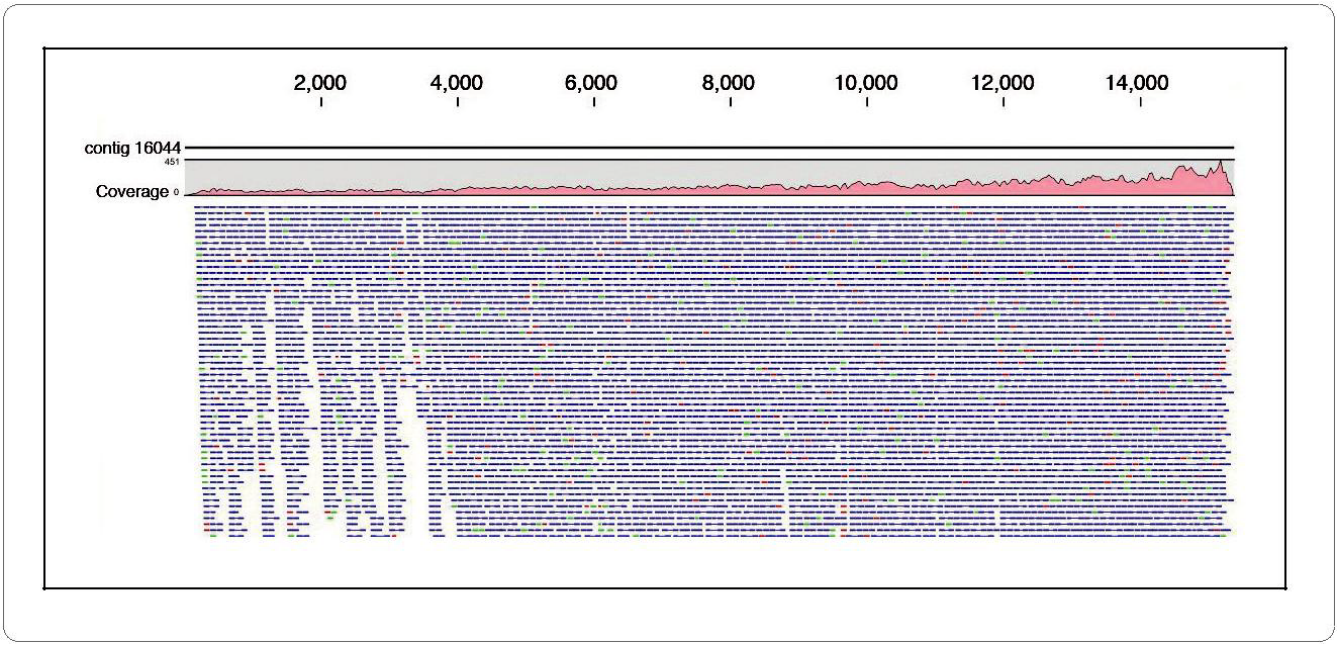


Figure 3: The longest assembled contig with 15,368 nt in the read mapping view of CLC Genomics Workbench. This contig represents the longest plant transcript, a homolog of the *Arabidopsis* auxin transport protein (BIG) gene.

Optimization of word size

To optimize the assembly output you can change the parameters, e.g. the word size.

In our experiment, we repeated the *de novo* assembly on the same data set with different word sizes as listed in Table 1. The table shows that the word size of 54 produced the longest (on average) contigs and a graphical repre-sentation is shown in Figure 4.

Performing a *de novo* assembly with word size 54, we obtained 46.5K contigs with the final assembly length 45 Mb. The average size of the contigs was 966 nt and in total 94.1% of the reads were mapped.

It is difficult to provide any general recommendations on the optimal word size but for high quality reads the word size should generally be higher than for low quality reads.

To further improve the assembly, other parameters such as the alignment stringency can also be adjusted.

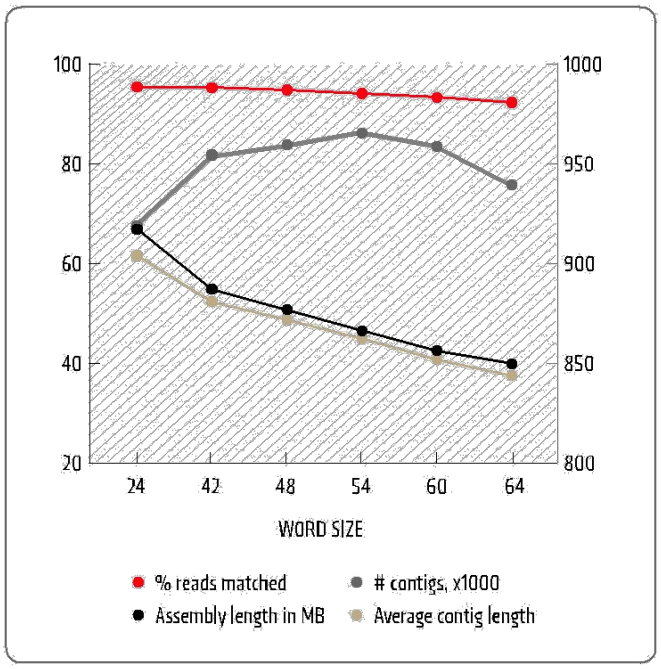


Figure. 4: The results from six assemblies of *Lactuca serriola* reads using different word sizes

16.05.2013 CLC bio

CLC bio · EMEA
Silkeborgvej 2 · Prismet
8000 Aarhus C
Denmark
Phone: +45 7022 5509

CLC bio · Americas
10 Rogers St # 101
Cambridge · MA 02142
USA
Phone: +1 (617) 945 0178

CLC bio · AsiaPac
69 · Lane 77 · Xin Ai Road · 7th fl.
Neihu District · Taipei · Taiwan 114
Taiwan
Phone: +886 2 2790 0799

APPENDIX I

The targeted sequence-enrichment technologies for NGS

Technology	Approach	Platform	Website
Agilent/SureSelect	Array- and solution-based	Illumina/Roche/ABI	http://www.chem.agilent.com/
RainDance	Microdroplet-based	Illumina/Roche/ABI	http://www.raindancetechnologies.com/
NimbleGen/SeqCap/EZ Exome	Array- and solution-based	Illumina/Roche/ABI	http://www.nimblegen.com/products/seqcap/index.html
Febit/HybSelect	Microarray-based	Illumina/Roche/ABI	http://www.febit.com/microarray-sequencing/index.cfm
Fluidigm	PCR-based	Illumina/Roche/ABI	http://www.fluidigm.com/targeted-resequencing.html
Mycroarray/Myselect	Solution-based	Illumina/Roche/ABI	http://www.mycroarray.com/products/myselect.html
LC Sciences	Microarray-based	Illumina/Roche/ABI	http://www.lcsciences.com/applications/genomics/
Qiagen/SeqTarget	Long-range PCR-based	Illumina/Roche/ABI	http://www.qiagen.com/products/seqtargetsystem.aspx
Illumina/TruSeq	Solution-based	Illumina/Roche/ABI	http://www.illumina.com/applications.ilmn

The alignment, assembly and utility bioinformatics tools for NGS.

Program	Function	Platform	Website
De novo assembly			
Abyss	Alignment/assembly	Illumina	http://www.bcgsc.ca/platform/bioinfo/software/abyss
ALLPATHS	Alignment/assembly	Illumina	http://www.broadinstitute.org/science/programs/genome-biology/crd
AMOScmp	Alignment/assembly	Roche	http://sourceforge.net/projects/amos/files/
ARACHNE	Alignment/assembly	Roche	http://www.broadinstitute.org/science/programs/genome-biology/crd
CAP3	Alignment/assembly	Roche	http://pbil.univ-lyon1.fr/cap3.php
Consensus/SeqCons	Alignment/assembly	Roche	http://www.seqan.de/downloads/projects.html
Curtain	Alignment/assembly	Illumina/Roche/ABI	http://code.google.com/p/curtain/
Edena	Alignment/assembly	Illumina	http://www.genomic.ch/edena
Euler-SR	Alignment/assembly	Illumina/Roche	http://euler-assembler.ucsd.edu/portal/?q=team
FuzzyPath	Alignment/assembly	Illumina/Roche	ftp://ftp.sanger.ac.uk/pub/zn1/fuzzypath/fuzzypath_v3.0.tgz
IDBA	Alignment/assembly	Illumina	http://www.cs.hku.hk/ẅalse/idba/
MIRA/MIRA3	Alignment/assembly	Illumina/Roche	http://chevreux.org/projects_mira.html
Newbler	Alignment/assembly	Roche	roche-applied-science.com/
Phrap	Alignment/assembly	Illumina/Roche	http://www.phrap.org/consed/consed.html#howToGet
RGA	Alignment/assembly	Illumina	http://rga.cgrb.oregonstate.edu/
QSRA	Alignment/assembly	Illumina	http://qsra.cgrb.oregonstate.edu/
SHARCGS	Alignment/assembly	Illumina	http://sharcgs.molgen.mpg.de/
SHORTY	Alignment/assembly	ABI	http://www.cs.sunysb.edu/ẅskiena/shorty/
SHRAP	Alignment/assembly	Roche Illumina	By request http://soap.genomics.org.cn
SOAPdenovo	Alignment/assembly	Illumina/ABI	http://www.physics.rutgers.edu/%7Eanirvans/SOPRA/
SOPRA	Alignment/assembly	Roche	http://bioserver.cs.put.poznan.pl/sr-asm-short-reads-assembly-algorithm
SR-ASM	Alignment/assembly	Illumina/Roche	http://www.bcgsc.ca/platform/bioinfo/software/ssake
SSAKE	Alignment/assembly	Illumina	http://sourceforge.net/projects/taipan/files/
Taipan	Alignment/assembly	Illumina/Roche	http://sourceforge.net/projects/vcake
VCAKE	Alignment/assembly	Illumina/Roche/ABI	http://www.ebi.ac.uk/%7Ezerbino/velvet
Velvet	Alignment/assembly		
Reference-based assembly			
BFAST	Alignment/assembly	Illumina/ABI	http://sourceforge.net/apps/mediawiki/bfast/index.php?title=Main_Page

Bowtie	Alignment/assembly	Illumina/Roche/ABI	http://bowtie-bio.sourceforge.net
BWA CoronaLite	Alignment/assembly	Illumina/ABI	http://bio-bwa.sourceforge.net/bwa.shtml
CABOG	Alignment/assembly	ABI	http://solidsoftwaretools.com/gf/project/corona/
ELAND/ELAND2	Alignment/assembly	Roche/ABI	http://wgs-assembler.sf.net
EULER	Alignment/assembly	Illumina/ABI	http://www.illumina.com/
Exonerate	Alignment/assembly	Illumina	http://euler-assembler.ucsd.edu/portal/
EMBF	Alignment/assembly	Roche	http://www.ebi.ac.uk/~guy/exonerate
GenomeMapper	Alignment/assembly	Illumina	http://www.biomedcentral.com/1471-2105/10?issue=S1
GMAP	Alignment/assembly	Illumina	http://1001genomes.org/downloads/genomemapper.html
gnumap	Alignment/assembly	Illumina	http://www.gene.com/share/gmap
ICON	Alignment/assembly	Illumina	http://dna.cs.byu.edu/gnumap/
Karma	Alignment/assembly	Illumina	http://icorn.sourceforge.net/
LAST	Alignment/assembly	Illumina/ABI	http://www.sph.umich.edu/csg/pha/karma/
LOCAS	Alignment/assembly	Illumina	http://last.cbrc.jp/
Mapreads	Alignment/assembly	Illumina	http://www-ab.informatik.uni-tuebingen.de/software/locas
MAQ	Alignment/assembly	ABI	http://solidsoftwaretools.com/gf/project/mapreads/
MOM	Alignment/assembly	Illumina/ABI	http://maq.sourceforge.net
Mosaik	Alignment/assembly	Illumina	http://mom.csbc.vcu.edu/
mrFAST/mrsFAST	Alignment/assembly	Illumina/Roche/ABI	http://bioinformatics.bc.edu/marthlab/Mosaik
MUMer	Alignment/assembly	Illumina	http://mrfast.sourceforge.net/
Nexalign	Alignment/assembly	ABI Illumina	http://mummer.sourceforge.net/
Novocraft	Alignment/assembly	Illumina	http://genome.gsc.riken.jp/osc/english/dataresource/
PerM	Alignment/assembly	Illumina/ABI	http://www.novocraft.com/
RazerS	Alignment/assembly	Illumina/ABI	http://code.google.com/p/perm/
RMAP	Alignment/assembly	Illumina	http://www.seqan.de/projects/razers.html
segemehl	Alignment/assembly	Illumina/Roche	http://rulai.cshl.edu/rmap
SeqCons	Alignment/assembly	Roche	http://www.bioinf.uni-leipzig.de/Software/segemehl/
SeqMap	Alignment/assembly	Illumina	http://www.seqan.de/projects/seqcons.html
SHRiMP	Alignment/assembly	Illumina/Roche/ABI	http://biogibbs.stanford.edu/*jiangh/SeqMap/
Slider/SliderII	Alignment/assembly	Illumina	http://compbio.cs.toronto.edu/shrimp
SOCS	Alignment/assembly	ABI	http://www.bcgsc.ca/platform/bioinfo/software/slider
SOAP/SOAP2	Alignment/assembly	Illumina/ABI	http://solidsoftwaretools.com/gf/project/socs/
	Alignment/assembly		http://soap.genomics.org.cn

(continued on next page)

Table 3 (continued)

Program	Function	Platform	Website
SSAHA/SSAHA2	Alignment/assembly	Illumina/Roche	http://www.sanger.ac.uk/Software/analysis/SSAHA2
Stampy	Alignment/assembly	Illumina	http://www.well.ox.ac.uk/wmarting/
SXOligoSearch	Alignment/assembly	Illumina	http://synasite.mgrc.com.my:8080/sxog/NewSXOligoSearch.php
SHORE	Alignment/assembly	Illumina	http://1001genomes.org/downloads/shore.html
Vmatch	Alignment/assembly	Illumina	http://www.vmatch.de/
Diagnostics/utilities			
Artemis/ACT	Visualization tool	Illumina/Roche	http://www.sanger.ac.uk/resources/software/artemis/
CASHX	Pipeline	Illumina	http://seqanswers.com/wiki/CASHX
Consed	Visualization tool	Illumina/Roche	http://www.genome.washington.edu/consed/consed.html
EagleView	Visualization tool	Illumina/Roche	http://bioinformatics.bc.edu/marthlab/EagleView
FastQC	Quality assessment	Illumina/ABI	http://www.bioinformatics.bbsrc.ac.uk/projects/fastqc/
Gambit	Visualization tool	Illumina/Roche	http://bioinformatics.bc.edu/marthlab/Gambit
Goby	Data management	Illumina/Roche/ABI	http://campagnelab.org/software/goby/
G-SQZ	Data management	Illumina/ABI	http://public.tgen.org/sqz
Hawkeye	Visualization tool	Illumina/Roche	http://amos.sourceforge.net/hawkeye
Hybrid-SHREC	Error Correction	Illumina/Roche/ABI	http://www.cs.helsinki.fi/u/lmsalmel/hybrid-shrec/
IGV	Visualization tool	Illumina	http://www.broadinstitute.org/igv/?q¼home
LookSeq	Visualization tool	Illumina/Roche	http://lookseq.sourceforge.net
MagicViewer	Visualization tool	Illumina	http://bioinformatics.zj.cn/magicviewer/
MapView	Visualization tool	Illumina	http://evolution.sysu.edu.cn/mapview/
NGSView	Visualization tool	Illumina/ABI	http://ngsview.sourceforge.net
PIQA	Quality assessment	Illumina	http://bioinfo.uh.edu/PIQA
Reconciliation	Assembly pipeline	Illumina	http://www.genome.umd.edu/software.htm
RefCov	Sequence coverage	Illumina/Roche	http://genome.wustl.edu/tools/cancer-genomics
SAM Tools	Utilities	Illumina/Roche	http://sourceforge.net/projects/samtools/files/
Savant	Visualization tool	Illumina/Roche	http://compbio.cs.toronto.edu/savant/
ShortRead	Quality assessment	Illumina/Roche	http://bioconductor.org/packages/2.6/bioc/html/ShortRead.html
SHREC	Error Correction	Illumina/Roche	http://www.informatik.uni-kiel.de/jasc/Shrec/
Staden Tools (GAP5)	Pipeline	Illumina/Roche	http://sourceforge.net/projects/staden/files/
Tablet	Visualization tool	Illumina/Roche	http://bioinf.scri.ac.uk/tablet
TagDust	Data cleaning	Illumina	http://genome.gsc.riken.jp/osc/english/software/.
TileQC	Quality assessment	Illumina	http://www.science.oregonstate.edu/wdolanp/tileqc
XMatchView	Visualization tool	Illumina/Roche	http://www.bcgsc.ca/platform/bioinfo/software/xmatchview
Yenta	Visualization tool	Illumina	http://genome.wustl.edu/tools/cancer-genomics
Geneus	Data management	Illumina/ABI	http://www.genologics.com/solutions/research-informatics/

The genetic variant prediction and detection bioinformatic programs for NGS data analysis.

Variant prediction/detection	Platform	Website
Functional variant prediction		
B-SIFT		http://research-pub.gene.com/bsift/
MAPP		http://mendel.stanford.edu/supplementarydata/stone_MAPP_2005
PhD-SNP		http://gpcr.biocomp.unibo.it/wemidio/PhD-SNP/PhD-SNP
PolyPhen-2/PolyPhen		http://genetics.bwh.harvard.edu/pph2/
SIFT SNAP		http://blocks.fhcrc.org/sift/SIFT.html
SNAPper/Pedant		http://www.rostlab.org/services/SNAP
		http://pedant.gsf.de/snapper
Variant detection		
Structural/genomic variant		
BreakDancer	Roche/Illumina/ABI	http://genome.wustl.edu/tools/cancer-genomics/

BreakDancer/BD- Mini	Roche/Illumina/ABI	http://seqanswers.com/wiki/BreakDancer
Breakway	Roche/Illumina/ABI	http://sourceforge.net/projects/breakway/files/
CNVSeq	Roche	http://tiger.dbs.nus.edu.sg/CNV-seq/
cnvHMM	Illumina	http://genome.wustl.edu/pub/software/cancer-genomics/cnvHMM/
cnD	Illumina	http://www.sanger.ac.uk/resources/software/cnd.html
GASV/GSV	Illumina	http://cs.brown.edu/people/braphael/software.html
Hydra	Illumina	http://code.google.com/p/hydra-sv/
MoDIL	Illumina	http://compbio.cs.toronto.edu/modil/
mrFAST	Illumina	http://mrfast.sourceforge.net/
NovelSeq	Roche/Illumina/ABI	http://compbio.cs.sfu.ca/strvar.htm
PEMer	Roche/Illumina/ABI	http://sv.gersteinlab.org/pemer/
Pindel	Illumina	http://www.ebi.ac.uk/ẁkye/pindel/
SegSeq	Illumina/ABI	http://www.broadinstitute.org/
SOAPsv	Roche/Illumina/ABI	http://soap.genomics.org.cn
Solid large Indel tool	ABI	http://solidsoftwaretools.com/gf/project/large_indel/
Solid CNV tool SWT	ABI	http://solidsoftwaretools.com/gf/project/cnv/
VariationHunter/VH-CR	Illumina	http://genome.wustl.edu/pub/software/cancer-genomics/GSTAT/
VARiD	Illumina	http://compbio.cs.sfu.ca/strvar.html
	ABI	http://compbio.cs.utoronto.ca/varid
Single nucleotide variant		
Atlas-SNP2	Roche/Illumina	http://www.hgsc.bcm.tmc.edu/cascade-tech-software-ti.hgsc
BOAT	Illumina	http://boat.cbi.pku.edu.cn/
DNA Baser	Roche	http://www.dnabaser.com/help/manual.html
DNAA	Roche/Illumina/ABI	http://sourceforge.net/projects/dnaa/
Galign	Illumina	http://shahamlab.rockefeller.edu/galign/galign.htm
GigaBayes/PbShort	Roche/Illumina	http://bioinformatics.bc.edu/marthlab/GigaBayes
GSNAP inGAP	Roche/Illumina	http://share.gene.com/gmap.
ngs_backbone	Roche/Illumina	http://sites.google.com/site/nextgengenomics/ingap
Omixon Variant	Roche/Illumina	http://bioinf.comav.upv.es/ngs_backbone/index.html
PyroBayes	ABI Roche	http://www.omixon.com/omixon/index.html
ssahaSNP	Illumina/Roche	http://bioinformatics.bc.edu/marthlab/PyroBayes
Slider	Illumina	http://www.sanger.ac.uk/Software/analysis/ssahaSNP
SNP-o-matic	Illumina	http://www.bcgsc.ca/platform/bioinfo/software/slider
SNPSeeker	Illumina	http://snpomatic.sourceforge.net
SNVMix	Illumina	http://www.genetics.wustl.edu/rmlab/
SOAPsnp	Roche/Illumina/ABI	http://compbio.bccrc.ca
SWA454	Roche	http://soap.genomics.org.cn
SVA	Illumina Illumina	http://www.broadinstitute.org/science/programs/genome-biology/crd
VAAL	Roche/Illumina	http://www.svaproject.org/
VarScan	Roche/Illumina/ABI	http://www.broadinstitute.org/science/programs/genome-biology/crd
VARiD		http://genome.wustl.edu/tools/cancer-genomics
		http://compbio.cs.utoronto.ca/varid
Differences between genomes		
DIAL	Illumina	http://www.bx.psu.edu/miller_lab/
SomaticCall	Illumina	http://www.broadinstitute.org/science/programs/genome-biology/crd
SWAP454	Roche	http://www.broadinstitute.org/science/programs/genome-biology/crd
VAAL	Illumina	http://www.broadinstitute.org/science/programs/genome-biology/crd

SOFTWARE/TOOLS FOR WORKSHOP

QA/QC:

1. FastQC
2. FastxToolkit
3. FastScreen
4. Cutadapt

Reference mapping/alignment:

1. Bowtie2

Alignment processing:

1. Samtools
2. Bcftools

Alignment QC:

1. Qualimap

De novo assembly and reference guided assembly:

1. Velvet

Gap filling using iterative approach:

1. IMAGE

Scaffolding:

1. SSPACE

Other scripts:

1. *cmpfastq_pe.pl*
2. *countFasta.pl*

I. QUALITY CONTROL

TOOL 1: FastQC

Task 1: Open FastQC

/FastQC/fastqc

Task 2: Upload read files

File > Open = Upload read file in fastq format

Task 3: Do the quality checks

TOOL 2: FASTX TOOLKIT

Task 1: Generate quality score statistics

fastx_quality_stats -Q33 -i <r1.fq> -o <r1.stats>
fastx_quality_stats -Q33 -i <r2.fq> -o <r2.stats>

Task 2: Generate quality score boxplot

fastq_quality_boxplot_graph.sh -i <r1.stats> -o <r1.png>
fastq_quality_boxplot_graph.sh -i <r2.stats> -o <r2.png>

Task 3: Trimming reads

fastx_trimmer -Q33 -f <int> -l <int> -i <read.fq> -o <t_read.fq>

or

fastx_trimmer -Q33 -t <int> -i <read.fq> -o <t_read.fq>

Task 4: Quality filtering

fastq_quality_filter -Q33 -v -q <int> -p <int> -i <r1.fq> -o <f_r1.fq>
fastq_quality_filter -Q33 -v -q <int> -p <int> -i <r2.fq> -o <f_r2.fq>

TOOL 3: Fastq Screen

Task 1: Configure database

Edit the “fastq_screen.conf” in order to configure database required to run fastq screen.

Task 2: Create bowtie 2 index

bowtie2-build ref.fa ref_bowtie2

Example format:

```
############################
DATABASE Bacteria_Bowtie2 /fastq_screen/ref_bowtie2 BOWTIE2
############################
```

Task 3: Download the PhiX data

Download the PhiX data and configure it in “fastq_screen.conf”

Task 4: Run the screening

Single-end reads

fastq_screen --aligner bowtie2 reads.fastq

Paired-end reads

fastq_screen --aligner bowtie2 --paired r1.fq r2.fq

TOOL 4: Cutadapt

Task 1: Discard sequencing adapters

cutadapt -b <adapter sequence> --discard-trimmed r1.fq > adp_r1.fq

cutadapt -b <adapter sequence> --discard-trimmed r2.fq > adp_r2.fq

II. REFERENCE MAPPING/ALIGNMENT

TOOL 5: BOWTIE 2

Task 1: Creating reference index

Build bowtie index from a set of DNA sequences. Bowtie2-build outputs a set of 6 files with suffixes .1.bt2, .2.bt2, .3.bt2, .4.bt2, .rev.1.bt2, and .rev.2.bt2. These files together constitute the index: they are all that is needed to align reads to that reference. The original sequence FASTA files are no longer used by Bowtie 2 once the index is built.

Command:

bowtie2-build [options] <reference_sequence> <index_name>*

Example:

bowtie2-build ref.fa ref_bowtie2

Task 2:

Aligning reads against reference sequence

Command:

bowtie2 -x <reference index> -p <number of processors to be used> -S (generate output in SAM format) -1 <read_1> -2 <read_2> <output.sam>

Example:

bowtie2 -x ref_bowtie2 -p 10 -S -1 r1.fq -2 r2.fq aln.sam

TOOL 6: SAMTOOL

Samtools is a set of utilities that manipulate alignments in the BAM format. It imports from and exports to the SAM (Sequence Alignment/Map) format, does sorting, merging and indexing.

Task 3:

Index reference sequence in the FASTA format. faidx will index the file and create <ref.fasta>.fai on the disk.

Command:

samtools faidx ref.fa

Task 4:

Import SAM to BAM

Command:

samtools view -bS aln.sam > aln.bam

-b: Output in the BAM format.
-s: Input in the SAM format.

Task 5:

Sort alignments by leftmost coordinates. File <out.prefix>.bam will be created. This command may also create temporary files <out.prefix>.%d.bam when the whole alignment cannot be fitted into memory

Command:

samtools sort aln.bam aln.sorted

Task 6:

Index sorted alignment for fast random access. Index file <aln.bam>.bai will be created.

Command:

samtools index aln.sorted.bam

Task 7:

Remove potential PCR duplicates

Command:

samtools rmdup -S aln.sorted.bam rmdup.bam

-S : Treat paired-end reads and single-end reads

Task 8:

Sort alignments by leftmost coordinates. File <out.prefix>.bam will be created. This command may also create temporary files <out.prefix>.%d.bam when the whole alignment cannot be fitted into memory

Command:

samtools sort rmdup.bam rmdup_ aln.sorted

Task 9:

Index sorted alignment for fast random access. Index file <aln.bam>.bai will be created.

Command:

samtools index rmdup_aln.sorted.bam

III. VARIANT CALLING

TOOL 7: BCFTOOLS

Task 10:

Generate BCF or pileup for one or multiple BAM files.

Command:

samtools mpileup -ugf ref.fa rmdup_aln.sorted.bam | bcftools view -bvcg - > var.raw.bcf

-u Generate uncompress BCF output
-g Generate BCF output
-f Faidx indexed reference sequence file

Task 11:

Call SNPs and short INDELs from BCF files

Command:

bcftools view var.raw.bcf | vcfutils.pl varFilter -D 100 > 100_var.flt.vcf

IV. ALIGNMENT QC

TOOL 8: Qualimap

Task 1: Run “bamqc” in qualimap

qualimap_v0.7.1/qualimap bamqc -bam aln.sorted.bam -c -outformat PDF

-bam input mapping file
-c paint chromosome limits inside charts

QC on alignment file is very essential for checking genome coverage, insert size distribution and duplication level.

Run Qualimap before and after duplicate removal and check if there is any major changes in the alignment file.

V. WHOLE GENOME DE NOVO ASSEMBLY

TOOL 9: VELVET

Task1: Velveth

Velveth takes in a number of sequence files, produces a hashtable, then outputs two files in an output directory, Sequences and Roadmaps, which are necessary to velvetg.

Command:

velveth output_directory kmer_length [-read_type] [-read_file_type] [-file_format] read_1 read_2

Example:

velveth out_31 31 -shortPaired -separate -fastq r1.fq r2.fq

Task 2: Velvetg

Velvetg is the core of velvet where the de Bruijn graph is built

Command:

velvetg output_directory -cov_cutoff auto -exp_cov auto

Note: Repeating the task 1 and 2 with multiple kmers helps to obtain optimum kmer value. Document the following assembly statistics for each kmers.

1. Number of contigs
2. N50 value
3. Largest contig
4. Consensus/Total size
5. Reads used for assembly

Task 3: Observe insert length

Command:

perl /usr/local/bin/observed-insert-length.pl <Velvet_out_directory>

Example:

perl /usr/local/bin/observed-insert-length.pl 31_assembly

Note: The observed insertion length information will be displayed on the screen

Task 4: Final velvetg run with all required parameters

Command:

velvetg <output directory> -cov_cutoff <INT> -exp_cov <INT> -ins_length <INT> -ins_length_sd <INT>

Example:

velvetg sorted_out -cov_cutoff 8 -exp_cov 16 -ins_length 310 -ins_length_sd 88

VI. REFERENCE GUIDED ASSEMBLY

Task 1: Extract properly paired reads

perl cmpfastq_pe.pl r1.fq r2.fq

cat r1.fq-unique.out r2.fq-unique.out > singletons.fq

Task 2: Align both paired and singleton reads against reference

bowtie2-build ref.fa ref_bowtie2

bowtie2 -x ref_bowtie2 -p 20 -S -1 r1.fq.cmn.out -2 r2.fq.cmn.out paired.sam

bowtie2 -x ref_bowtie2 -p 20 -S -U singletons.fq singletons.sam

Task 3. Sort BAM files

sort paired.sam > sorted_paired.sam

sort singletons.sam > sorted_singletons.sam

Task 3. Reference guided assembly using velvet

velveth ref_ass_31 31 -reference -fasta mg_super.fa -shortPaired -sam

sorted_paired.sam -short2 -sam sorted_singletons.sam

velvetg ref_ass_31 -cov_cutoff auto -exp_cov auto

perl observed-insert-length.pl ref_ass_31

Upon running this script, suggested parameter for velvetg will be displayed on the terminal

velvetg ref_ass_31 -cov_cutoff auto -exp_cov auto inst_length -ins_length_sd

VII. ITERATIVE MAPPING AND ASSEMBLY

TOOL 10: IMAGE

Task 1: Closing gaps using iterative approach

Create a folder “image”

Copy “contigs.fa” into this folder

rename it “scafflods.fa”

Copy only properly paired reads into this directory

cp r1.fq.cmn.out and r2.fq.cmn.out to “image” folder

Now run

IMAGE_version2/image.pl -kmer 31 -vel_ins_len 350 -scaffolds scaffolds.fa -prefix image -iteration 1 -all_iteration 10 -dir_prefix ite

Once it completes you can see folders named ite1, ite2 .... ite10.

Inside “ite10” you can see a file called “image.contigs.fa” . This is the final contigs from gap filling assignment

VIII. SCAFFOLDING

TOOL 11: SSPACE

Task 1: Prepare library information

Create a folder called “scaffolding”

cp image.contigs.fa into this directory

create a file named “libraries.txt” and copy the following line into this file

lib1 r1.fq-common.out r2.fq-common.out 350 0.75 FR

Task 2. Scaffolding

perl SSPACE-BASIC-2.0_linux-x86_64/SSPACE_Basic_v2.0.pl -l libraries.txt -s image.contigs.fa -T 8 -v 1 -p 1 -b sspace

Once it completes you can see a file named “sspace.final.scaffolds.fasta”. This is your final assembled scaffold sequences.

Task 3: Generate assembly statistics

perl countFasta.pl sspace.final.scaffolds.fasta

This will create following statistics (Example)

Total length of sequence:	41574181 bp
Total number of sequences:	2508
N25 stats:	25% of total sequence length is contained in the 45 sequences >= 162575 bp
N50 stats:	50% of total sequence length is contained in the 139 sequences >= 77328 bp
N75 stats:	75% of total sequence length is contained in the 375 sequences >= 25630 bp
Total GC count:	21266020 bp
GC %:	51.15 %