Next-Generation Sequencing Data Analysis

Next-Generation Sequencing Data Analysis

Xinkun Wang
Northwestern University
Chicago, Illinois, USA

CRC Press
Taylor & Francis Group
Boca Raton London New York

CRC Press is an imprint of the
Taylor & Francis Group, an **informa** business
A CHAPMAN & HALL BOOK

CRC Press
Taylor & Francis Group
6000 Broken Sound Parkway NW, Suite 300
Boca Raton, FL 33487-2742

Version Date: 20160127

International Standard Book Number-13: 978-1-4822-1789-6 (eBook - PDF)

Contents

Section I Introduction to Cellular and Molecular Biology

Section III Application-Specific NGS Data Analysis

Section I

Introduction to Cellular and Molecular Biology

1

The Cellular System and the Code of Life

1.1 The Cellular Challenge

A cell, although minuscule with a diameter of less than 50 μm, works wonders if you compare it to any human-made system. Moreover, it perpetuates itself using the information coded in its DNA. In case you ever had the thought of designing an artificial system that shows this type of sophistication, you would know the many insurmountable challenges such a system needs to overcome. A cell has a complicated internal system, containing many types of molecules and parts. To sustain the system, a cell needs to perform a wide variety of tasks—the most fundamental of which are to maintain its internal order, prevent its system from malfunctioning or breaking down, and reproduce or even improve the system—in an environment that is constantly changing.

Energy is needed to maintain the internal order of the cellular system. Without constant energy input, the entropy of the system will gradually increase, as dictated by the second law of thermodynamics, and ultimately lead to the destruction of the system. Besides energy, raw "building" material is also constantly needed to renew its internal parts or build new ones, as the internal structure of a cell is dynamic and responds to constant changes in environmental conditions. Therefore, to maintain the equilibrium inside and with the environment, it requires a constant influx of energy and raw material, and excretion of its waste. Guiding the capture of the requisite energy and raw material for its survival and the perpetuation of the system is the information encoded in its DNA sequence.

Because of evolution, a great number of organisms no longer function as a single cell. The human body, for example, contains trillions of cells. In a multicellular system, each cell becomes specialized to perform a specific function, for example, β-cells in our pancreas synthesize and release insulin, and cortical neurons in the brain perform neurobiological functions that underlie learning and memory. Despite this division of labor, the challenges a single-cell organism faces still hold true for each one of these cells. Instead of dealing with the external environment directly, they interact with and respond to changes in their microenvironment.

1.2 How Cells Meet the Challenge

Many cells, like algae and plant cells, directly capture energy from the sun or other energy sources. Other cells (or organisms) obtain energy from the environment as heterotrophs. For raw material, cells can either fix carbon dioxide in the air using the energy captured into simple organic compounds, which are then converted to other requisite molecules, or directly obtain organic molecules from the environment and convert them to requisite materials. In the meantime, existing cellular components can also be broken down when not needed for the reuse of their building material. This process of energy capture and utilization, and synthesis, interconversion, and breaking down for reuse of molecular material, constitutes the cellular metabolism. Metabolism, the most fundamental characteristic of a cell, involves numerous biochemical reactions.

Reception and transduction of various signals in the environment are crucial for cellular survival. Reception of signals relies on specific receptors situated on the cell surface, and for some signals, those inside the cell. Transduction of incoming signals usually involves cascades of events in the cell, through which the original signals are amplified and modulated. In response, the cellular metabolic profile is altered. The cellular signal reception and transduction network is composed of circuits that are organized into various pathways. Malfunctioning of these pathways can have a detrimental effect on the cell's response to the environment and eventually its survival.

Perpetuation and evolution of the cellular system rely on DNA replication and cell division. The replication of DNA (to be detailed in Chapter 2) is a high-fidelity, but not error-free, process. While maintaining the stability of the system, this process also provides the mechanism for the diversification and evolution of the cellular system. The cell division process is also tightly regulated, for the most part to ensure equal transfer of the replicated DNA into daughter cells. For the majority of multicellular organisms that reproduce sexually, during the process of germ cell formation the DNA is replicated once but cell division occurs twice, leading to the reduction of DNA material by half in the gametes. The recombination of DNA from female and male gametes leads to further diversification in the offspring.

1.3 Molecules in Cells

Different types of molecules are needed to carry out the various cellular processes. In a typical cell, water is the most abundant, representing 70% of the total cell weight. Besides water, there is a large variety of small and large

molecules. The major categories of small molecules include inorganic ions (e.g., Na^+, K^+, Ca^{2+}, Cl^-, and Mg^{2+}), monosaccharides, fatty acids, amino acids, and nucleotides. Major varieties of large molecules are polysaccharides, lipids, proteins, and nucleic acids (DNA and RNA). Among these components, the inorganic ions are important for signaling (e.g., waves of Ca^{2+} represent important intracellular signal), cell energy storage (e.g., in the form of Na^+/K^+ cross-membrane gradient), or protein structure/function (e.g., Mg^{2+} is an essential cofactor for many metalloproteins). Carbohydrates (including monosaccharides and polysaccharides), fatty acids, and lipids are major energy-providing molecules in the cell. Lipids are also the major component of cell membrane. Proteins, which are assembled from 20 types of amino acids in different order and length, underlie almost all cellular activities, including metabolism, signal transduction, DNA replication, and cell division. They are also the building blocks of many intracellular structures, such as cytoskeleton (see Section 1.4). Nucleic acids carry the code of life in their nearly endless nucleotide permutations, which not only provide instructions on the assembly of all proteins in cells but also exert control on how such assembly is carried out based on environmental conditions.

1.4 Intracellular Structures or Spaces

Cells maintain a well-organized internal structure (Figure 1.1). Based on the complexity of their internal structure, cells are divided into two major categories: prokaryotic and eukaryotic cells. The fundamental difference between them is whether a nucleus is present. Prokaryotic cells, being the more primordial of the two, do not have a nucleus, and as a result their DNA is located in a nucleus-like but nonenclosed area. Prokaryotic cells also lack organelles, which are specialized and compartmentalized intracellular structures that carry out different cellular functions (detailed next). Eukaryotic cells, on the other hand, contain a distinct nucleus dedicated for DNA storage, maintenance, and expression. Furthermore, they contain various organelles including the endoplasmic reticulum (ER), Golgi apparatus, cytoskeleton, mitochondrion, and chloroplast (plant cells). Following is an introduction to the various intracellular structures and spaces, including the nucleus, the organelles, and other subcellular structures and spaces such as the cell membrane and cytoplasm.

1.4.1 Nucleus

Since DNA stores the code of life, it must be protected and properly maintained to avoid possible damage, and ensure accuracy and stability. As proper execution of the genetic information embedded in the DNA is critical

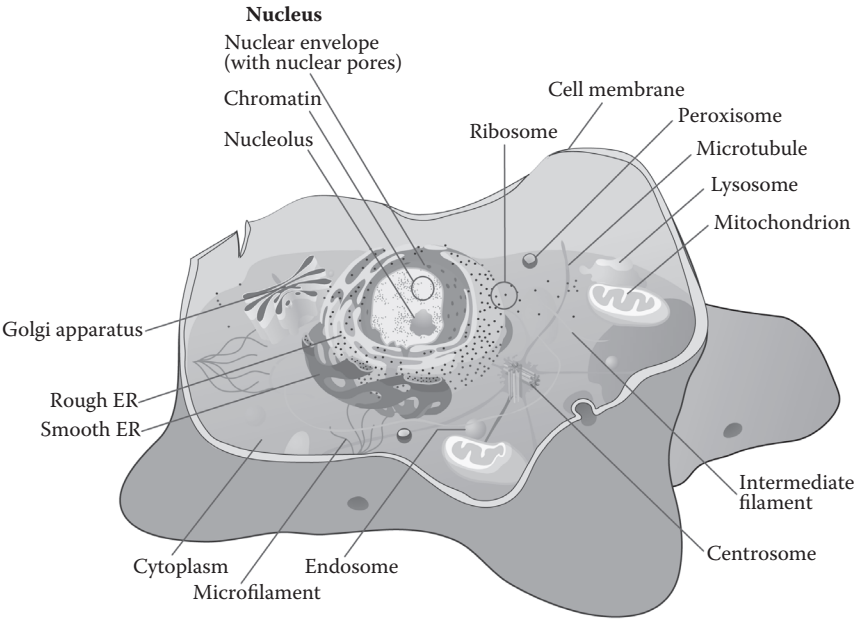

FIGURE 1.1
The general structure of a typical eukaryotic cell. Shown here is an animal cell.

to the normal functioning of a cell, gene expression must also be tightly regulated under all conditions. The nucleus, located in the center of most cells in eukaryotes, offers a well-protected environment for DNA storage, maintenance, and gene expression. The nuclear space is enclosed by a nuclear envelope consisting of two concentric membranes. To allow movement of proteins and RNAs across the nuclear envelope, which is essential for gene expression, there are pores on the nuclear envelope that span the inner and outer membrane. The mechanical support of the nucleus is provided by the nucleoskeleton, a network of structural proteins called lamins. Inside the nucleus, long strings of DNA molecules, through binding to certain proteins called histones, are heavily packed to fit into the limited nuclear space. In prokaryotic cells, a nucleus-like irregularly shaped region that does not have a membrane enclosure called the nucleoid, provides a similar but not as well-protected space for DNA.

1.4.2 Cell Membrane

The cell membrane serves as a barrier to protect the internal structure of a cell from the outside environment. Biochemically, the cell membrane, as well as all other intracellular membranes such as the nuclear envelope, assumes a

lipid bilayer structure. While offering protection to their internal structure, the cell membrane is also where cells exchange materials, and concurrently energy, with the outside environment. Since the membrane is made of lipids, most water-soluble substances, including ions, carbohydrates, amino acids, and nucleotides, cannot directly cross it. To overcome this barrier, there are channels, transporters, and pumps, all of which are specialized proteins, on the cell membrane. Channels and transporters facilitate passive movement, that is, in the direction from high to low concentration, without consumption of cellular energy. Pumps, on the other hand, provide active transportation of the molecules, since they transport the molecules against the concentration gradient and therefore consume energy.

The cell membrane is also where a cell receives most incoming signals from the environment. After signal molecules bind to their specific receptors on the cell membrane, the signal is relayed to the inside, usually eliciting a series of intracellular reactions. The ultimate cellular response that the signal induces is dependent on the nature of the signal, as well as the type and condition of the cell. For example, upon detecting insulin in the blood via the insulin receptor in their membrane, cells in the liver respond by taking up glucose from the blood for storage.

1.4.3 Cytoplasm

Inside the cell membrane, cytoplasm is the thick solution that contains the majority of cellular substances, including all organelles in eukaryotic cells but excluding the nucleus in eukaryotic cells and the DNA in prokaryotic cells. The general fluid component of the cytoplasm that excludes the organelles is called the cytosol. The cytosol makes up more than half of the cellular volume and is where many cellular activities take place, including a large number of metabolic steps such as glycolysis and interconversion of molecules and most signal transduction steps. In prokaryotic cells, due to the lack of a nucleus and other specialized organelles, the cytosol is almost the entire intracellular space and where most cellular activities take place.

Besides water, the cytosol contains large amounts of small and large molecules. Small molecules, such as inorganic ions, provide an overall biochemical environment for cellular activities. In addition, ions such as Na^+, K^+, and Ca^{2+} also have substantial concentration differences between the cytosol and the extracellular space. Cells spend a lot of energy maintaining these concentration differences, and use them for signaling and metabolic purposes. For example, the concentration of Ca^{2+} in the cytosol is normally kept very low at ~10^{-7} M, whereas in the extracellular space it is ~10^{-3} M. The rushing in of Ca^{2+} under certain conditions through ligand- or voltage-gated channels serves as an important messenger, inducing responses in a number of signaling pathways, some of which lead to altered gene expression. Besides small molecules, the cytosol also contains large numbers of macromolecules. Far from being simply randomly diffusing in the cytosol,

these large molecules form molecular machines that collectively function as a "bustling metropolitan city" [1]. These supramacromolecular machines are usually assembled out of multiple proteins, or proteins and RNA. Their emergence and disappearance are dynamic and regulated by external and internal conditions.

1.4.4 Endosome, Lysosome, and Peroxisome

Endocytosis is when cells bring in macromolecules, or other particulate substances such as bacteria or cell debris, into the cytoplasm from the surroundings. Endosome and lysosome are two organelles that are involved in this process. To initiate endocytosis, part of the cell membrane forms a pit, engulfs the external substances, and then an endocytotic vesicle pinches off from the cell membrane into the cytosol. Endosome, normally in the size range of 300 to 400 nm in diameter, forms from the fusion of these endocytotic vesicles. The internalized materials contained in the endosome are sent to other organelles such as lysosome for further digestion.

The lysosome is the principal site for intracellular digestion of internalized materials as well as obsolete components inside the cell. Like the condition in our stomach, the inside of the lysosome is acidic (pH at 4.5–5.0), providing an ideal condition for the many digestive enzymes within. These enzymes can break down proteins, DNA, RNA, lipids, and carbohydrates. Normally the lysosome membrane keeps these digestive enzymes from leaking into the cytosol. Even in the event of these enzymes leaking out of the lysosome, they can do little harm to the cell, since their digestive activities are heavily dependent on the acidic environment inside the lysosome, whereas the pH of the cytosol is slightly alkaline (around 7.2).

Peroxisome is morphologically similar to the lysosome, however it contains a different set of proteins, mostly oxidative enzymes that use molecular oxygen to extract hydrogen from organic compounds to form hydrogen peroxide. The hydrogen peroxide can then be used to oxidize other substrates, such as phenols or alcohols, via peroxidation reaction. As an example, liver and kidney cells use these reactions to detoxify various toxic substances that enter the body. Another function of the peroxisome is to break down long-chain fatty acids into smaller molecules by oxidation. Despite its important functions, the origin of peroxisome is still under debate. One theory proposes that this organelle has an endosymbiotic origin [2]. If this theory holds true, all genes in the genome of the original endosymbiotic organism must have been transferred to the nuclear genome. Another theory proposes that the peroxisome is a remnant of an ancient organelle that served to lower intracellular oxygen levels when the oxygen that we depend on today was still highly toxic to most cells, while exploiting the chemical reactivity of oxygen to carry out useful oxidative reactions for the host cell. Also based on this theory, the mitochondrion (see later) that emerged later releases energy from many of the same oxidative reactions that had previously taken place

in the peroxisome but without generating any energy, thereby rendering the peroxisome largely irrelevant except for carrying out the remnant oxidative functions.

1.4.5 Ribosome

Ribosome is the protein assembly factory in cells, translating genetic information carried in messenger RNAs (mRNAs) into proteins. There are vast numbers of ribosomes, usually from thousands to millions, in a typical cell. Whereas both prokaryotic and eukaryotic ribosomes are composed of two components (or subunits), eukaryotic ribosomes are larger than their prokaryotic counterparts. In eukaryotic cells, the two ribosomal subunits are first assembled inside the nucleus in a region called the nucleolus and then shipped out to the cytoplasm. In the cytoplasm, ribosomes can be either free or get attached to another organelle (the ER). Biochemically, ribosomes contain more than 50 proteins and several ribosomal RNA (rRNA) species. Because ribosomes are highly abundant in cells, rRNAs are the most abundant in total RNA extracts, accounting for 85% to 90% of all RNA species. For profiling cellular RNA populations using next-generation sequencing (NGS), rRNAs are usually not of interest despite their abundance and therefore need to be depleted to avoid generation of overwhelming amounts of sequencing reads from them.

1.4.6 Endoplasmic Reticulum (ER)

As indicated by the name, the endoplasmic reticulum (ER) is a network of membrane-enclosed spaces throughout the cytosol. These spaces interconnect and form a single internal environment called the ER lumen. There are two types of ERs in cells: rough ER and smooth ER. The rough ER is where all cell membrane proteins, such as ion channels, transporters, pumps, and signal molecule receptors, as well as secretory proteins, such as insulin, are produced and sorted. The characteristic surface roughness of this type of ER comes from the ribosomes that bind to them on the outside. Proteins destined for cell membrane or secretion, once emerging from these ribosomes, are threaded into the ER lumen. This ER-targeting process is mediated by a signal sequence, or "address tag," located at the beginning part of these proteins. This signal sequence is subsequently cleaved off inside the ER before the protein synthesis process is complete. Functionally different from the rough ER, the smooth ER plays an important role in lipid synthesis for the replenishment of cellular membranes. Besides membrane and secretory protein preparation and lipid synthesis, one other important function of the ER is to sequester Ca^{2+} from the cytosol. In Ca^{2+}-mediated cell signaling, shortly after entry of the calcium wave into the cytosol, most of the incoming Ca^{2+} needs to be pumped out of the cell and/or sequestered into specific organelles such as the ER and mitochondria.

1.4.7 Golgi Apparatus

Besides the ER, the Golgi apparatus also plays an indispensable role in sorting as well as dispatching proteins to the cell membrane, extracellular space, or other subcellular destinations. Many proteins synthesized in the ER are sent to the Golgi apparatus via small vesicles for further processing before being sent to their final destinations. Therefore, the Golgi apparatus is sometimes metaphorically described as the "post office" of the cell. The processing carried out in this organelle includes chemical modification of some of the proteins, such as adding oligosaccharide side chains, which serve as "address labels." Other important functions of the Golgi apparatus include synthesizing carbohydrates and extracellular matrix materials, such as the polysaccharide for the building of the plant cell wall.

1.4.8 Cytoskeleton

Cellular processes like the trafficking of proteins in vesicles from the ER to the Golgi apparatus or the movement of a mitochondrion from one intracellular location to another are not simply based on diffusion. Rather, they follow a certain protein-made skeletal structure inside the cytosol, that is, the cytoskeleton, as tracks. Besides providing tracks for intracellular transport, the cytoskeleton, like the skeleton in the human body, plays an equally important role in maintaining cell shape and protecting the cell framework from physical stresses, as the lipid bilayer cell membrane is fragile and vulnerable to such stresses. In eukaryotic cells, there are three major types of cytoskeletal structures: microfilament, microtubule, and intermediate filament. Each type is made of distinct proteins and has its own unique characteristics and functions. For example, microfilament and microtubule are assembled from actins and tubulins, respectively, and have different thicknesses (the diameter is about 6 nm for microfilament and 23 nm for microtubule). Although biochemically and structurally different, both the microfilament and the microtubule have been known to provide tracks for mRNA transport in the form of large ribonucleoprotein complexes to specific intracellular sites, such as the distal end of a neuronal dendrite, for targeted protein translation [3,4]. Besides its role in intracellular transportation, the microtubule also plays a key role in cell division through attaching to the duplicated chromosomes and moving them equally into two daughter cells. In this process, all microtubules involved are organized around a small organelle called centrosome. Previously thought to be only present in eukaryotic cells, cytoskeletal structures have also been discovered in prokaryotic cells [5].

1.4.9 Mitochondrion

The mitochondrion is the "powerhouse" in eukaryotic cells. While some energy is produced from the glycolytic pathway in the cytosol, most energy

is generated from the Krebs cycle and the oxidative phosphorylation process that take place in the many mitochondria contained in a cell. The number of mitochondria in a cell is ultimately dependent on its energy demand. The more energy a cell needs, the more mitochondria it has. Structurally, the mitochondrion is an organelle enclosed by two membranes. The outer membrane is highly permeable to most cytosolic molecules, and as a result the intermembrane space between the outer and inner membranes is similar to the cytosol. Most of the energy-releasing process occurs in the inner membrane and in the matrix, that is, the space enclosed by the inner membrane. For the energy release, high-energy electron carriers generated from the Krebs cycle in the matrix are fed into an electron transport chain embedded in the inner membrane. The energy released from the transfer of high-energy electrons through the chain to molecular oxygen (O_2), the final electron acceptor, creates a proton gradient across the inner membrane. This proton gradient serves as the energy source for the synthesis of ATP, the universal energy currency in cells. In prokaryotic cells, since they do not have this organelle, ATP synthesis takes place on their cytoplasmic membrane instead.

The origin of the mitochondrion, based on the widely accepted endosymbiotic theory, is an ancient α-proteobacteria. So not surprisingly, the mitochondrion carries its own DNA, but the genetic information contained in mitochondrial DNA (mtDNA) is extremely limited compared to nuclear DNA. Human mitochondrial DNA, for example, is 16,569 bp in size coding for 37 genes, including 22 for transfer RNAs (tRNAs), 2 for rRNAs, and 13 for mitochondrial proteins. Although it is much smaller compared to the nuclear genome, there are multiple copies of mtDNA molecules in each mitochondrion. Since cells usually contain hundreds to thousands of mitochondria, there are a large number of mtDNA molecules in each cell. In comparison, most cells only contain two copies of the nuclear DNA. As a result, when sequencing cellular DNA samples, sequences derived from mitochondrial DNA usually comprise a notable, sometimes substantial, percentage of total generated reads. Although small, the mitochondrial genomic system is fully functional and has the entire set of protein factors for mtDNA transcription, translation, and replication. As a result of its activity, when cellular RNA molecules are sequenced, those transcribed from the mitochondrial genome also generate significant amounts of reads in the sequence output.

The many copies of mtDNA molecules in a cell may not all have the same sequence due to mutations in individual molecules. Heteroplasmy occurs when cells contain a heterogeneous set of mtDNA molecules. In general, mitochondrial DNA has a higher mutation rate than its nuclear counterpart. This is because the transfer of high-energy electrons along the electron transport chain can produce reactive oxygen species as byproducts, which can oxidize and cause mutations in mtDNA. To make this situation even worse, the DNA repair capability in mitochondria is rather limited. Increased heteroplasmy has been associated with a higher risk of developing aging-related diseases, including Alzheimer disease, heart disease, and Parkinson's disease [6–8].

Furthermore, certain mitochondrial DNA mutations and deletions have been known to underlie a number of diseases that mostly affect the nervous system and muscle due to their high energy demand [9]. Characteristically, these diseases are maternally inherited, as mitochondrial DNA is passed on from mother to offspring.

1.4.10 Chloroplast

In animal cells, the mitochondrion is the only organelle that contains an extranuclear genome. Plant and algae cells have another extranuclear genome besides the mitochondrion, the plastid genome. Plastid is an organelle that can differentiate into various forms, the most prominent of which is the chloroplast. The chloroplast carries out photosynthesis by capturing the energy in sunlight and fixing it into carbohydrates using carbon dioxide as substrate, and releasing oxygen in the same process. For energy capturing, the green pigment called chlorophyll first absorbs energy from sunlight, which is then transferred through an electron transport chain to build up a proton gradient to drive the synthesis of ATP. Despite the energy source, the buildup of proton gradient for ATP synthesis in the chloroplast is very similar to that for ATP synthesis in the mitochondrion. The chloroplast ATP derived from the captured light energy is then spent on CO_2 fixation. Similar to the mitochondrion, the chloroplast also has two membranes, a highly permeable outer membrane and a much less permeable inner membrane. The photosynthetic electron transport chain, however, is not located in the inner membrane but in the membrane of a series of saclike structures called thylakoids located in the chloroplast stroma (analogous to the mitochondrial matrix).

Plastid is believed to have evolved from an endosymbiotic cyanobaterium, which has gradually lost the majority of its genes in its genome over millions of years. The current size of most plastid genomes is 100 to 200 kb, coding for rRNAs, tRNAs, and proteins. In higher plants there are about 85 genes coding for various proteins of the photosynthetic system [10]. The transmission of plastid DNA (ptDNA) from parent to offspring is more complicated than the maternal transmission of mtDNA usually observed in animals. Based on the transmission pattern, it can be classified into three types: (1) maternal, inheritance only through the female parent; (2) paternal, inheritance only through the male parent; or (3) bioparental, inheritance through both parents [11]. Similar to the situation in mitochondrion, there exist multiple copies of ptDNA in each plastid, and as a result there are large numbers of ptDNA molecules in each cell with potential heteroplasmy. Transcription from these ptDNA also generates copious amounts of RNAs in the organelle. Therefore, sequence reads from ptDNA or RNA comprise part of the data when sequencing plant and algae DNA or RNA samples, along with those from mtDNA or RNA.

1.5 The Cell as a System

1.5.1 The Cellular System

From the aforementioned description of a typical cell, it is obvious that the cell is a self-organizing system, containing many different molecules and structures that work together coherently. Unlike other nonbiological systems, including natural and artificial systems such as a car or a computer, the cell system is unique as it continuously renews and perpetuates itself without violating the laws of the physical world. It achieves this by obtaining energy from and exchanging materials with its environment. The cellular system is also characterized by its autonomy, that is, all of its activities are self-regulated. This autonomy is conferred by the genetic instructions coded in the cell's DNA. Besides the above characteristics, the cell system is highly robust, as its homeostasis is not easily disturbed by changes in its surroundings. This robustness is a result of billions of years of evolution, which has led to the building of tremendous complexity into the system. To study this complexity, biologists have been mostly taking a reductionist approach to study the different cellular molecules and structures piece by piece. This approach has been highly successful and much knowledge has been gathered on most parts of the system. For a cell to function as a single entity, however, these different parts do not work alone. To study how it operates as a whole, the different parts need to be studied in the context of the entire system and therefore a holistic approach is also needed. It has become clearer to researchers in the life science community that the interactions between the different cellular parts are equally, if not more, important as any part alone.

1.5.2 Systems Biology of the Cell

Systems biology is an emerging field that studies the complicated interactions among the different parts of biological systems. It is an application of the systems theory to the biological field. Introduced by the biologist Ludwig von Bertalanffy in the 1940s, this theory aims to investigate the principles common to all complex systems and to describe these principles using mathematical models. This theory is applicable to many disciplines including physics, sociology, and biology, and one goal of this theory is to unify the principles of systems as uncovered from the different disciplines. It is expected, therefore, that principles uncovered from other systems may be applicable to biological systems and provide guidance to better understanding of their working.

In the traditional reductionist approach, a single gene or protein is the basic functioning unit. In systems biology, however, the basic unit is a genetic circuit. A genetic circuit can be defined as a group of genes (or the proteins they code) that work together to perform a certain task. There are a multitude of

tasks in a cell that need to be carried out by genetic circuits, from the transduction of extracellular signal to the inside, to the step-by-step breakdown of energy molecules (such as glucose) to release energy, to the replication of DNA prior to cell division. It is these genetic circuits that underlie cellular behavior and physiology. If the information or material flux in a genetic circuit is blocked or goes awry, the whole system will be influenced, which might lead to the malfunctioning of the system and likely a diseased state.

Based on the hierarchical organization principle of systems, gene circuits interact with each other and form a complicated genetic network. Mapping out a genetic network is a higher goal of systems biology. A genetic network has been shown to share some common characteristics with nonbiological networks such as the human society or the Internet [12]. One such characteristic is modularity, for instance, when genes (or proteins) that often work together to achieve a common goal form a module and the module is used as a single functional unit when needed. Another common characteristic is the existence of hub or anchor nodes in the network; for example, a small number of highly connected genes (or proteins) in a genetic network serve as hubs or anchors through which other genes (or proteins) are connected to each other.

1.5.3 How to Study the Cellular System

Research into the systems biology of the cell is largely enabled by technological advancements in genomics, proteomics, and metabolomics. High-throughput genomics technologies, for example, allow simultaneous analysis of tens of thousands of genes in an organism's genome. Genome refers to the whole set of genetic material in an organism's DNA, including both protein-coding and noncoding sequences. Similarly, proteome and metabolome are defined as the complement of proteins and metabolites (small molecules), respectively, in a cell or population of cells. Proteomics, through simultaneous separation and identification of proteins in a proteome, provides answers to the questions of how many proteins are present in the target cell(s) and at what abundance levels. Metabolomics, on the other hand, through analyzing a large number of metabolites simultaneously, monitors the metabolic status of target cells.

The development of modern genomics technologies was mostly initiated when the human genome was sequenced by the Human Genome Project. The completion of the sequencing of this genome and the genomes of other organisms, and the concurrent development of genomics technologies, have for the first time offered an opportunity to study the systems properties of the cell. The first big wave of genomics technologies was mostly centered on microarray, which enables analysis of the transcriptome and subsequent study of genome-wide sequence polymorphism in a population. By studying all RNAs transcribed in a cell or population of cells, transcriptomic analysis investigates what genes are active and how active. Determination

of genome-wide sequence variations among individuals in a population enables examination of the relationship between certain genomic polymorphisms and cellular dysfunctions, phenotypic traits, or diseases. More recently, the development of NGS technologies provides more power, coverage, and resolution to the study of the genome (for details on the development of NGS technologies, see Chapter 4). These NGS technologies, along with recent technological developments in proteomics and metabolomics, further empower the study of the cellular system.

2

DNA Sequence: The Genome Base

2.1 The DNA Double Helix and Base Sequence

Among the different types of molecules in cells, DNA has a structure that makes it ideal to code the blueprint of life. The building blocks of DNA are nucleotides, which are made up of three chemical groups: a five-carbon sugar (deoxyribose), phosphate, and one of four nucleobases. The spatial structure of DNA is a double helix comprising two strands. The backbone of each strand is made of the sugar moiety and phosphate, which are invariably connected in an alternating fashion and therefore do not carry genetic information. The "rungs" that connect the two strands are composed of nucleobases, which are where the information is stored. Since the discovery of this structure in 1953 by Watson and Crick, the elegance and simplicity of this structure has fascinated generations of biologists, chemists, and scientists from other fields.

There are four different types of nucleobases (or simply bases) in DNA: two purines (adenine, usually abbreviated as A; and guanine, G) and two pyrimidines (cytosine, C; and thymine, T). Nucleobases in the two DNA strands that form the rung structure interact via hydrogen bonding in a fixed manner: A always pairing with T, and C with G. This complementary base-pairing pattern enables the DNA molecule to assume the most thermodynamically favorable structure. The fixed pairing pattern between the bases makes it easy to provide coding for life and to replicate for perpetuation.

The almost endless arrangements of the base pairs in DNA provide the basis for DNA's role as the genetic information carrier. The information embedded in the DNA base sequence dictates what, when, and how many proteins are made in a cell at a certain point of time. At a deeper level, the information codes for the entire operating logic of the cellular system. It contains all instructions needed to form a new life and for it to grow, develop, and reproduce. From the medical point of view, alterations or polymorphisms in the DNA base sequence can predispose us to certain diseases as well as underlie our responses to medications.

2.2 How DNA Molecules Replicate and Maintain Fidelity

The DNA's double helix structure and complementary base-pairing make it robust to copy the bioinformation it carries through its replication. To replicate, the two strands of the parent DNA molecule are first unwound by an enzyme called helicase. The two unwound strands then serve as templates for the synthesis of new complementary strands, giving rise to two offspring DNA molecules. The enzyme that carries out the new strand synthesis is called DNA polymerase, which assembles nucleotides into a new strand by adding one nucleotide at a time to a preexisting primer sequence based on complementary base-pairing with the template strand (Figure 2.1). Biochemically, the enzyme catalyzes the formation of a covalent phosphodiester bond between the 5′-phosphate group of the incoming complementary nucleotide and the 3′-hydroxyl group on the elongating strand end. Besides elongating the new DNA strand, most DNA polymerases also have proofreading capability. If a nucleotide that is not complementary to the template is accidentally attached to the end of the elongating strand (i.e., mispairing), the enzyme will turn around and cleave the wrong nucleotide off. This proofreading activity is important to maintain the high fidelity of the DNA replication process. Mutations, or sudden changes of nucleotide sequence in DNA, would occur much more frequently without this activity.

Many sequencing technologies are based on the process of DNA replication. These technologies, often referred to as sequencing-by-synthesis, use this process to read the nucleotide sequence off one strand of the sequencing DNA target. Corresponding to the components required in the DNA replication process, these sequencing systems require the following basic components: (1) a sequencing DNA target, which provides the template; (2) nucleotides; (3) a primer; and (4) a DNA polymerase. Since the DNA polymerase extends the new strand by attaching one nucleotide at a time, detecting the attached nucleotide after each extension cycle generates a readout of the nucleotide sequence on the template DNA strand. To facilitate the detection, the nucleotides used in sequencing reactions are usually chemically modified, including labeling with fluorescent tags. Chapter 4 focuses on the evolution of sequencing technologies.

Besides the high fidelity of DNA polymerases, an efficient DNA repair system is also crucial to maintain genome stability and keep the mutation rate low. Even under normal conditions, a DNA nucleotide sequence can be accidentally altered by many physical and chemical factors in the environment, including intracellularly generated reactive oxygen and nitrogen species, radiation in the environment (such as UV, x-ray, or γ-ray), and other chemical mutagens. If left uncorrected, these changes will accumulate and cause disturbances to normal cell function or even cause cell death, leading to diseases. To maintain the fidelity of DNA molecules, cells invest heavily on DNA repair enzymes. These enzymes constantly scan genomic DNA and

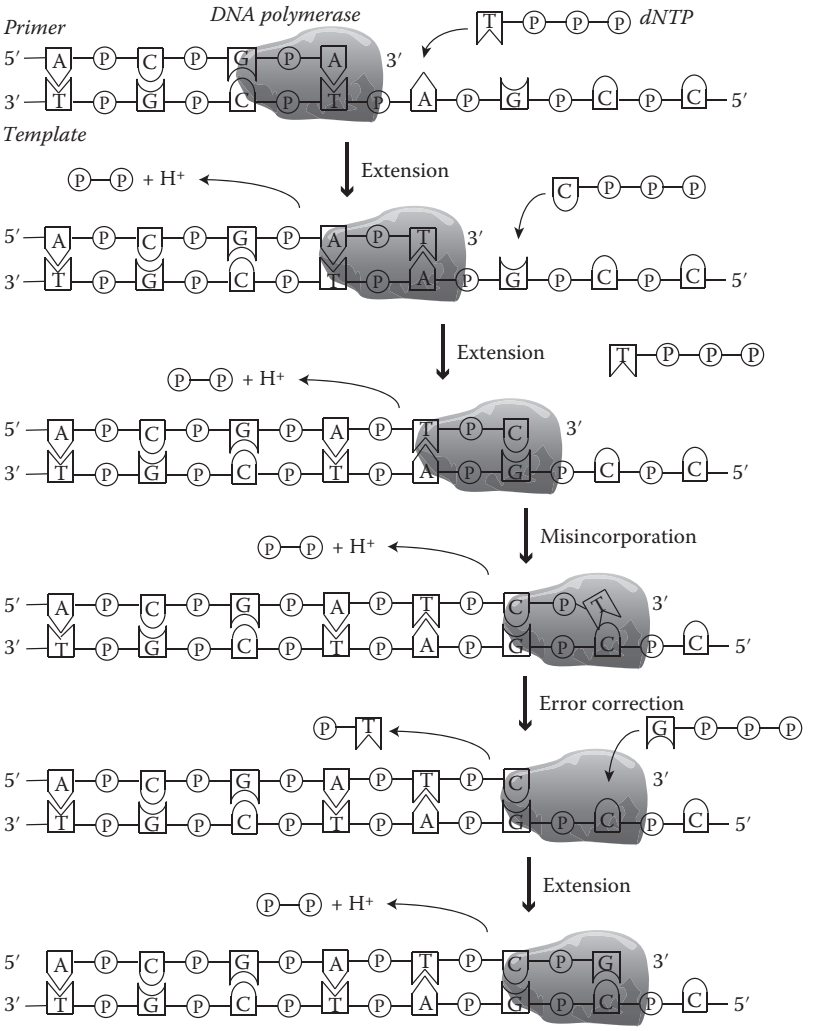

FIGURE 2.1
The DNA replication process. To initiate the process, a primer, which is a short DNA sequence complementary to the start region of the DNA template strand, is needed for DNA polymerase to attach nucleotides and extend the new strand. The attachment of nucleotides is based on complementary base-pairing with the template. If an error occurs due to mispairing, the DNA polymerase removes the mispaired nucleotide using its proofreading function. Due to the biochemical structure of the DNA molecule, the direction of the new strand elongation is from its 5′ end to the 3′ end (the template strand is in the opposite direction; the naming of the two ends of each DNA strand as 5′ and 3′ is from the numbering of carbon atoms in the nucleotide sugar ring).

make repairs if damage is detected. The serious consequences of a weakened DNA repair system can be exemplified by mutations in *BRCA2*, a gene coding for a DNA repair enzyme, which lead to breast and ovarian cancers.

2.3 How the Genetic Information Stored in DNA Is Transferred to Protein

While the logic of the cellular system is written in the nucleotide sequence of its genomic DNA, almost all cellular activities are executed by the wide array of proteins in the cell's proteome. The bioinformation flow from DNA to protein, known as the central dogma (Figure 2.2), provides a fundamental framework for modern molecular biology and genetic engineering. Based on this framework, a gene's DNA sequence is first transcribed to make mRNA, and then the nucleotide sequence in mRNA is used to guide the assembly of amino acids into a protein. The translation of the mRNA nucleotide sequence to the protein amino acid sequence is based on the triplet genetic code. A continuous segment of DNA that contains the full set of triplet codon for protein translation, from start to stop, is often called an open reading frame (or ORF). The synthesis of one type of biopolymer molecule based on information stored in another biopolymer is one of the greatest "inventions" of nature.

Since its initial introduction, the central dogma has been gradually modified with increased sophistication. In its original form, one gene is translated into one protein via one mRNA. This one gene–one protein paradigm was later found to be too simplistic, as one gene can generate multiple forms of

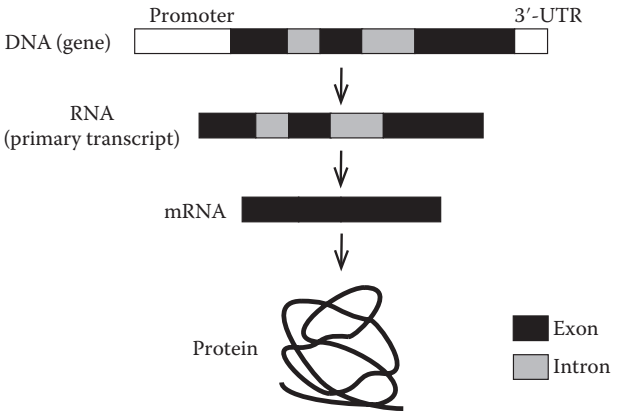

FIGURE 2.2
The central dogma.

proteins through alternative splicing (see Chapter 3). In addition, the information flow between DNA and RNA is not simply one-way from DNA to RNA, but RNA can also be reverse transcribed to DNA in some organisms. On the additional role of RNAs in this information flow, some non-protein-coding RNAs can silence gene expression through mechanisms such as inhibiting gene transcription or translation, or protect genomes through mechanisms like preventing the movement of transposable elements (or transposons, mobile DNA elements that copy themselves to different genomic loci) (also see Chapter 3). Furthermore, chemical modifications of DNA and some DNA-interacting proteins constitute the epigenome, which also regulates the flow of genetic information.

2.4 The Genomic Landscape

2.4.1 The Minimal Genome

After understanding the flow of bioinformation from DNA to protein, the next question is what is the minimum amount of genetic information needed to make the cellular system tick, that is, what constitutes the minimal genome. Attempts to define the minimal genome started in the late 1950s, shortly after the discovery of the double helix structure of DNA. The answer to this important question is not straightforward, however, as the amount of genetic information needed for a minimal life form is dependent on the specific environment it lives in. Considering the basic functions that a cell has to perform, the minimal genome needs to contain genes at least for DNA replication, RNA synthesis and processing, protein translation, energy, and molecular metabolism. The current estimate is that 150 to 300 genes are required at a minimum for any genome. A small bacterium, *Mycoplasma genitalium*, containing a genome of 580,076 bp, has often been used as a model of a naturally existing minimal genome for a free-living organism because of its minimal metabolism and little genomic redundancy [13]. Among the ~480 protein-coding genes contained in its genome, 382 are shown to be essential [14].

2.4.2 Genome Sizes

For the least sophisticated organisms, such as *Mycoplasma genitalium*, a minimal genome is sufficient. For increased organismal complexity, more genetic information and, therefore, a larger genome is needed. As a result, there is a positive correlation between organismal complexity and genome size, especially in prokaryotes. In eukaryotes, however, this correlation becomes much weaker, largely due to the existence of noncoding DNA elements in

TABLE 2.1

Genome Sizes and Total Gene Numbers in Major Model Organisms (Ordered by Genome Size)

Organism	Genome Size (bp)[a]	Number of Coding Genes
Mycoplasma genitalium (bacterium)	580,076	476
Haemophilus influenzae (strain 86-028NP) (bacterium)	1,914,490	1792
Escherichia coli (strain K-12) (bacterium)	4,646,332	4227
Saccharomyces cerevisiae (yeast)	12,157,105	6692
Caenorhabditis elegans (nematode)	103,022,290	20,447
Arabidopsis thaliana (thale cress)	135,670,229	27,416
Drosophila melanogaster (fruit fly)	168,736,537	13,937
Medicago truncatula (legume)	309,576,036	44,115
Oryza sativa (japonica subspecies) (rice)	374,424,240	35,679
Danio rerio (zebrafish)	1,505,581,940	26,459
Rattus norvegicus (rat)	2,573,362,844	22,777
Zea mays (maize)	3,233,616,351	39,469
Homo sapiens (human)	3,381,944,086	20,364
Mus musculus (mouse)	3,482,005,469	22,606

[a] Data based on Ensembl genome databases as of February 2015.

varying amounts in different eukaryotic genomes (for details on noncoding DNA elements, see Section 2.4.4). In terms of total gene number, the currently documented range is 182 in the genome of *Candidatus* Carsonella ruddii (a parasitic/endosymbiotic bacterium) [15] to 30,907 genes in the genome of *Daphnia pulex* (a water flea) [16]. Table 2.1 shows the total number of genes in some of the most studied organisms.

2.4.3 Protein-Coding Regions of the Genome

The protein-coding regions are the part of the genome that we foremost study and know most about. The content of these regions directly affects protein synthesis and protein diversity in cells. In prokaryotic cells, functionally related protein-coding genes are often arranged next to each other and regulated as a single unit known as an operon. The gene structure in eukaryotic cells is more complicated. The coding sequences (CDSs) of almost all eukaryotic genes are not continuous and interspersed among noncoding sequences. The noncoding intervening sequences are called introns (*int* for intervening), whereas the coding regions are called exons (*ex* for expressed) (see Figure 2.2). During gene transcription, both exons and introns are transcribed. In the subsequent mRNA maturation process, introns are spliced out and exons are joined together for protein translation.

In the human genome, the average number of exons per gene is 8.8. The *titin* gene, coding for a large abundant protein in striated muscle, has

363 exons, the most in any single gene, and also has the longest single exon (17,106 bp) among all currently known exons. The total number of currently known exons in the human genome is around 180,000. With a combined size of 30 Mb, they constitute 1% of the human genome. This collection of all exons in the human genome, or in other eukaryotic genomes, is termed as the exome. Different from the transcriptome, which is composed of all actively transcribed mRNAs in a particular sample, the exome includes all exons contained in a genome. Although it only covers a very small percentage of the genome, the exome represents the most important and the best annotated part of the genome. Sequencing of the exome has been used as a popular alternative to whole genome sequencing. While it lacks on coverage, exome sequencing is more cost effective, faster, and easier for data interpretation.

2.4.4 Noncoding Genomic Elements

Although protein-coding genes are the most studied genomic element, they may not necessarily be the most abundant part of the genome. Prokaryotic genomes are usually rich in protein-coding gene sequences, for example, they account for approximately 90% of the *E. coli* genome. In complex eukaryotic genomes, however, their percentage is lower. For example, only about 1.5% of the human genome codes for proteins (Figure 2.3). Among the non-protein-coding sequences in eukaryotic genomes are introns, regulatory sequences, and other unique noncoding DNA elements. The regulatory sequences are genomic elements that are known to regulate gene expression, including promoters, terminators, enhancers, repressors, and silencers. In comparison, our current understanding of the other unique noncoding DNA elements is the most rudimentary. We know nearly nothing about these elements, with the exception of noncoding RNA genes, which include ribosomal RNAs (rRNAs), transfer RNAs (tRNAs), and other functionally important RNA species that will be detailed in Chapter 3. As mentioned in Chapter 1, rRNAs are key structural components of the ribosome and directly involved in protein translation, whereas tRNAs transport proper amino acids to the ribosome for protein translation based on the genetic code.

Repetitive sequences occupy more than half of the human genome and are even more pervasive in some other eukaryotic genomes. For example, in some plants and amphibians, 80% of the genome is composed of repetitive sequences. The percentage of repetitive sequences in prokaryotic genomes is relatively lower but still significant. With respect to their internal structures, some repetitive sequences are tandem repeats, with the basic repeating units connected head to tail. In this type of sequence repeats, the length of the repeating units is highly variable, from <10 bp to thousands of base pairs. The other major type of sequence repeats is interspersed repeats, present as a single copy in many genomic loci. These are either transposons or retrotransposons that copy themselves via RNA intermediates. Discovered

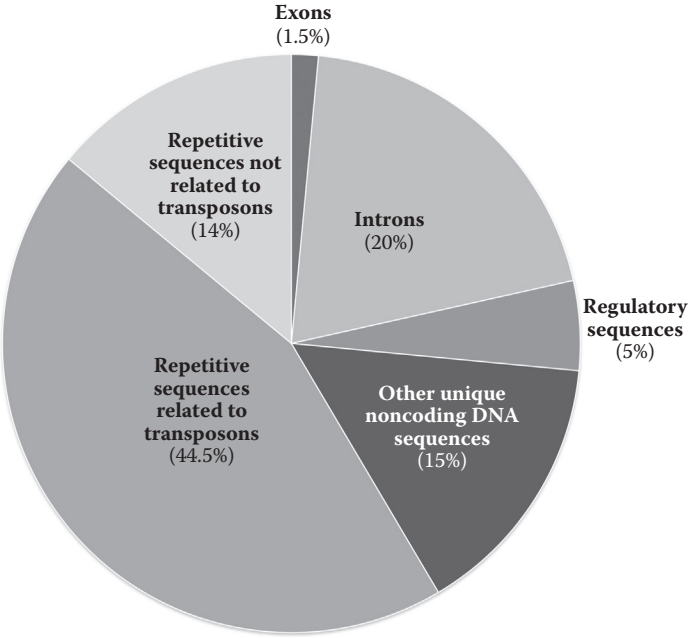

FIGURE 2.3
The composition of the human genome.

by geneticist Barbara McClintock, transposons (also called transposable elements, or "jumping genes") are DNA sequences that move from one genomic location to another. Repeat sequence units of this type are usually 100 bp to over 10 kb in length, and may appear in over 1 million loci dispersed across the genome.

Many highly repetitive DNA sequences exist in inert parts of chromosomes, such as the centromere and telomere. The centromere, the region where two sister chromatids are linked together before cell division, contains tandem repeat sequences. The telomere, existing at the ends of chromosomes, is also composed of highly repetitive DNA sequences. The telomeric structure protects chromosomal integrity and thereby maintains genomic stability. Besides being essential in maintaining the chromosomal structure, repeat sequences have other functions in the genome, for example, they play an architectonic role in higher-order physical genome structuring [17]. Despite their abundance and function, because sequences associated with repeat regions are not unique, they create a major hurdle for assembling a genome *de novo* from sequencing reads or mapping reads originated from these regions to a preassembled genome.

2.5 DNA Packaging, Sequence Access, and DNA–Protein Interactions

2.5.1 DNA Packaging

In the nucleoid of prokaryotic cells, multiple proteins fold and condense genomic DNA into a supercoiled structure to make it fit into the rather limited space. While being generally condensed, parts of the DNA need to be exposed to allow sequence access for transcription by related protein factors. Although these processes have been studied in prokaryotic cells, DNA packaging and sequence accessing are better studied and understood in eukaryotic cells. In these cells, because of their much larger genome size, genomic DNA is condensed in the nucleus to a much higher degree. For instance, the total length of human genomic DNA is about 2 m when fully stretched out, but the diameter of the human cell nucleus is only 6 μm. Bound to specific proteins called histones, eukaryotic DNA is packaged in the form of chromatin, in which the positively charged histones bind to the negatively charged DNA molecules through electrostatic interactions. This packaging process involves compacting DNA at different levels. At the first level, DNA wraps around a protein complex composed of eight histone subunits to form the basic structure of nucleosome. Each nucleosome contains about 200 nucleotide pairs and has a diameter of 11 nm. At the second level, the nucleosome structure is compacted into a fiber structure. This fiber, with a diameter of 30 nm, is the form most chromatin takes in the interphase between two cell divisions. Prior to cell division, this chromatin fiber is further condensed by two additional levels into chromosome, the extremely condensed form that we can observe under a light microscope.

2.5.2 Sequence Access

Since different DNA sequences in the genome are constantly being transcribed, instead of being permanently locked into the compacted form, DNA sequences at specific loci need to be dynamically exposed to allow transcriptional access to protein factors such as transcription factors and coactivators. Furthermore, DNA replication and repair also require chromatin unpackaging. This unpackaging of the chromatin structure is carried out through two principal mechanisms. One is through histone modification, such as acetylation of lysine residues on histones by histone acetyltransferases, which reduces the positive charge on histones and therefore decreases the electrostatic interactions between histones and DNA. Deacetylation by histone deacetylases, on the other hand, restricts DNA access and represses transcription. The other unpackaging mechanism is through the actions of chromatin remodeling complexes. These large protein complexes consume

ATP and use the released energy to expose DNA sequences for transcription through nucleosomal repositioning, nucleosomal eviction, or local unwrapping.

2.5.3 DNA–Protein Interactions

While DNA is the carrier of the code of life, the DNA code cannot be executed without DNA-interacting proteins. Nearly all of the processes mentioned earlier, including DNA packaging/unpackaging, transcription, repair, and replication, rely on such proteins. Besides histones, examples of these proteins include transcription factors, RNA polymerases, DNA polymerases, and nucleases (for DNA degradation). Many of these proteins, such as histones and DNA/RNA polymerases, interact with DNA regardless of their sequence or structure. Some DNA-interacting proteins bind to DNA of special structure/conformation, for example, high-mobility group (HMG) proteins that have high affinity for bent or distorted DNA. Some other DNA-interacting proteins bind only to regions of the genome that have certain characteristics such as having damage, the examples of which are DNA repair enzymes such as BRCA1, BRCA2, RAD51, RAD52, and TDG.

The most widely studied DNA-interacting proteins are transcription factors, which bind to specific DNA sequences. Through binding to their specific recognition sequences in the genome, transcription factors regulate transcription of gene targets that contain such sequences in their promoter region. Since they bind to more than one gene location in the genome, transcription factors regulate the transcription of a multitude of genes in a coordinated fashion, usually as a response to certain internal or external environmental changes. For instance, NRF2 is a transcription factor that is activated in response to oxidative stress. Upon activation, it binds to a short segment of a specific DNA sequence called the antioxidant response element (ARE), located in the promoter region of those genes that are responsive to oxidative stress. Through binding to this sequence element in many regions of the genome, NRF2 regulates the transcription of its target genes and thereby elicits coordinated responses to counteract the damaging effects of oxidative stress.

Study of DNA–protein interactions provides insights into how the genome responds to various conditions. For example, determination of transcription factor binding sites, such as those of NRF2, across the genome can unravel what genes might be responsive to the conditions that activate the transcription factors. Although such sites can be predicted computationally, only a wet-lab experiment can determine where a transcription factor actually binds in the genome under a certain condition. ChIP-Seq, or chromatin immunoprecipitation coupled with sequencing, is one application of next-generation sequencing (NGS) that is developed to study genomic binding of transcription factors and other DNA-interacting proteins. Chapter 11 focuses on ChIP-Seq data analysis.

2.6 DNA Sequence Mutation and Polymorphism

Although DNA replication is a high-fidelity process and the nucleus maintains an army of DNA repair enzymes, sequence mutation does happen, though at a very low frequency. In general, the rate of mutation in prokaryotic and eukaryotic cells is at the scale of 10^{-9} per base per cell division. In multicellular eukaryotic organisms, germline cells have a lower mutation rate than somatic cells. In these organisms, because most cells, including germline cells, undergo multiple divisions in the organisms' lifetime, the per-generation mutation rate is significantly higher. For example, whole genome sequencing data collected from human blood cell DNA estimates a mutation rate of 1.1×10^{-8} per base per generation, corresponding to about 70 new mutations in each human diploid genome [18]. Depending on the nature of the change, mutations may have deleterious, neutral, or, rarely, beneficial effects on the organism. Mutations lead to sequence variation and are ultimately the basis of genome evolution and diversification for those carried through the germline. Although mutations in somatic cells are not passed on to the next generation, they can lead to diseases, including cancer, and affect the survival of the individual.

There are various forms of DNA mutations, from single nucleotide substitutions, to small insertions/deletions (or indels), to structural variations (SVs) that involve larger genomic regions. Among these different types of mutations, single nucleotide substitutions, also called point mutations, are the most common. These substitutions can be either transitions or transversions. Transitions involve the substitution of a purine for the other purine (i.e., A↔G) or a pyrimidine for the other pyrimidine (i.e., C↔T). Transversions, on the other hand, involve the substitution of a purine for a pyrimidine or vice versa. Theoretically, there are more combinations of transversions than transitions, but due to the nature of the underlying biochemical processes, transitions actually occur more frequently than transversions. If a single nucleotide substitution takes place in a protein-coding region, it might lead to a change in amino acid coding. If it causes the substitution of one amino acid for another, it is a missense mutation, which may lead to a change of protein function. If it introduces a stop codon and as a result leads to the generation of a truncated protein, it is a nonsense mutation. Both the missense and nonsense mutations are nonsynonymous mutations. If it does not change the coded amino acid due to the redundancy in the genetic code, it is a synonymous mutation and has no effect on protein function. Because of its common occurrence, single nucleotide variation (SNV) is the most frequently observed sequence variation. If an SNV is commonly observed in a population, it is called a single nucleotide polymorphism (SNP). More than 100 million SNPs in the human genome have been cataloged. Because of their high density in the genome, SNPs are often used as flagging markers to cover the entire genome in high

resolution when scanning for genomic region(s) that are associated with a phenotype or disease of interest.

Besides single nucleotide substitutions, indels are another common type of mutation. Most indels involve small numbers of nucleotides. In protein-coding regions, small indels lead to the shift of ORF (unless the number of nucleotides involved is a multiple of three), resulting in the formation of a vastly different protein product. Indels that involve large regions (>1 kb in size) lead to alterations of genomic structure and are usually considered as a form of SV. Besides large indels, SVs also include inversions, translocations, or duplications that involve alterations of larger DNA regions (typically >1 kb). Copy number variation (CNV) is a subcategory of SV, usually caused by large indel or segmental duplication. Although they affect larger genomic region(s) and some lead to observable phenotypic changes or diseases, many CNVs, or SVs in general, have no detectable effects. The frequency of SVs in the genome was previously underestimated due to technological limitations. The emergence of NGS has greatly enabled SV detection, which has led to the realization of its wide existence [19].

2.7 Genome Evolution

The spontaneous mutations that lead to sequence variation and polymorphism in a population are also the fundamental force behind the evolution of genomes and eventually the Darwinian evolution of the host organisms. Gradual sequence change and diversification of early genomes, over billions of years, have evolved into the extremely large number of genomes that had existed or are functioning in varying complexity today. In this process, existing DNA sequences are constantly modified, duplicated, and reshuffled. Most mutations in protein-coding or regulatory sequences disrupt the protein's normal function or alter its amount in cells, causing cellular dysfunction and affecting organismal survival. Under rare conditions, however, a mutation can improve existing protein function or lead to the emergence of new functions. If such a mutation offers its host a competitive advantage, it is more likely to be selected and passed on to future generations.

Gene duplication provides another major mechanism for genome evolution. If a genomic region containing one or multiple gene(s) is duplicated resulting in the formation of an SV, the duplicated region is not under selection pressure and therefore becomes substrate for sequence divergence and new gene formation. Although there are other ways of adding new genetic information to a genome such as interspecies gene transfer, DNA duplication is believed to be a major source of new genetic information generation. Gene duplication often leads to the formation of gene families. Genes in the same family are homologous, but each member has its specific function and

expression pattern. As an example, in the human genome there are 339 genes in the olfactory receptor gene family. Odor perception starts with the binding of odorant molecules to olfactory receptors located on olfactory neurons inside the nose epithelium. To detect different odorants, a combination of different olfactory receptors that are coded by genes in this family is required. Based on their sequence homology, members of this large family can be even further grouped into different subfamilies [20]. The existence of pseudogenes in the genome is another result of gene duplication. After duplication, some genes may lose their function and become inactive from additional mutation. Pseudogenes may also be formed in the absence of duplication by the disabling of a functional gene from mutation. A pseudogene called *GULO* mapped to the human chromosome 8p21 provides such an example. The functional *GULO* gene in other organisms codes for an enzyme that catalyzes the last step of ascorbic acid (vitamin C) biosynthesis. This gene is knocked out in primates, including humans, and becomes a pseudogene. As a result, we have to get this essential vitamin from food. The inactivation of this gene is possibly due to the insertion to the gene's coding sequence of a retrotransposon-type repetitive sequence called Alu element [21].

DNA recombination, or reshuffling of DNA sequences, also plays an important role in genome evolution. Although it does not create new genetic information, by breaking existing DNA sequences and rejoining them, DNA recombination changes the linkage relationships between different genes and other important regulatory sequences. Without recombination, once a harmful mutation is formed in a gene, the mutated gene will be permanently linked to other nearby functional genes, and it becomes impossible to regroup all the functional genes into the same DNA molecule. Through this regrouping, DNA recombination makes it possible to avoid gradual accumulation of harmful gene mutations. Most DNA recombination events happen during meiosis in the formation of gametes (sperm or eggs) as part of sexual reproduction.

2.8 Epigenome and DNA Methylation

Besides the regulatory DNA sequences introduced earlier, chemical modifications of specific nucleotides in the genome, like the acetylation and deacetylation of histones, offer another layer of regulation on genetic activities. Since they provide additional genetic activity regulation, these chemical modifications on DNA and histones constitute the epigenome. Methylation of the fifth carbon on cytosine (5-methylcytosine, or 5mC) is currently the most studied epigenomic modification in many organisms. Enzymatically this methylation is carried out and maintained by DNA methyltransferases (three identified in mammals: DNMT1, DNMT3A, and DNMT3B). The

cytosines that undergo methylation can occur in three different sequence contexts—CpG, CHG, and CHH (H can be A, G, or T)—each involving different pathways [22]. Most methylated cytosines exist in the CpG context, where the methylation reduces gene expression through recruiting gene silencing proteins or preventing transcription factors from binding to the DNA. The methylation of cytosines in this context also affects nucleosome positioning and chromatin remodeling, as methyl-CpG binding domain (MBD) proteins that specifically bind to 5mC at CpG sites can recruit histone-modifying proteins and those in the chromatin remodeling complex [23]. The effects of cytosine methylation in the CHG and CHH contexts are less clear, but available data seems to suggest that they may play a regulatory role in repetitive regions [24].

Just like deacetylation counteracts the effects of acetylation in histones, demethylation of cytosines should be similarly important to reverse the effects of 5mC when the methylation is no longer needed. It is until recently that the steps involved in the cytosine demethylation process begin to be understood. In this process, the 5mC is first oxidized to 5-hydroxymethylcytosine (5hmC), and then to 5-formylcytosine (5fC) and 5-carboxylcytosine (5caC) in mammals. These oxidative conversions are catalyzed by enzyme systems such as the TET family proteins. The subsequent base excision repair of 5fC/5caC by an enzyme called TDG, or 5mC directly by the glycosylase enzyme in plants, completes the DNA demethylation process [25]. Compared to 5mC, the levels of these demethylation intermediate products are detected to be much lower in most cells (except that 5hmC has been found to be relatively abundant in embryonic stem cells and in the brain).

Different from the genome, which is static, the epigenome is dynamic and changes with environmental conditions. These dynamically changing epigenomic modifications regulate gene expression and thereby play important roles in embryonic development, cell differentiation, stem cell pluripotency, genomic imprinting, and genome stability. In accordance with their regulatory functions, these modifications are highly site specific. To study where cytosine methylations take place in the genome, multiple NGS-based approaches, which will be detailed in Chapter 12, have been developed and widely applied to epigenomics studies. Methodological development for the study of cytosine demethylation is currently still at an early stage.

2.9 Genome Sequencing and Disease Risk

The wide accessibility of DNA sequences, largely fueled by the rapid development of new sequencing technologies, has uncovered extensive sequence variation in individual genomes within a population. The extensiveness in sequence variation was not envisioned in early days of genetics, not even

when the Human Genome Project was completed in 2003. This has gradually led to a paradigm shift in disease diagnosis and prevention. As a result, the public becomes more aware of the role of individual genomic makeup in disease development and predisposition. In addition, the easier accessibility to our DNA sequence has further prompted us to look into our genome and use that information for preemptive disease prevention. The declining cost of genome sequencing has also enabled the biomedical community to dig deeper into the genomic underpinnings of diseases, by unraveling the linkage between sequence polymorphism in the genome and disease incidence. Following is a brief overview of the major categories of human diseases that have an intimate connection with DNA mutation, polymorphism, genome structure, and epigenomic abnormality.

2.9.1 Mendelian (Single-Gene) Diseases

The simplest form of hereditary diseases is caused by mutation(s) in a single gene, and therefore called monogenic or Mendelian diseases. For example, sickle cell anemia is caused by a mutation in the *HBB* gene located on the human chromosome 11. This gene codes for the β subunit of hemoglobin, an important oxygen-carrying protein in the blood. A mutation of this gene leads to the replacement of the sixth amino acid, glutamic acid, with another amino acid valine in the coded protein. This change of a single amino acid causes conformational change of the protein, leading to the generation of sickle-shaped blood cells that die prematurely. This disease is recessive, meaning that it only appears when both copies (or alleles) of the gene carry the mutation. In dominant diseases, however, one mutant allele is enough to cause sickness. Huntington's disease, a neurodegenerative disease that leads to gradual loss of mental faculties and physical control, is such a dominant single-gene disease. It is caused by mutation in a gene called *HTT* on the human chromosome 4, coding for a protein called huntingtin. The involved mutation is an expanded and unstable trinucleotide (CAG) repeat. Individuals carrying one copy of the mutant *HTT* gene usually develop the disease later in life.

2.9.2 Complex Diseases That Involve Multiple Genes

Most common diseases, including heart disease, diabetes, hypertension, obesity, and Alzheimer's disease (AD), are caused by multiple genes. In the case of AD, while its familial or early-onset form can be attributed to one of three genes (*APP*, *PSEN1*, and *PSEN2*), the most common form, sporadic AD, involves a large number of genes [26]. In this type of complex diseases, the contribution of each gene is modest, and it is the combined effects of mutations in these genes that predispose an individual to these diseases. Besides genetic factors, lifestyle and environmental factors often also play a role in these complex diseases. For example, a history of head trauma, lack

of mentally stimulating activities, and high cholesterol levels are all risk factors for developing AD. Because of the number of genes involved and their interactions with nongenetic factors, complex multigene diseases are more challenging to study than single-gene diseases.

2.9.3 Diseases Caused by Genome Instability

Aside from the gene-centered disease models introduced earlier, diseases can also occur as consequences of large-scale genomic changes such as rearrangement of large genomic regions, alterations of chromosome number, and general genome instability. For example, when a genome becomes unstable in an organism, it can cause congenital developmental defects, tumorigenesis, premature aging, and so forth. Dysfunction in genome maintenance, such as DNA repair and chromosome segregation, can lead to genome instability. Fanconi anemia, a disease caused by genome instability, is characterized by growth retardation, congenital malformation, bone marrow failure, high cancer risk, and premature aging. The genome instability in this disease is caused by mutations in a cluster of DNA repair genes, and manifested by increased mutation rates, cell cycle disturbance, chromosomal breakage, and extreme sensitivity to reactive oxygen species and other DNA damaging agents.

Cancer, to a large degree, is also caused by genome instability. This can be hinted by the fact that two well-known high-risk cancer genes, *BRCA1* and *BRCA2*, are both DNA damage repair genes. Mutations in the two genes greatly increase the susceptibility to tumorigenesis, such as breast and ovarian cancers. In general, many cancers are characterized by chromosomal aberrations and genome structural changes, involving deletion, duplication, and rearrangement of large genomic regions. The fact that genome instability is intimately related to major aspects of cancer cells, such as cell cycle regulation and DNA damage repair, also points to the important role of genome instability in cancer development.

2.9.4 Epigenomic/Epigenetic Diseases

Besides gene mutations and genome instability, abnormal epigenomic/epigenetic patterns can also lead to diseases. Examples of diseases in this category include fragile X syndrome, ICF syndrome, Rett syndrome, and Rubinstein-Taybi syndrome. In ICF syndrome, for example, the gene *DNMT3B* is mutated leading to the deficiency of DNA methyltransferase 3B. Patients afflicted with this disease invariably have DNA hypomethylation, and have symptoms such as facial anomaly, immunodeficiency, and chromosome instability. Cancer, as a genome disease that is caused by more than one genetic/genomic factor, is also characterized by abnormal DNA methylation, including both hypermethylation and hypomethylation. The hypermethylation is commonly observed in the promoter CpG islands of tumor

suppressor genes [27], which leads to their suppressed transcription. The hypomethylation is mostly located in highly repetitive sequences, including tandem repeats in the centromere and interspersed repeats. This lowered DNA methylation has been suggested to play a role in promoting chromosomal rearrangements and genome instability [28].

3

RNA: The Transcribed Sequence

3.1 RNA as the Messenger

The blueprint of life is written in DNA, but almost all life processes are executed by proteins. To convert the information coded in the DNA into the wide array of proteins in each cell, segments of DNA sequence in the genome must be copied into messenger RNAs (mRNAs) first. The transcribed nucleotide sequences in the mRNAs are then translated into proteins through an information decoding process carried out by ribosomes. Because of the intermediary role played by mRNAs between DNA and proteins, the composition of mRNAs in a cell or population of cells—the transcriptome—is often used to study cellular processes and functions. Unlike the genome, which is mostly static and the same for every cell in an organism, the transcriptome is dynamically regulated and therefore can be used as a proxy of cellular functional status.

3.2 The Molecular Structure of RNA

Structurally, RNA is closely related to DNA and also made of nucleotides. The nucleotides that make up the RNA molecule are slightly different from those of DNA. Instead of deoxyribose, its five-carbon sugar moiety is a ribose. Among the four nucleobases, uracil (U) is used in place of thymine (T), but the remaining three (A, C, and G) are the same. Unlike the double-stranded structure of DNA, RNA molecules are single stranded, which gives them great flexibility. If intramolecular sequence complementarity exists between two regions of a single RNA molecule, this structural flexibility allows the regions to bend back on each other and form intramolecular interactions.

As a result of its structural flexibility and internal sequence complementarity, an RNA molecule can assume secondary structures, such as hairpins and stem-loops, and tertiary structures depending on its specific sequence. These

structures can sometimes afford them special chemical properties in cells. For example, some nonmessenger RNAs can catalyze chemical reactions like protein enzymes and are therefore called RNA enzymes (or ribozymes; more details in Section 3.4.1). Some RNA molecules may assume tertiary structures that enable them to bind to other small molecules such as ligands or large molecules such as RNA-binding proteins. For mRNAs, their structures may also be important for various steps of their life cycle (see next section for details). One example of this is riboswitch, a region in some mRNAs that binds to small molecule ligands such as metabolites or ions, and thereby regulates their transcription, translation, or splicing via changes in RNA structure upon ligand binding [29]. Binding of proteins to mRNA elements like those located in the 3′ untranslated region (UTR) can also induce structural changes of these elements and affect mRNA translation [30]. Transport of mRNAs to specific cellular locations, such as distal dendritic regions of a neuron, also requires the mRNAs to assume specific structures for RNA-binding proteins to bind as a prerequisite of the transport process. To study structures of individual RNAs, computational prediction and experimental approaches, such as RNA fingerprinting that uses a variety of chemical and enzymatic probes, have been the classic methods. With the advent of RNA sequencing based on next-generation sequencing (NGS), that is, RNA-Seq, transcriptome-wide RNA structural mapping is enabled when integrated with these classic approaches [31].

3.3 Generation, Processing, and Turnover of RNA as a Messenger

When a protein is needed in a cell, its coding gene is first transcribed to mRNA, which is then used as the template to translate to the requisite protein. In a prokaryotic cell, mRNA transcription is immediately followed by protein translation. In a eukaryotic cell, the information flow from DNA to protein through mRNA is more complex, because the two steps of transcription and translation are physically separated and eukaryotic genes contain introns that need to be removed before translation. In the eukaryotic system, initial transcript (also called primary transcript) is first synthesized from the DNA template and then processed, including intron removal, to produce mature mRNA in the nucleus. Then the mRNA is transported from the nucleus to the cytoplasm for translation. When they are no longer needed, the mRNAs are degraded and recycled by the cell. It should also be noted that the transcription process generates a number of mRNA copies from a gene, and the copy number varies from condition to condition and from gene to gene depending on cellular functional status.

3.3.1 DNA Template

To initiate transcription, a gene's DNA sequence is first exposed through altering its packing state. In order to transcribe the DNA sequence, the two DNA strands in the region are first unwound, and only one strand is used as the template strand for transcription. Since it is complementary to the RNA transcript in base pairing (A, C, G, and T in the DNA template are transcribed to U, G, C, and A, respectively, in the RNA transcript), this DNA template strand is also called the antisense or negative (–) strand (Figure 3.1). The other DNA strand has the same sequence as the mRNA (except with T's in DNA being replaced with U's in RNA) and is called the coding, sense, or positive (+) strand. It should be noted that either strand of the genomic DNA can be potentially used as the template, and which strand is used as the template for a gene depends on the orientation of the gene along the DNA. It should also be noted that the triplet nucleotide genetic code that determines how amino acids are assembled in proteins refers to the triplet sequence in the mRNA sequence.

3.3.2 Transcription of Prokaryotic Genes

RNA polymerase catalyzes the transcription of RNA from its DNA template. In prokaryotic cells, there is only one type of RNA polymerase. The prokaryotic RNA polymerase holoenzyme contains a core enzyme of five subunits that catalyzes RNA transcription from a DNA template, and another subunit called the sigma factor that is required for initiation of transcription. The sigma factor initiates the process by enabling binding of the core enzyme to the promoter region and guiding it to the transcription start site (TSS). Promoter is the region upstream of the protein-coding sequence of a gene or an operon. Prokaryotic promoters share some core sequence elements, such as the motif centered at 10 nucleotides upstream of the TSS with the

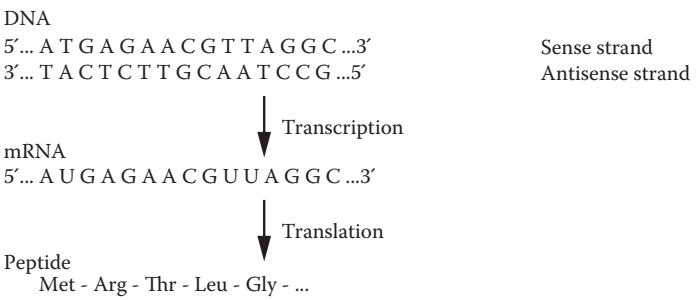

FIGURE 3.1
How the two strands of DNA template match the transcribed mRNA in sequence, and the genetic code in mRNA sequence corresponds to the peptide amino acid sequence.

consensus sequence TATAAT. Once reaching the TSS, the sigma factor disassociates from the core enzyme. The core RNA polymerase, unlike DNA polymerase, does not need a primer, but otherwise the enzyme catalyzes the attachment of nucleotides to the nascent RNA molecule one at a time in the 5′→3′ direction. At a speed of approximately 30 nucleotides/second, the RNA polymerase slides through the DNA template carrying the elongating RNA molecule.

Although the attachment of new nucleotides to the elongating RNA is based on base pairing with the DNA template, the new elongating RNA does not remain associated with the template DNA via hydrogen bonding. On the same template, multiple copies of RNA transcripts can be simultaneously synthesized by multiple RNA polymerases one after another. During transcript elongation, these polymerases hold on tightly to the template and do not disassociate from the template until the stop signal is transcribed. The stop signal is provided by a segment of palindromic sequence located at the end of the transcribed sequence. Right after transcription, the inherent self-complementarity in the palindromic sequence leads to the spontaneous formation of a hairpin structure. An additional stop signal is also provided by a string of four or more uracil residues after the hairpin structure, which forms weak associations with the complementary A's on the DNA template. The hairpin structure pauses further elongation of the transcript, and the weak associations between the U's on the RNA and the A's on the DNA dissociate the enzyme and the transcript from the template.

Regulation of prokaryotic transcription is conferred by promoters and protein factors such as repressors and activators. Promoter strength, that is, the number of transcription initiation events per unit time, varies widely in different operons. For example, in *E. coli*, genes in operons with weak promoters can be transcribed once in 10 minutes, while those with strong promoters can be transcribed 300 times in the same amount of time. The strength of an operon's promoter is based on the host cell's demand for its protein products and dictated by its sequence. Specific protein factors may also regulate gene transcription. Repressors, the best known among these factors, prevent RNA polymerase from initiating transcription through binding to an intervening sequence between the promoter and TSS called an operator. Activators exert an opposite effect and induce higher levels of transcription. The sigma factor, being the initiation factor of the prokaryotic RNA polymerase, provides another mechanism for regulation. There are different forms of this factor in prokaryotic cells, each of which mediates sequence-specific transcription. Differential use of these sigma factors, therefore, provides another level of transcriptional regulation in prokaryotic cells.

3.3.3 Initial Transcription of Pre-mRNA from Eukaryotic Genes

In eukaryotic cells there are three types of RNA polymerases, among which RNA polymerase II transcribes protein-coding genes, while RNA

polymerases I and III transcribe ribosomal RNA (rRNA), transfer RNA (tRNA), and various types of small RNAs. Transcription in eukaryotic cells is in general much more sophisticated, because of the highly compressed packaging of chromosomal DNA, the complex structure of eukaryotic genes, and intricate regulation by multiple factors. Prior to transcription, the highly compressed DNA in the chromatin needs to be uncompressed and the gene sequence exposed for access by RNA polymerase.

To perform the transcription of protein-coding genes, besides RNA polymerase II, a variety of other proteins in the nucleus are also required, including transcription factors and coactivators. Transcription factors include general and specific transcription factors. General transcription factors, such as TFIIA, TFIIB, and TFIID, are required in all transcription initiation. Their function is to position the RNA polymerase at the promoter region and unwind the template DNA strands for transcription. Specific transcription factors, which are detailed next, provide key regulatory function to the transcription initiation process. Coactivators bring together all requisite transcription factors to form the transcription initiation complex. Once transcription is initiated, most of the protein factors in the complex are released and the RNA elongation process is carried out by RNA polymerase II in a manner similar to what occurs in prokaryotic cells. The termination of the elongation process in eukaryotic cells is provided by the signal sequence AAUAAA, which also serves as the signal for cleavage of the transcribed RNA to generate the 3′ end and for polyadenylation (see Section 3.3.4). After completion of the transcription process, the transcript contains both exons and introns, and is called the primary transcript or pre-mRNA.

During RNA transcript elongation in both the eukaryotic and prokaryotic systems, like in DNA replication by the DNA polymerase, there is a certain probability of introducing mismatched nucleotides and therefore errors. For proofreading, the prokaryotic and eukaryotic RNA polymerases have 3′→5′ exonuclease activity. If a wrong nucleotide is added to the elongating RNA chain, the RNA polymerase will backtrack and correct the error. Because of this activity, the overall error rate of the transcriptional process in both systems is estimated to be 10^{-4} to 10^{-5} per base [32]. Although this is higher than the DNA mutation rate, the transcriptional errors are seldom harmful, because there are multiple copies for each transcript, and transcripts carrying premature stop codons are quickly removed by a process called nonsense-mediated decay.

Besides the step of gene sequence exposure through histone modifications and chromatin remodeling, the eukaryotic gene transcription process is mostly regulated at the initiation step through the use of specific transcription factors. As a large group of DNA-interacting proteins (Chapter 2), these transcription factors bind to specific sequence elements in the promoter region of genes, through which they help assemble general transcription factors and the RNA polymerase into the transcription initiation complex. In addition, specific transcription factors may also bind to specific regulatory

sequences at distant locations that are called enhancers or *cis*-regulatory modules. Different from transcription factor binding sites in the promoter regions, enhancers function independent of sequence orientation and from a distance as far as megabases away from the regulated gene, and are sometimes embedded in intergenic regions that otherwise have no known function. Having a significant effect on gene transcription, enhancers exert their regulatory function by DNA looping, which brings enhancer and promoter sequences together affecting formation of the transcription initiation complex. The binding of specific transcription factors to enhancers can have a stimulatory, or inhibitory (through the recruitment of repressors), effect on gene transcription. In general, the transcription of a gene is often regulated by multiple specific transcription factors, and the combined signal input from these transcription factors determines whether the gene will be transcribed, and if yes, at what level. A particular transcription factor can also bind to multiple genomic sites, coordinating the transcription of functionally related genes. NGS-based approaches, such as ChIP-Seq (Chapter 11), are often used to locate the binding sites of specific transcription factors across the genome.

3.3.4 Maturation of mRNA from Pre-mRNA

In prokaryotic cells, there is no posttranscription RNA processing, and transcripts are immediately ready for protein translation after transcription. In fact, while mRNAs are still being transcribed, ribosomes are already binding to the transcribed portions of the elongating mRNAs synthesizing peptides. In eukaryotic cells, however, primary transcripts undergo several steps of processing in the nucleus to become mature mRNAs. These steps are (1) capping at the 5′ end, (2) splicing of exons and introns, and (3) addition of a poly-A tail at the 3′ end.

The first step, adding a methylated guanosine triphosphate cap to the 5′ end of nascent pre-mRNAs, takes place shortly after the initiation of transcription when the RNA chains are still less than 30 nucleotides long. This step is carried out by adding a guanine group to the 5′ end of the transcripts, followed by methylation of the group. This cap structure marks the transcripts for subsequent transport to the cytoplasm, protects them from degradation, and promotes efficient initiation of protein translation. Once formed, the cap is bound by a protein complex called cap-binding complex.

The second step, splicing of exons and introns, is the most complicated of the three steps. As introns are noncoding intervening sequences, they need to be spliced out while exons are retained to generate mature mRNAs. The molecular machinery that carries out the splicing, called the spliceosome, is assembled from as many as 300 proteins and 5 small nuclear RNAs (snRNAs). The spliceosome identifies and removes introns from primary transcripts, using three positions within each intron: the 5′ end (starts with the consensus sequence 5′-GU, serving as the splice donor), the 3′ end (ends with the consensus sequence AG-3′, as the splice acceptor), and the branch

point, which starts around 30 nucleotides upstream of the splicing acceptor and contains an AU-rich region. The actual excision of each intron and the concomitant joining of the two neighboring exons are a three-step process: (1) cleavage at the 5′ end splice donor site; (2) attachment of the cleaved splice donor site to the branch point to form a lariat or loop structure; and (3) cleavage at the 3′ end splicing acceptor site to release the intron and join the two exons.

Beyond simply removing introns from primary transcripts, the splicing process also employs differential use of exons, and sometimes even includes some introns, to create multiple mature mRNA forms from the same primary transcript. This differential splicing, also called alternative splicing (Figure 3.2), provides an additional regulatory step in the production of mRNA populations. When it was first reported in 1980, alternative splicing was considered to be an exception rather than the norm. Currently available data has shown that primary transcripts from essentially all multiexon genes are alternatively spliced [33,34]. The biological significance of alternative splicing is obvious: by enabling production of multiple mRNAs and thereby proteins from the same gene, it greatly augments protein and, consequently, functional diversity in an organism without significantly increasing the number of genes in the genome, and offers explanation to why more evolved organisms do not contain many more genes in their genomes (see Chapter 2, Table 2.1).

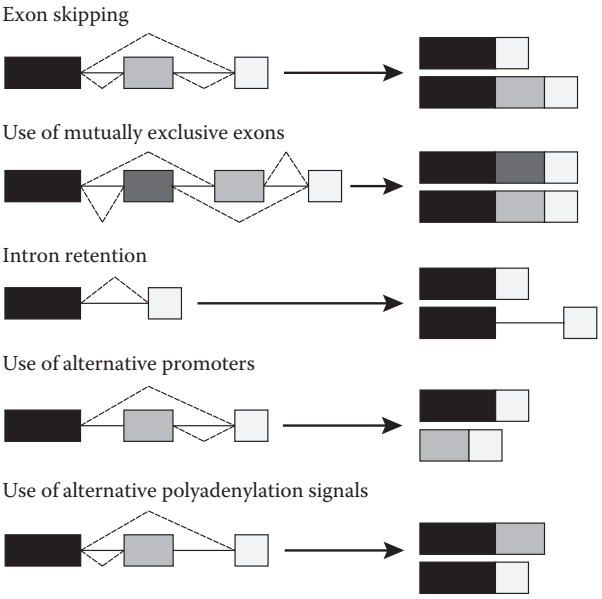

FIGURE 3.2
Varying forms of RNA transcript splicing.

In the third step, once the new primary transcript passes the termination signal sequence, it is bound by several termination-related proteins. One of the proteins cleaves the RNA at a short distance downstream of the termination signal to generate the 3′ end. This is followed by a polyadenylation step that adds 50 to 200 A's to the 3′ end by an enzyme called poly-A polymerase. This poly-A tail, like the 5′ end cap, increases the stability of the resulting mRNA. This tail is bound and protected by a poly(A)-binding protein, which also promotes its transport to the cytoplasm.

Besides these three major constitutive processing steps, some transcripts may undergo additional processing steps. RNA editing, although considered to be rare, is among the best known of these steps. RNA editing refers to the change in RNA nucleotide sequence after it is transcribed. The most common types of RNA editing are conversions from A to I (inosine, read as G during translation), which are catalyzed by enzymes such as ADARs (adenosine deaminases that act on RNA), or from C to U, catalyzed by cytidine deaminases. As a result of these conversions, an edited RNA transcript no longer fully matches the sequence on the template DNA. RNA editing has the potential to change genetic codons, introduce new or remove existing stop codons, or alter splicing sites [35]. Evidence shows that RNA editing and other RNA processing events such as splicing can be coordinated [36].

3.3.5 Transport and Localization

After maturation, mRNAs need to be exported out of the nucleus to the cytoplasm for protein translation. While allowing mature mRNAs to be transported out, the nucleus keeps unprocessed or partially processed transcripts, as well as processed side products like removed introns, inside the nucleus. To move across the nucleus envelope through the nuclear pore complexes, mature mRNAs are packaged into large ribonucleoprotein (RNP) complexes. Once in the cytoplasm, many mRNA species can be used to start synthesizing proteins right away. As the cytoplasm is a crowded place, they may randomly drift in the cytoplasm while translating. Some translations, however, take place at highly localized sites. For example, in neurons some mRNAs are required to be transported to distal dendritic regions for translation. Local protein translation at such target sites has been known to underlie important biological functions, such as synaptic plasticity that underlies learning and memory [37]. In order to transfer mRNAs to these special locations, the mRNAs bind to special proteins to form mRNA-protein complexes, which are then attached to protein motors to move along cytoskeletal tracks.

3.3.6 Stability and Decay

Steady-state mRNA concentrations, which are the detection target of transcriptomic analyses such as RNA-Seq, are determined by not only rates of mRNA production but also their decay. In general, prokaryotic mRNAs are

unstable and quickly degraded by endoribonucleases and exoribonucleases after transcription. As a result, most of them are short lived and the average prokaryotic mRNA half-life, that is, the amount of time to have half of the mRNAs degraded, is under 10 minutes [38]. This high turnover rate allows prokaryotic cells to quickly respond to environmental changes by altering transcription. In comparison, eukaryotic mRNAs are in general more stable and have a longer average half-life of 7 to 10 hours [39,40]. As a general rule, mRNAs for regulatory or inducible proteins, such as transcription factors or stress response proteins, tend to have shorter half-lives (e.g., less than 30 minutes), while those for housekeeping proteins, such as those of metabolism and cellular structure, have long half-lives (e.g., days). The stability and half-lives of mRNAs are also regulatable based on developmental stage or environmental factors. For example, the stability and half-lives of mRNAs of muscle-specific transcription factors, such as myogenin and myoD, are the highest during muscle differentiation but quickly decline once the differentiation is completed [41].

The regulation of eukaryotic mRNA degradation is not well understood but has been known to involve interactions between some sequence elements on mRNAs and protein as well as small RNA factors. One example of the mRNA stability regulatory sequences is the AU-rich element, a region on the 3′ untranslated region of many short-lived mRNAs that, as the name suggests, are rich in adenines and uridines. Many protein factors interact with this element to modulate mRNA turnover, such as the AU-rich binding factor 1 (or AUF1). Small RNAs, including microRNA (miRNA), small interfering RNA (siRNA), and Piwi-interacting RNA (piRNA), are also important regulators of mRNA stability and degradation (see Section 3.4.4 for details). P-bodies (processing bodies), a granular structure in eukaryotic cells, are the focal point of mRNA turnover mediated by protein and small RNA factors [42].

Most eukaryotic mRNA decay starts with deadenylation at the 3′ end, that is, removal of the poly-A tail by deadenylase. The deadenylation then leads to mRNA degradation through two alternative mechanisms. One mechanism is through decapping of mRNA at the 5′ end, which leaves the mRNA vulnerable to degradation by 5′→3′ exoribonuclease. The other mechanism is direct 3′→5′ decay from the tail end, which is carried out by a multiprotein complex called exosome. Besides these major deadenylation-dependent mRNA decay pathways, there are also other pathways that do not rely on deadenylation [43].

3.3.7 Major Steps of mRNA Transcript Level Regulation

As indicated earlier, the regulation of both prokaryotic and eukaryotic transcription is mostly applied at the initiation step, and this regulation is heavily dependent on specific protein–DNA interactions. In the prokaryotic system, besides promoter strength, the regulation of transcriptional initiation is provided by protein factors including repressors and activators, both of which

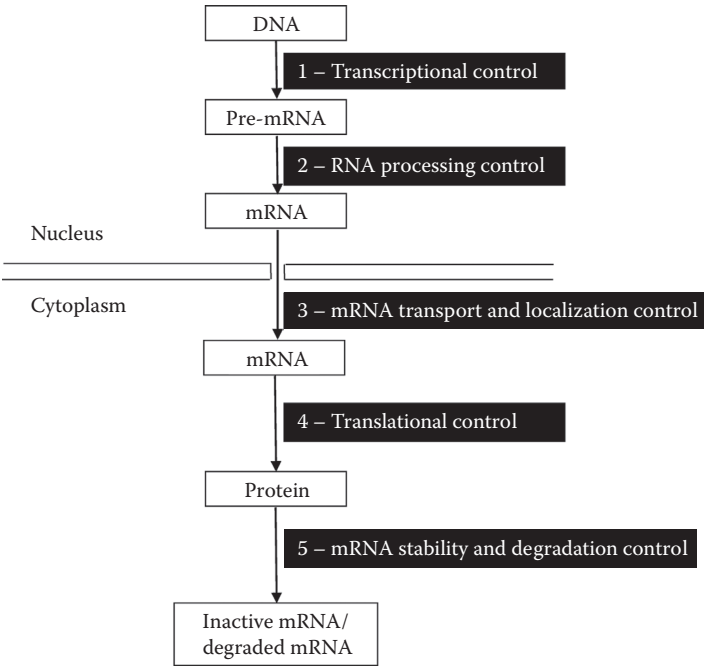

FIGURE 3.3
The regulation of eukaryotic gene expression at multiple levels.

bind to specific promoters sequences. In the eukaryotic system, specific transcription factors that bind to specific sequences in promoters and/or enhancers offer most of the regulation. In addition, prior to the engagement of transcription initiation complex, gene sequence access is regulated through histone modification and chromatin remodeling. Since the generation of mRNA in the eukaryotic system is a multistep process, regulatory mechanisms are also applied at subsequent steps (Figure 3.3). During mRNA maturation, regulation of exon and intron splicing leads to generation of alternative splicing variants. Trafficking of mRNAs to localized cellular domains provides additional regulatory mechanism for some genes [44]. Equally important in determining steady-state mRNA levels, mRNA decay is another important but less studied step upon which regulation is also exerted.

3.4 RNA Is More Than a Messenger

Despite their apparent indispensability, mRNAs constitute only about 5% of total cellular RNA. Besides rRNAs and tRNAs, there is a rapidly growing

number of non-protein-coding RNAs that play important roles in regulating protein-coding genes or carry out essential cellular functions. These noncoding RNAs include miRNAs, piRNAs, siRNAs, snRNAs, small nucleolar RNAs (snoRNAs), long-noncoding RNAs (lncRNAs), and RNAs that function as catalysts (ribozymes). Some of these noncoding RNAs have been extensively studied, such as the ribozymes, the discovery of which won the 1989 Nobel Prize in Chemistry and has led to the "RNA world" hypothesis. Based on this hypothesis, early life forms were solely based on RNA, and DNA and protein evolved later. The rRNAs, tRNAs, and ribozymes are thought by this hypothesis as evolutionary remnants of the original RNA world [45]. The functional importance and diversity of other noncoding RNAs, such as the many forms of small RNAs and lncRNAs, are still in the process of being fully appreciated, because they were discovered more recently. However, because of their wide presence and importance, the 2006 Nobel Prize in Physiology or Medicine was awarded to the discovery of RNA interference (RNAi) by small RNAs. Due to the diverse and important roles that noncoding RNAs play in cells, RNA is not treated as simply a messenger any more.

3.4.1 Ribozyme

Similar to proteins, RNAs can form complicated three-dimensional structures, and some RNA molecules carry out catalytic functions. These catalytic RNAs are called ribozymes. A classic example of a ribozyme is one type of intron called group I intron, which splices itself out of the pre-mRNA that contains it. This self-splicing process, involving two transesterification steps, is not catalyzed by any protein. A group I intron is about 400 nucleotides in length and mostly found in organelles, bacteria, and the nucleus of lower eukaryotes. When a precursor RNA that contains a group I intron is incubated in a test tube, the intron splices itself out of the precursor RNA autonomously. Despite variations in their internal sequences, all group I introns share a characteristic spatial structure, which provides active sites for catalyzing the two steps. Another example of ribozyme is the 23S rRNA contained in the large subunit of the prokaryotic ribosome. This rRNA catalyzes the peptide bond formation between an incoming amino acid and the existing peptide chain. Although the large subunit contains more than 30 proteins, rRNA is the catalytic component, while the proteins only provide structural support and stabilization [46].

Also similar to protein catalysts, the dynamics of the reactions catalyzed by ribozymes follows the same characteristics as those of protein enzyme-catalyzed reactions, which are usually described by the Michaelis-Menten equation. Further similarities of ribozymes to protein enzymes include that ribozyme activity can also be regulated by ligands, usually small molecules, the binding of which leads to structural change in the ribozyme. For instance, a ribozyme may contain a riboswitch, which as part of the ribozyme can bind to a ligand to turn on or off the ribozyme activity.

3.4.2 snRNA and snoRNA

Although a group I intron can self-splice, most pre-mRNA introns are not of this type and need the spliceosome for splicing. The spliceosome, even larger than the ribosome in size, contains five snRNAs (U1, U2, U4, U5, and U6), and a large number of proteins. The splicing process heavily depends on the interactions between these snRNAs and pre-mRNAs. For example, to initiate splicing, U1 interacts with the 5′ splice donor site and U2 with the branch site via base pairing. Later in the process, U6 binds to the 5′ splice site prior to its cleavage. Although the spliceosome contains a large number of proteins, the roles played by these snRNAs are indispensable.

Similarly, snoRNAs are indispensable for pre-rRNA processing. The eukaryotic ribosome contains four rRNAs—28S, 18S, 5.8S, and 5S—with the first three initially transcribed into a single large rRNA precursor. To generate the three rRNAs, the precursor rRNA needs to be first chemically modified and then cleaved. The chemical modification includes methylation at over 100 nucleotides and isomerization of uridine at another 100 sites. The snoRNAs are required in this process to identify the specific sites for modification. There are many different types of snoRNAs, each of which can form a complementary region with the precursor rRNA via base pairing. These duplex regions are then recognized as targets for modification.

3.4.3 RNA for Telomere Replication

Located at the tips of a chromosome, telomeres seal the ends of chromosomal DNA. Without telomeres, the integrity of chromosomes would be compromised since DNA repair enzymes would recognize the DNA termini as break points. Inside each telomeric region is a long string of highly repetitive DNA sequences. Normally, shortening of telomere length occurs with each chromosome replication, since chromosomal DNA duplication enzymes cannot reach to the very ends of the DNA (the end replication problem). To prevent this problem in germ cells and stem cells, an enzyme called telomerase is responsible for replenishing the telomeric region. The telomerase is a large complex comprising an RNA component, which serves as a template for the repeat sequence, and a catalytic protein component (reverse transcriptase), which uses the RNA template to synthesize the repetitive telomeric DNA sequence. The telomerase activity is usually turned off or at very low levels in most somatic cells. Therefore, these cells can only divide a certain number of times before reaching senescence due to the gradual shortening of the telomere.

3.4.4 RNAi and Small Noncoding RNAs

RNAi, as a cellular mechanism that uses small RNAs to silence gene expression, offers an excellent illustration of the significance of noncoding RNAs

in the regulation of protein-coding genes. RNAi achieves gene silencing by suppressing mRNA translation, degrading mRNAs, or inhibiting gene transcription [47]. As a native gene regulation mechanism in a wide range of organisms, RNAi plays an essential role in organismal development and various cellular processes. As viral RNA can activate the RNAi pathway in host cells leading to degradation of the viral RNA, RNAi is also used by plants and some animals to fight viral infections. Furthermore, RNAi can also silence mobile elements in the genome, such as transposons, to maintain genome stability. Currently, large amounts of data have established the pervasiveness of small RNA mediated RNAi in many organisms. For example, in the human genome, over 60% of genes are regulated by small noncoding RNAs [48]. Since its discovery, RNAi has been applied as a powerful research tool to silence virtually any gene in the genome in order to decipher their functions. Clinically, small RNAs have been tested as a strategy for gene therapy through turning off faulty genes that underlie many genetic diseases.

RNAi is mediated by three principal groups of small noncoding RNAs: miRNA, siRNA, and piRNA. All these small RNAs induce RNAi through the same basic pathway that involves a ribonucleoprotein complex called the RNA-induced silencing complex (RISC). Following is a more detailed introduction to these three groups of small RNAs and their differences.

3.4.4.1 miRNA

Mature miRNA, in the size range of 19 to 24 bases, induces gene silencing through mRNA translational repression or decay. The precursor of miRNA is usually transcribed from non-protein-coding genes in the genome (Figure 3.4). The primary transcript, called pri-miRNA, contains an internal hairpin structure and is much longer than mature miRNA. For initial processing, the pri-miRNA is first trimmed in the nucleus by a ribonuclease called Drosha that exists as part of a protein complex called the microprocessor, to an intermediate molecule called pre-miRNA, about 70 nucleotides in size. Alternatively, some miRNA precursors originate from introns spliced out from protein-coding transcripts. These precursors, to be processed for the generation of mirtrons (miRNAs derived from introns), bypass the microprocessor complex in the nucleus. For further processing, the pre-miRNA and the mirtron precursor are exported out of the nucleus into the cytoplasm, where they are cleaved by the endoribonuclease Dicer to form double-stranded miRNA. The double-stranded miRNA is subsequently loaded into RISC. Argonaute, the core protein component of RISC, unwinds the two miRNA strands and discard one of them [49]. The remaining strand is used by Argonaute as the guide sequence to identify related mRNA targets through imperfect base pairing with a seed sequence usually located in the 3′-UTR of mRNAs. Through this miRNA–mRNA interaction, RISC induces silencing of target genes through repressing translation of the mRNAs and/or

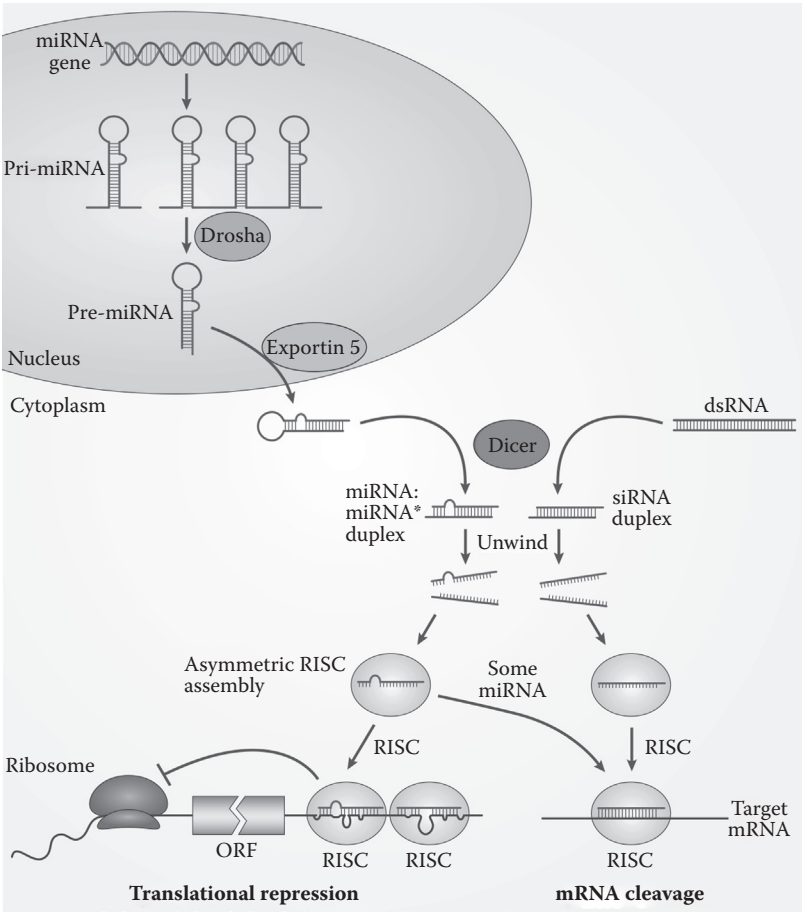

FIGURE 3.4
The generation and functioning of miRNA and siRNA in suppressing target mRNA activity. Genomic regions that code for miRNAs are first transcribed into pri-miRNAs, which are processed into smaller pre-miRNAs in the nucleus by Drosha. The pre-miRNAs are then transported by exportin 5 into the cytoplasm, where they are further reduced to miRNA:miRNA* duplex by Dicer. While both strands of the duplex can be functional, only one strand is assembled into the RNA-induced silencing complex (RISC), which induces translational repression or cleavage of target mRNAs. Long double-stranded RNA can also be processed by Dicer to generate siRNA duplex, which also uses RISC to break down target mRNA molecules. (From L He and GJ Hannon, MicroRNAs: Small RNAs with a big role in gene regulation, *Nature Reviews Genetics* 2004, 5:522–531. With permission.)

their deadenylation and degradation. Because the base pairing is imperfect, one miRNA can target multiple target genes' mRNAs. Conversely, one mRNA may be targeted by multiple miRNAs.

3.4.4.2 siRNA

While being similar in size and using basically the same system for gene silencing, siRNA differs from miRNA in a number of aspects. On origin, siRNA is usually exogenously introduced, such as from viral invasion or artificial injection. But they can also be generated endogenously, for example, from repeat-sequence-generated transcripts (such as those from telomeres or transposons), or RNAs synthesized from convergent transcription (in which both strands of a DNA sequence are transcribed from the two opposite orientations with corresponding promoters), or other naturally occurring sense–antisense transcript pairs [50]. To generate mature siRNA, exogenously introduced double-stranded RNA, or endogenously transcribed precursor that is transported from the nucleus to the cytoplasm, is cleaved by Dicer. The mature siRNA is then loaded into RISC for silencing target mRNAs by Argonaute. On target mRNA identification, siRNA differs from miRNA in that it has perfect or nearly perfect sequence complementarity with their target. On the mechanism of gene silencing, siRNA usually leads to endonucleolytic cleavage, also called slicing, of the mRNAs.

3.4.4.3 piRNA

piRNA is a relatively newer class of small noncoding RNAs between 24 and 31 nucleotides in length and have functions in animal germline tissues. While using a similar basic RNAi mechanism, piRNA is different from miRNA and siRNA in two major aspects. One is that its biogenesis does not involve Dicer, and the other is that, for gene silencing, it specifically interacts with Piwi, a different clade of Argonaute proteins. The biogenesis of piRNA starts from transcription of long RNAs from specific loci of the genome called piRNA clusters. With regard to these clusters, it has been found that while their locations in the genome do not show much change in related species, their sequences are not conserved even in closely related species. After transcription, the RNAs are transported out of the nucleus, and then subjected to a parsing process by endonuclease(s) that is currently not clearly known. To induce gene silencing, mature piRNA is loaded into RISC that contains Piwi, which uses the piRNA sequence as a guide to silence target mRNAs by slicing. In addition, piRNA-loaded mature RISC can also be transported into the nucleus, where it finds and silences target mRNAs that are still in the process of being transcribed. This transcriptional gene silencing is achieved through interactions with other protein factors in the nucleus, and histone modification that alters chromatin structure and gene access. The currently best-known function of piRNAs, through post-transcriptional and

transcriptional gene silencing, is to repress transposon activity and thereby maintain genome stability in germline cells. Nontransposon gene targets of Piwi-interacting RNAs have also been reported such as those related to early development.

3.4.5 Long Noncoding RNAs

Some noncoding RNAs, unlike the small RNAs, are rather long with an average length of over 200 nucleotides in their mature form. These RNAs, called long noncoding RNAs, have been discovered more recently and are therefore less studied. The biogenesis of lncRNAs is somewhat similar to that of mRNAs, as many of them are transcribed by RNA polymerase II and subject to splicing, capping at the 5′ end, and polyadenylation at the 3′ end. Unlike mRNAs, however, they are usually shorter with a median length of ~600 nucleotides, have fewer exons, and are generally expressed at levels lower than those of mRNAs. Furthermore, their expression displays higher tissue and developmental stage specificity than mRNAs, and are mostly localized in the nucleus rather than transported to the cytoplasm.

Although they are relatively new, evidence on their importance in regulating fundamental cellular functions is rapidly accumulating [51,52]. They have been known to control many steps of gene activity, including chromatin remodeling, transcriptional regulation, mRNA processing, stability, localization, and translation [53,54]. For example, some lncRNAs, such as Xist and HOTAIR, repress gene transcription at target genomic sites by interacting with chromatin remodeling protein complexes [55,56]. A class of lncRNAs that was recently discovered by NGS is transcribed from enhancer regions of protein-coding genes. These transcripts, called eRNAs (or enhancer RNAs), have been shown to affect transcription of protein-coding genes that are regulated by the enhancers [57]. In general, lncRNAs regulate gene activity via binding to transcription factors, repressing promoter activity, and interacting with mRNA-binding proteins and splicing factors. In addition, lncRNAs can directly interact with mRNAs and thereby influence their stability and translation [58,59]. Because of their functional importance, it is not surprising that abnormal lncRNA expression can lead to diseases such as cancer [60].

3.4.6 Other Noncoding RNAs

Deep sequencing of the cellular transcriptome has led to the discovery of other noncoding RNAs. For example, circular RNAs (circRNAs) exist in many species and cell types. Unlike linear RNAs, which include all the RNA species introduced so far, circRNAs have their 5′ and 3′ ends joined forming a loop structure. This structure makes them less vulnerable to attacks from RNases and expectedly more stable. Because their widespread existence was not unveiled until 2012 with the use of RNA-Seq, the function of most circRNAs is still largely unknown. Some reports suggest that

they have regulatory potency, including acting as miRNA sponges [61,62]. Besides the major noncoding RNAs introduced in this chapter, there are also other classes of noncoding RNA species in cells that perform a remarkable array of functions [63]. It is highly possible that new classes of noncoding RNAs will continue to be discovered through RNA sequencing.

3.5 The Cellular Transcriptional Landscape

Traditionally, protein-coding mRNA transcripts used to be the major targets of transcriptional studies and as a result were often regarded as the major component of a transcriptome. However, with the evolution of transcriptomics technologies and as a result of the discovery of the wide variety of noncoding RNAs, it has been gradually realized that protein-coding transcripts only constitute a minor fraction of a cell's transcribed sequences. Large-scale studies on the landscape of cellular transcription, as carried out by consortia including the FANTOM (Functional Annotation Of the Mammalian genome) and ENCODE (Encyclopedia of DNA Elements), have revealed that the majority of the genome is transcribed, and a large proportion of the transcriptome is noncoding RNAs [64,65]. For example, after studying the transcriptional landscape of 15 human cell lines, encompassing RNA populations isolated from different cellular sub-compartments including the cytosol and the nucleus, the ENCODE consortium found that the transcription of the genome is pervasive and 75% of genomic sequences, including those located in gene-poor regions, are present in transcripts. Many of the transcripts come from intronic and intergenic regions that are not currently characterized and therefore novel. The complex cellular RNA landscape adds further evidence that RNA is not simply a messenger between DNA and protein.

Section II

Introduction to Next-Generation Sequencing (NGS) and NGS Data Analysis

4

Next-Generation Sequencing (NGS) Technologies: Ins and Outs

4.1 How to Sequence DNA: From First Generation to the Next

The sequence of nucleotides in a DNA molecule can be determined in multiple ways. Early in the 1970s, biochemists (Drs. Walter Gilbert and Frederick Sanger) devised different methods to sequence DNA. Gilbert's method is based on chemical procedures that break down DNA specifically at each of the four bases. Sanger's method, on the other hand, takes advantage of the DNA synthesis process. In this process, a new DNA chain is synthesized base by base using sequence information on the template (Chapter 2). The use of chemically modified nucleotides, that is dideoxynucleotides, as irreversible DNA chain terminators in Sanger's method randomly stops the synthesis process at each base position, so a series of new DNA chains of various lengths that differ by one base are produced (Figure 4.1). Determining the lengths in single base resolution of specifically broken DNA molecules in Gilbert's method, or new DNA chains that are randomly terminated at each of the four nucleotides in Sanger's method, enabled sequencing of the template DNA. Over the years the Sanger method was further developed. The integration of automation into the process reduced human involvement and improved efficiency. The use of fluorescently labeled terminators, instead of the radioactively labeled terminators that were initially used, made it safer to operate and sequence detection more robust. The improved separation of DNA chains with the use of capillary electrophoresis, instead of slab gels, enabled high-confidence base calls. All these developments led to the Sanger method being widely adopted and the method of choice for the Human Genome Project. Even today this method is still widely used for single or low-throughput DNA sequencing. With the coming of next-generation sequencing (NGS), this method has become the synonym of first-generation sequencing.

Although it is robust in sequencing individual DNA fragments, the Sanger method cannot easily achieve high throughput, which is the key to

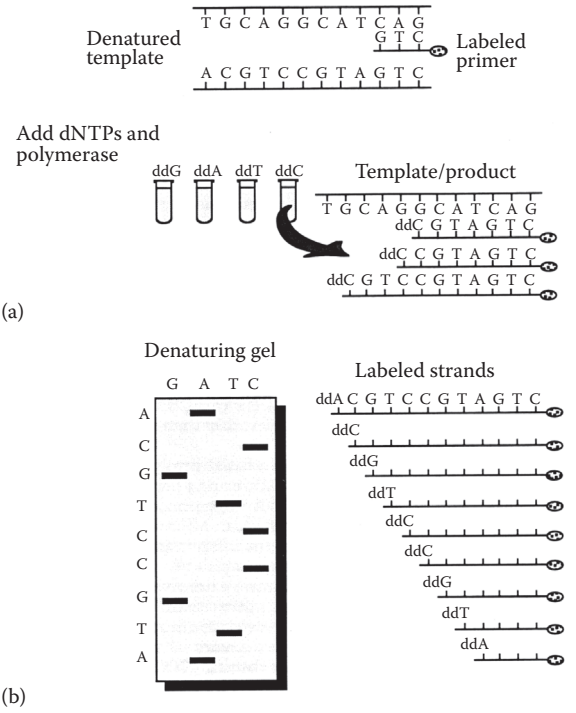

FIGURE 4.1
The Sanger sequencing method as originally proposed. This method involves a step for new DNA strand synthesis using the sequencing target DNA as the template (panel a), followed by sequence deduction through resolution of the newly synthesized DNA strands (panel b). In the first step, the new strand synthesis reaction contains denatured DNA template, radioactively labeled primer, DNA polymerase, and dNTPs. For the dNTPs, the Sanger method is characterized by the use of dideoxynucleotides (ddG, ddA, ddT, and ddC, as shown in the figure) along with regular unmodified nucleotides. The DNA polymerase in the reaction incorporates dideoxynucleotides into the elongating DNA strand like regular nucleotides, but once a dideoxynucleotide is incorporated, the strand elongation terminates. In this sequencing scheme, each of the four dideoxynucleotides is run in a separate reaction, and the ratio of these dideoxynucleotides to their regular counterparts in each reaction is controlled so that the polymerization can randomly terminate at each base position. The end product of each reaction is a population of DNA fragments with different lengths, with the length of each fragment dependent on where the dideoxynucleotide is incorporated. Panel b illustrates the separation of these fragments in a denaturing gel by electrophoresis, in which smaller fragments migrate faster than larger ones and appear toward the bottom of the gel. The radioactive labeling on the primer enables visualization of the fragments as bands on the gel. Shown on the right are DNA fragments that correspond to each of the bands, respectively. From the arrangement of DNA bands the complementary sequence of the original DNA template can be deduced (shown on the left of the sequencing gel, read from bottom upward). (From P Moran, Overview of commonly used DNA techniques, in LK Park, P Moran, and RS Waples, eds., *Application of DNA Technology to the Management of Pacific Salmon*, 1994, 15–26, Department of Commerce, NOAA Technical Memorandum NMFS-NWFSC-17. © Paul Moran, NOAA's Northwest Fisheries Science Center. With permission.)

lowering sequencing cost, largely due to the segregation of its DNA synthesis process and the subsequent DNA chain separation/detection process. Its principle of sequencing-by-synthesis, however, becomes the basis of several NGS technologies, all of which are characterized by extremely high throughput. These technologies generally use nucleotides with reversible terminator or other cleavable chemical modifications, or regular unmodified nucleotides, so the new DNA strand synthesis is not permanently terminated and therefore can be monitored as or after each base is incorporated. This development, along with advancements in other relevant fields, makes it possible to conduct sequencing of millions of DNA fragments simultaneously.

One of the early NGS technologies, 454 (later acquired by Roche), achieved high-throughput sequencing by further developing a method called pyrosequencing. This method is based on the detection of the pyrophosphate released after each nucleotide incorporation in the new DNA strand synthesis [66]. In this technology, each of the four different nucleotides is added into the sequencing reaction in a fixed order one at a time. If complementary, the corresponding nucleotide (or more than one, if there is a homopolymer on the template) is ligated to the new strand, and as part of the ligation reaction a pyrophosphate is released as a side product. An enzyme called ATP sulfurylase converts this pyrophosphate to ATP, which in turn is used to convert luciferin to oxyluciferin by luciferase. The generated oxyluciferin emits light, and the amount of light emitted is generally proportional to the number of nucleotides incorporated. By detecting light emission after each cycle of nucleotide addition, the sequence on each DNA template is deduced. High throughput is achieved when massive numbers of DNA templates are sequenced in this fashion simultaneously. Using the 454/Roche technology, sequence reads of 400 to 500 bp in length are generated. A number of widely used NGS technologies, including Illumina reversible dye-terminator sequencing, Ion Torrent semiconductor sequencing, and Pacific Biosciences single-molecule real-time sequencing, are all based on the sequencing-by-synthesis principle. The specifics of these methods will be detailed in Section 4.3.

Not all NGS technologies are based on the principle of sequencing-by-synthesis. For example, the SOLiD (Sequencing by Oligonucleotide Ligation and Detection) system from Life Technologies uses a sequencing-by-ligation process. Nanopore sequencing, as commercialized by companies such as Oxford Nanopore Technologies, deduces the DNA sequence through detecting differential electric field disturbances caused by different nucleotides when a strand of DNA is threaded through a nanopore structure. Despite the differences in how different NGS technologies work in principle, the overall workflow of an NGS experiment is more or less similar. Next is an overview of a typical NGS experimental workflow as conducted in a wet lab, along with early-phase data analysis.

4.2 A Typical NGS Experimental Workflow

Sequencing genomic DNA, or RNA transcripts, with NGS technologies involves multiple steps (Figure 4.2). The early steps in this process are to construct sequencing libraries from DNA or RNA molecules extracted from biological samples of interest. As they are usually too large to be directly handled by most NGS technologies, especially those that produce short reads, the extracted DNA or RNA molecules need to be broken into smaller fragments first. This fragmentation can be achieved with different techniques, including sonication, nebulization, acoustic shearing, or enzymatic treatment. The fragmentation step is usually followed by a size selection step to collect fragments in a certain target range.

A key step in the sequencing library construction process is the ligation of adapters to the two ends of DNA fragments. For RNA fragments, they are usually converted to complementary DNA (cDNA) first before adding the adapters. The adapters are artificial sequences that contain multiple components that serve several purposes in the sequencing process. These sequence components include (1) universal sequencing primer sequence(s) that initiate sequencing reactions on each fragment; (2) polymerase chain reaction (PCR) amplification primer sequences for sequencing template enrichment; (3) anchoring sequences that enable presequencing attachment of the DNA fragments to a solid support, such as glass slide or bead, where sequencing reactions take place; and (4) indexing (or "barcode") sequences to differentiate multiple samples when they are sequenced together. While they generally serve similar functions in different NGS platforms, the actual standard adapter sequences are specific to each platform. It is also possible to design custom adapters to meet special needs as long as key adapter sequence elements essential for a platform are in place. Prior to adapter ligation, the two ends of the DNA (or cDNA) fragments need to be prepared in an end repair step (not shown in Figure 4.2). After adapter ligation, the sequencing DNA templates in the resultant library usually need to be enriched through a PCR amplification step using the primer sequence built in the adapters. Alternatively, the constructed library may be sequenced PCR-free without enrichment on some platforms.

To sequence the constructed DNA libraries, different platforms use different approaches and collect sequencing signals that are of different nature. For example, as to be detailed next, optical or physicochemical signals are often captured and processed to generate sequence readout of the DNA fragments. The optical signal is usually based on either direct light emission (as in the 454/Roche platform) or fluorescence generated from the use of chemically modified nucleotides that carry fluorescent labels (as in the Illumina and Pacific Biosciences systems). The physicochemical signal is measured from physical or chemical activity associated with the sequencing process, such as the release of H^+ and the concomitant pH change (as

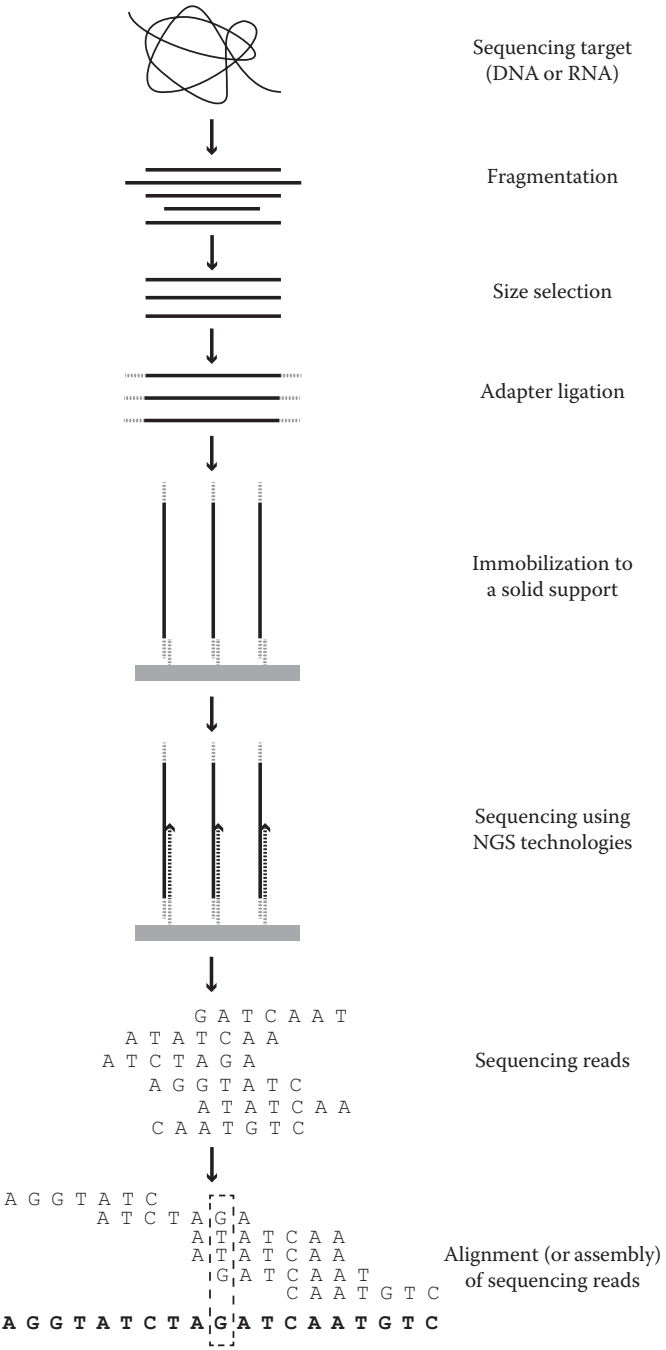

FIGURE 4.2
The general workflow of an NGS experiment.

in the Ion Torrent system), or electric field disturbance (as in Nanopore sequencing). Processing of such optical or physicochemical signals leads to sequence deduction of the DNA fragments. For data output and storing, the raw, unprocessed signals are usually stored in a platform-specific format, whereas the processed sequence reads are usually reported in a file format that is more universal (for details, see Chapter 5).

Although a DNA sequence can be read from only one end of a DNA template (i.e., single-read sequencing), it can also be read from both ends of the DNA fragment (called paired-end sequencing). Besides doubling the total number of sequence reads, paired-end sequencing also has the advantage of facilitating subsequent alignment to a reference genome (Chapter 5) or genome assembly (Chapter 10). Because DNA fragments are usually size selected and therefore their approximate length known, the resultant paired reads and the distance between them provide additional information on how to align the reads to a reference genome or assemble them into a new genome. Most current NGS platforms can accommodate paired-end sequencing.

From the aforementioned general NGS experimental workflow, it is clear that besides the ingenuity in the development of new sequencing chemistries or schemes, the success of NGS technologies in achieving extremely high throughput with the simultaneous detection of millions of DNA molecules is also due to modern engineering and computing feats. Advancements in microfluidics and microfabrication make signal detection from microvolume of sequencing reaction possible. Developments in modern optics, microscopy, and imaging technology enable tracking of sequencing reactions in high resolution, high fidelity, and high speed. Some NGS platforms also rely on the decades of progress in the semiconductor industry or more recent but rapid development in nanopore technology (such as the Ion Torrent and Nanopore platforms, respectively). High-performance computing makes it possible to process and deconvolve the torrent of signals recorded from millions of these processes.

4.3 Ins and Outs of Different NGS Platforms

The NGS technologies mentioned earlier have generated the vast majority of NGS data in existence today. Several of these platforms will continue to produce more NGS data for life science research, but some of these systems have been discontinued. For example, the Roche/454 pyrosequencing platform was discontinued in 2013, and the NGS company Helicos Biosciences filed for bankruptcy in 2012. Although sequencing data is still being generated from existing 454 and Helicos systems, this section focuses on the platforms that are currently most active and widely used. As NGS technologies continue to evolve, new platforms will appear while some current technologies become

obsolete. Although an overview of NGS platforms usually becomes outdated fairly soon, the guiding principles on the analysis of NGS data introduced in this book will remain.

4.3.1 Illumina Reversible Dye-Terminator Sequencing

4.3.1.1 Sequencing Principle

The Illumina NGS platform is by far the most popular and has generated the largest amounts of NGS data. At the core of the Illumina sequencing technology is the employment of fluorescently labeled nucleotides with reversible terminator [67]. As previously mentioned, this method is based on the same basic principle of sequencing-by-synthesis as the Sanger method; but unlike the Sanger method, after the incorporation of each of these specially modified nucleotides, the terminator moiety they carry only temporarily prevents the new DNA strand from extending. After optical detection of the incorporated nucleotide based on its specific fluorescent label, the terminator moiety is cleaved, and thereby the new strand synthesis resumes for the next cycle of nucleotide incorporation. For simultaneous detection of nucleotide incorporation in millions of sequencing reactions, dATP, dCTP, dGTP, and dTTP are labeled with different fluorescent labels so each nucleotide can be detected by the different fluorescence signal they emit. The fluorescent labels and the reversible terminator moiety are attached to the nucleotides via the same chemical bond, so both of them can be cleaved off in a single reaction after each nucleotide incorporation and detection cycle to prepare for incorporation of the next nucleotide.

4.3.1.2 Implementation

The sequencing reaction in an Illumina NGS system takes place in a flow cell (Figure 4.3). The microfluidic channels in the flow cell, often called lanes, are where sequencing reactions take place and sequencing signals are collected through scanning. The top and bottom surface of each lane is covered with a lawn of oligonucleotide sequences that are complementary to the anchor sequences in the ligated adapters. When sequencing libraries are loaded into each of the lanes, DNA templates in the libraries bind to these oligonucleotide sequences and become immobilized onto the lane surface (Figure 4.4). After immobilization, each template molecule is clonally amplified through a process called "bridge amplification," through which up to 1000 identical copies of the template are generated in close proximity (<1 micron in diameter) forming a cluster. During sequencing, these clusters are basic detection units, which generate enough signal intensity for base calling.

Under ideal conditions the simultaneous incorporation of nucleotides to the many identical copies of sequencing templates in a single cluster is expected to be in synchronization from step to step and therefore remain in

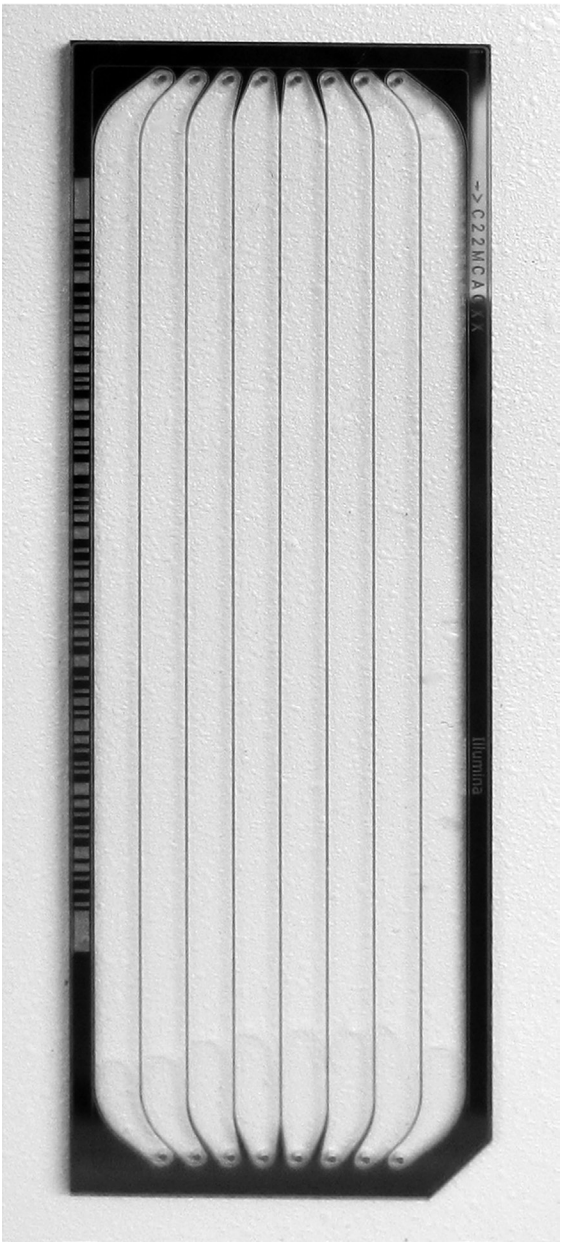

FIGURE 4.3
An Illumina sequencing flow cell. It is a special glass slide that contains fluidic channels inside (called lanes). Sequencing libraries are loaded into the lanes for massively parallel sequencing after template immobilization and cluster generation. In each step of the sequencing process, a DNA synthesis mixture, including DNA polymerases and modified dNTPs, is pumped into and out of each of the lanes through their inlet and outlet ports located at the two ends.

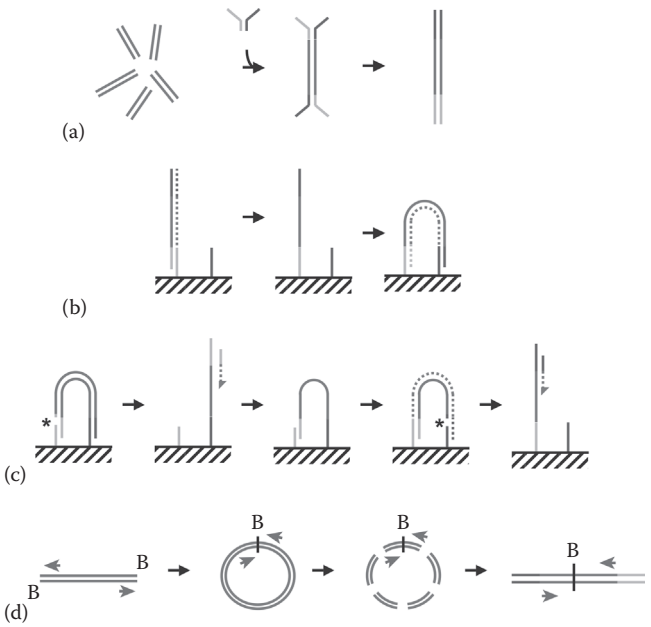

FIGURE 4.4
Illumina sequencing sample preparation and sequencing approaches. (a) Ligation of adapters in a forked configuration to fragmented and size-selected DNA. Each ligation product is then amplified using primers complementary to sequences in the adapters to generate blunt-end amplicons. (b) Clonal amplification by isothermal bridge amplification. To achieve this, DNA strands are first separated by denaturing, and each strand is attached to the flow cell surface with complementary sequences. After a new strand (dotted line) is synthesized on the flow cell surface, the original DNA strand is removed. The "free" end of the new strand then attaches to the other anchoring oligonucleotide sequence on the flow cell surface with sequence complementarity, which leads to the formation of a bridge configuration for synthesis of a new complementary strand. Repetition of this process generates many copies of the original sequence template in a cluster. (c) Sequencing from one end or both ends of the DNA templates. Prior to sequencing, one strand is cleaved within one adapter sequence (marked with an asterisk) and then removed after denaturing. The remaining strand is used as the template for sequencing from one end. For sequencing from the other end (i.e., paired sequencing), the sequencing template is rebuilt. The template rebuilding process includes removal of the first read's new strand after denaturing, generation of a complementary template (dotted line) with the bridge synthesis, and cleavage and removal of the original template. The new template is used to generate read 2 from the opposite end. (d) Mate-pair sequencing. This strategy enables sequencing of the two ends of long DNA fragments (e.g., >1 kb). In this strategy, the long DNA fragments are first circularized and then fragmented. Those fragments that contain the end junction are then sequenced using the paired-end process illustrated in b. (From DR Bentley, S Balasubramanian, HP Swerdlow, GP Smith, J Milton, CG Brown, KP Hall et al., Accurate whole human genome sequencing using reversible terminator chemistry, *Nature* 2008, 456:53–59. With permission.)

phase. In reality, a small percentage of templates lose sync with the majority of molecules in the same cluster, leading to either falling behind (called phasing), due to incomplete removal of the terminator as well as missing a cycle, or being one or several bases ahead (prephasing), due to incorporation of nucleotides with no terminators. The existence of phasing and prephasing in a cluster leads to increased background noise and decreased base call quality. When more and more sequencing cycles are conducted, this problem becomes worse. This is why platforms that are based on clonal amplification (which also include the Ion Torrent platform to be detailed next) have declining base call quality scores toward the end. Eventually the decrease in base call quality reaches a threshold beyond which the quality scores become simply unacceptable. The gradual loss of synchronicity is a major determinant of read length for these platforms.

4.3.1.3 Error Rate, Read Length, Data Output, and Run Time

The overall error rate of the Illumina sequencing method is below 1%, which makes it one of the most accurate NGS platforms currently available. The most common type of errors is single nucleotide substitution. On read length, the high-throughput HiSeq system can generate reads of up to 125 bases in length using the high-output run mode and 250 bases using the rapid run mode. The relatively low-throughput MiSeq system produces reads up to 300 bases long. With regard to data output, the HiSeq system can generate up to 8 billion paired-end reads on two 8-lane flow cells, with a total data yield of 1 Tb (terabase) using the high-output run mode. Using the rapid run mode, it can produce up to 1.2 billion pair-end reads on two 2-lane flow cells, with a total yield of 300 Gb (gigabase). The MiSeq system, on the low side, can generate up to 50 million paired-end reads, with a total data output of 15 Gb. With single-end sequencing, the total number of reads and data yield is half of the aforementioned numbers. Among the three major platforms presented here, the Illumina systems offer the shortest reads. However, at the read length of up to 250 bases, it can meet the needs of most NGS applications. It should be noted that the technical numbers listed here are as of early 2015 and will change with future system updates.

Concerning run time, a MiSeq run takes 5 to 55 hours, while a HiSeq 2500 run takes from 7 hours to up to 11 days depending on run modes (rapid vs. high output) and chemistry. The enzymatic step for the incorporation of nucleotides actually takes little time. The majority of time is spent imaging the clusters on the flow cell surface. For the imaging, the fluorescent labels on the nucleotides are illuminated with a red and a green laser and scanned through four different filters. After each cycle, four images are generated on each tile and the current high output flow cell contains 768 tiles. The imaging step can be sped up by decreasing the total scanning surface area (the 2-lane rapid run flow cell has only 128 tiles), but this comes with the tradeoff of decreased data volume and cost-efficiency.

There are three steps in the Illumina sequence data generation process. First, raw images captured after each cycle are analyzed to locate clusters and report signal intensity, coordinates, and noise level for each cluster. This step is conducted by the instrument control software. The output from this step is fed into the next step of base calling by the instrument Real Time Analysis (RTA) software, which uses cluster signal intensity and noise level to make base calls and quality score calculation. This step also filters out low-quality reads. In the third step, the base call files, or bcl files, are converted to compressed FASTQ files by the Illumina's proprietary software CASAVA. If samples are indexed and sequenced in a multiplex fashion, demultiplexing of the sequence data is also performed in the third step. The compressed and demultiplexed FASTQ files are what an end user receives from an NGS core facility after the completion of a run.

4.3.2 Ion Torrent Semiconductor Sequencing

4.3.2.1 Sequencing Principle

The Ion Torrent semiconductor sequencing system is the first NGS platform that does not rely on chemically modified nucleotides, fluorescence labeling, and the time-consuming step of image scanning, thereby achieving much faster speed, lower cost, and smaller equipment footprint. The Ion Torrent platform sequences DNA through detecting the H^+ ion released after each nucleotide incorporation in the sequencing-by-synthesis process. When a nucleotide is incorporated into a new DNA strand, the chemical reaction catalyzed by DNA polymerase releases a pyrophosphate group and a H^+ (proton). The release of H^+ leads to pH change in the vicinity of the reaction, which can be detected and used to determine the nucleotide incorporated in the last cycle. As the change in pH value is not nucleotide-specific, to determine DNA sequence, each of the substrate nucleotides (dATP, dGTP, dCTP, and dTTP) is added to the reaction in order at different times. A detected pH change after the introduction of a nucleotide suggests that the template strand contains its complementary base at the last position.

4.3.2.2 Implementation

The library construction process in this technology is similar to other NGS technologies, involving ligation of platform-specific primers to DNA shotgun fragments. The library fragments are then clonally amplified by emulsion PCR onto the surface of 3-micron diameter beads. The microbeads coated with the amplified sequence templates are then deposited into an Ion chip. Each Ion chip has a liquid flow chamber that allows influx and efflux of native nucleotides (introduced one at a time), along with DNA polymerase and buffer that are needed in the sequencing-by-synthesis process. For

measuring possible pH change associated with each introduction of nucleotide, there are millions of pH microsensors that are manufactured on the chip bottom by the employment of standard processes used in the semiconductor industry.

4.3.2.3 Error Rate, Read Length, Date Output, and Run Time

The overall error rate of the Ion Torrent platform is higher than the Illumina platform but lower than the Pacific Biosciences system (see Section 4.3.3.3). The major type of errors is indels caused by homopolymers. When the DNA template contains a homopolymeric region, which is a stretch of identical nucleotides (such as TTTTT), the signal in pH change is stronger and proportional to the number of nucleotides contained in the homopolymer. For example, if the template contains two T's, the influx of dATPs will generate a pH change signal that is about twice as strong as that generated for a single T. Accordingly the signal for 3 T's will be 1.5-fold that of 2 T's, and the signal for 6 T's will be reduced to 1.2-fold that of 5 T's. Therefore, with the increase in the total number of the repeat base, there is a gradual decrease in signal strength ratio, which reduces the reliability of calling the total number of the base correctly. It is estimated that the current error rate for calling a 5-base homopolymer is 3.5%.

The current (as of early 2015) Ion PGM system using an Ion 318 chip (v2) takes 4 to 7 hours to generate 4 million to 5.5 million reads that are 35 to 400 bases in length (~600 Mb to 2 Gb data). The higher throughput Ion Proton system using the Ion PI chip generates in 2 to 4 hours 10 Gb data with 60 million to 80 million reads that are up to 200 bases long, enough to sequence two human exomes. Use of the PII chip on the Proton system can generate 32 Gb data in 4 hours, enabling sequencing of the human genome 10 times. The data volume generated from this technology is expected to increase with continuous development of new chips by increasing total chip surface area and microwell/pH-microsensor density.

4.3.3 Pacific Biosciences Single Molecule Real-Time (SMRT) Sequencing

4.3.3.1 Sequencing Principle

The Pacific Biosciences' single molecule real-time (SMRT) sequencing platform is usually regarded as third-generation sequencing technology, as it is sensitive enough to sequence single DNA molecules and therefore bypasses any form of amplification [68]. In addition, this platform generates much longer reads than most other NGS platforms, with the current median length in the range of 8 to 10 kb. While it is also based on the principle of sequencing-by-synthesis, different from the Illumina method, SMRT sequencing uses nucleotides that carry fluorescent labels linked to

their end phosphate group but no terminator group. When a nucleotide is incorporated into an elongating DNA strand, with the cleavage of the end phosphate group (actually a pyrophosphate group as mentioned earlier), the fluorescent label is simultaneously released, which enables real-time signal detection. As this process does not involve a separate step of fluorescent label releasing and detection, the sequence-detecting signal is continuously recorded as a 75-frames-per-second movie instead of using scanner images.

4.3.3.2 Implementation

The single-molecule sensitivity of this technology is achieved by the use of zero-mode waveguide (or ZMW), a hole tens of nanometers in diameter microfabricated in a metal film of 100 nm thickness, which is in turn deposited onto a glass substrate. To conduct sequencing in a ZMW, a single DNA polymerase and a single DNA strand (sequencing target) are immobilized to its bottom. Because the diameter of a ZMW is smaller than the wavelength of visible light, and due to the natural behavior of visible light passing through such a small opening from the glass bottom, only the bottom 30 nm of the ZMW is illuminated. Having a detection volume of only 20 zeptoliters (10^{-21} L), this detection scheme greatly reduces background noise and enables detection of nucleotide incorporation into a single DNA molecule.

While the SMRT platform performs single molecule sequencing, it still requires a lot of DNA samples at the starting point (1 μg currently). The library prep process for SMRT is similar to other NGS platforms, including shotgun fragmentation of DNA into required size, which is multikilobases for this technology. This is followed by DNA fragment end repair and adapter ligation. The resultant sequencing templates are then annealed to sequencing primers, onto which DNA polymerases are subsequently bound. Prior to sequencing, the template–primer–polymerase complex is immobilized to the 150,000 ZMWs at the bottom of a SMRT cell. Due to technical restraints, currently only about one third (~50,000 to 60,000) of these ZMWs generate quality reads that pass filter.

4.3.3.3 Error Rate, Read Length, Data Output, and Run Time

One major disadvantage of this platform is its high error rate and run cost compared to the other platforms (Table 4.1). The error rates, at 10% to 15%, are higher than the other two platforms, with the most common error types being indels. On actual read length and data output, with the current movie length of 3 hours, the longest read that can be sequenced exceeds 30 kb (8.5 kb on average). The current total data output of a SMRT cell is 375 Mb on the current model (RSII).

TABLE 4.1

Comparison of Current NGS Platforms

Platform	Principle	Detection	Read Length	Data Output per Run	Cost per Gb[c]	Common Error Type	Error Rate[a]	Paired-End Sequencing	Required DNA Sample Amount
HiSeq 2500	Reversible Terminator	Fluorescence	125–250 bases[b]	1000 Gb	$	Single nucleotide substitution	10^{-3}	Yes	50–1000 ng
MiSeq	Reversible Terminator	Fluorescence	300 bases	15 Gb	$$	Single nucleotide substitution	10^{-3}	Yes	50–1000 ng
Ion Torrent PGM	Proton release and pH change	pH change	400 bases	Up to 2 Gb	$$$	Indels (mostly at homopolymers)	10^{-2}	Yes	100–1000 ng
Ion Proton	Proton release and pH change	pH change	200 bases	10 Gb (PI Chip)	$$$$	Indels (mostly at homopolymers)	10^{-2}	Yes	100–1000 ng
PacBio RSII	ZMW and single molecule sequencing	Fluorescence	8.5 kb average	375 Mb	$$$$$	Indels	10^{-1}	No	1000 ng

[a] *Source*: CW Fuller, LR Middendorf, SA Benner, GM Church, T Harris, X Huang, SB Jovanovich et al., The challenges of sequencing by synthesis, *Nature Biotechnology* 2009, 27:1013–1023. With some modifications.

[b] 125 bases: high-output mode; 250 bases: rapid mode.

[c] Relative sequencing cost: $, least expensive; $$$$$, most expensive.

4.4 Biases and Other Adverse Factors That May Affect NGS Data Accuracy

Just as a certain level of erroneous base calls is inherent to an NGS platform, the multiple steps that lead to the generation of sequence calls are not immune to biases. Different from errors, biases affect accurate representation of the original DNA or RNA population leading to higher (or lower) representation of some sequences than expected. The major source of biases in NGS are the molecular steps involved in the library construction and the sequencing process itself. Besides biases, there are also other potential factors that may lead to the generation of inaccurate sequencing signals. Detailed next are the various potential biases and other adverse factors during sequencing library construction and sequencing that may affect NGS data accuracy. It should be noted that while it is impossible to avoid them altogether, being aware of their existence is the first step toward minimizing their influence through careful experimental design and data analysis, and developing more robust analytic algorithms.

4.4.1 Biases in Library Construction

Biases in DNA fragmentation and fragment size selection. The initial step of library construction, that is, DNA fragmentation, is usually assumed to be a random process and not dependent on sequence context. This has been shown not to be the case [69]. For example, sonication and nebulization cause DNA strand breaks after a C residue more often than expected. After DNA fragmentation, the size selection process may also introduce bias. For example, if gel extraction is employed for this process, the use of a high gel melting temperature favors recovery of fragments with high GC content.

Ligation biases. After fragmentation and size selection, double-stranded DNA fragments are usually adenylated, after end repair, at the two 3′-ends generating 3′-dA tails that facilitate subsequent ligation of adapters that carry 5′-dT overhangs and thereby avoid self-ligation of DNA fragments or adapters. This AT-overhang-based adapter ligation process, however, tends to be biased against DNA fragments that start with a T [70]. The sequencing of large RNA species, such as mRNAs or long noncoding RNAs, is also affected by this bias, as cDNA molecules reverse transcribed from these species are also subjected to the same adapter ligation process. Small RNA sequencing is not affected by this bias, as the ligation of adapters in small RNA sequencing library preparation is carried out prior to the reverse transcription step. The small RNA adapter ligation step, however, introduces a different type of bias, which affects some small RNAs in a sequence-specific manner. Sequence specificity underlies small RNA secondary and tertiary structure, which is also affected by temperature, concentration of cations, and destabilizing organic agents (such as dimethyl sulfoxide [DMSO]) in the

ligation reaction mixture. The efficiency of small RNA adapter ligation is influenced by their secondary and tertiary structure [71].

PCR biases. After adapter ligation, the DNA library is usually enriched by PCR for sequencing on most of the current NGS platforms. PCR, based on the use of DNA polymerases, is known to be biased against DNA fragments that are extremely GC- or AT-rich [72]. This can lead to variation in the coverage of different genomic regions and underrepresentation of those regions that are GC- or AT-rich. Although optimization of PCR conditions can ameliorate this bias to some degree, especially for high-GC regions, this bias can only be eliminated via adoption of a PCR-free workflow. To achieve this, Illumina has introduced a PCR-free workflow. Similar workflow has also been established for the Ion Torrent platform.

4.4.2 Biases and Other Factors in Sequencing

Like PCR, the sequencing-by-synthesis process carried out by most current NGS systems is also based on the use of DNA polymerases, which introduces similar coverage bias against genomic regions of extreme GC or AT content. As the use of DNA polymerases is at the core of these technologies, it is difficult to completely eradicate this bias. This bias should be kept in mind though when sequencing genomes or genomic regions of extremely high GC or AT content (>90%). Besides this enzymatic procedure, other aspects of the sequencing process, including equipment operation and adjustment, image analysis, and base calling, may also introduce biases as well as artifacts. For example, air bubbles, crystals, dust, and lint in the buffers could obscure existing clusters (or beads) and lead to the generation of artificial signals. Misalignment of the scanning stage, or even unintended light reflections, can cause significant imaging inaccuracies. Unlike some of the inherent biases mentioned earlier, these artifacts can be minimized or avoided by experienced personnel.

The sequencing signal processing and base calling steps may also introduce bias. For example, on the Illumina platform, the four images generated from each tile after each cycle need to be overlaid (registered), and signal intensities extracted for each cluster and cycle. This procedure is complicated by two factors: (1) signals from the four detection channels are not independent, as there is crosstalk between A and C and between G and T channels, due to the overlapping in the emission spectra of their fluorescent labels; (2) signals from a particular cycle are also dependent on signals from the cycles before and after, due to phasing and prephasing. Although the Illumina's proprietary software is efficient at dealing with these factors for base calling, there are other commercial and open-source tools that employ different algorithms at these tasks and generate varying results. The algorithms these methods use (including the Illumina method) make different assumptions on signal distribution, which may not strictly represent the collected data, and therefore introduce method-specific bias to base calling.

4.5 Major Applications of NGS

4.5.1 Transcriptomic Profiling and Splicing Variant Detection (RNA-Seq)

NGS has gradually replaced microarray as the major means of detecting transcriptomic profiles and changes. The transcriptomic profile of a biological sample (such as a cell, tissue, or organ) is determined by and reflects on its developmental stage, and internal and external functional conditions. By sequencing all RNA species in the transcriptome, NGS provides answers to key questions such as what genes are active and at what activity levels. Transcriptomic studies are almost always comparative studies, contrasting one tissue/stage/condition with another. Besides gene-level analysis, RNA-Seq can also be used to study different transcripts derived from the same gene through alternative splicing. As an integral part of the transcriptome, small RNAs can be similarly studied by NGS. Compared to most DNA-based analyses, RNA-Seq data analysis has its own uniqueness. Analysis of NGS data generated from large and small RNA species is covered in Chapters 7 and 8, respectively.

4.5.2 Genetic Mutation and Variation Discovery

Detecting and cataloging genetic mutation or variation among individuals in a population is a major application of NGS. Existing NGS studies have already shown that severe diseases such as cancer and autism are associated with novel somatic mutations. Projects such as the 1000 Genomes Project have revealed the great amount of genetic variation in a population that accounts for individual differences in physical traits, disease predisposition, and drug response. Chapter 9 focuses on data analysis techniques and how to identify mutations and various types of variations, and test their associations with traits or diseases.

4.5.3 *De novo* Genome Assembly

Sanger sequencing used to be regarded as the golden standard for *de novo* genome assembly, but more and more genomes, including large complex genomes, have been assembled with NGS reads alone. Technological advancements in the NGS arena, including the stable increases of read length in short-read technologies and the development of new NGS technologies that produce very long sequence reads, has contributed to this trend. The development of new algorithms for NGS-based genome assembly is another force behind this progress. Chapter 10 focuses on how to use these algorithms to assemble a new genome from NGS reads.

4.5.4 Protein-DNA Interaction Analysis (ChIP-Seq)

The normal functioning of a genome depends on its interaction with a multitude of proteins. Transcription factors, for example, are among some of the

best-known DNA-interacting proteins. Many of these proteins interact with DNA in a sequence- or region-specific manner. To determine which regions of the genome these proteins bind to, the bound regions can be first captured by a process called chromatin immunoprecipitation (or ChIP) and then sequenced by NGS. ChIP-Seq can be applied to study how certain conditions, such as a developmental stage or disease, affect the binding of protein factors to their affinity regions. ChIP-Seq data analysis is covered in Chapter 11.

4.5.5 Epigenomics and DNA Methylation Study (Methyl-Seq)

Chemical modifications of certain nucleotides and histones provide an additional layer of genome modulation beyond the regulatory mechanism embedded in the primary nucleotide sequence of the genome. These modifications and the modulatory information they provide constitute the epigenome. NGS-based epigenomics studies have revealed how monozygotic twins display differences in certain phenotypes and how changes in the epigenomic profile can lead to diseases such as cancer. Cytosine methylation is a major form of epigenomic change. Chapter 12 covers analysis of DNA methylation sequencing data.

4.5.6 Metagenomics

To study a community of microorganisms like the microbiome in the gut or those in a bucket of seawater, where extremely large but unknown numbers of species are present, a brutal force approach that involves the study of all genomes contained in such a community is metagenomics. Recently, the field of metagenomics has been greatly fueled by the development of NGS technologies. By quickly sequencing everything in a metagenome, researchers can get a comprehensive profile of the makeup and functional state of a microbial community. Compared to NGS data generated from a single genome, the metagenomics data is much more complicated. Chapter 13 focuses on metagenomics NGS data analysis.

5

Early-Stage Next-Generation Sequencing (NGS) Data Analysis: Common Steps

In general, next-generation sequencing (NGS) data analysis is divided into three stages. In the primary analysis stage, bases are called based on deconvolution of the optical or physicochemical signals generated in the sequencing process. Regardless of sequencing platforms or applications, the base call results are usually stored in the standard FASTQ format. Each FASTQ file contains a massive number of reads, which are the sequence readouts of DNA fragments sampled from a sequencing library. In the secondary analysis stage, reads in the FASTQ files are quality checked, preprocessed, and then mapped to a reference genome. The data quality check or control (QC) step involves examining a number of sequence reads quality metrics. Based on data QC results, the NGS sequencing files are preprocessed to filter out low-quality reads, trim off portions of reads that have low-quality base calls, and remove adapter sequences or other artificial sequences (such as polymerase chain reaction [PCR] primers) if they exist. Subsequent mapping (or aligning) of the preprocessed reads to a reference genome aims to determine where in the genome the reads come from, the critical information required for most tertiary analysis (except *de novo* genome assembly). The stage of tertiary analysis is highly application-specific and detailed in the chapters of Section III. This chapter focuses on steps in the primary and secondary stages, especially on reads QC, preprocessing, and mapping, which are common and shared among most applications (Figure 5.1).

5.1 Base Calling, FASTQ File Format, and Base Quality Score

The process of base calling in the primary stage from fluorescence images, movies, or physicochemical measurements is carried out with platform-specific, proprietary algorithms. For example, Illumina uses its proprietary algorithm called Bustard for base calling. The implementation of these base callers may involve multiple steps, which eventually generate for each sequencing cycle a base call and an accompanying confidence score for the call. Most end users do not usually intervene in the base calling process but rather focus on analysis of the base calling results.

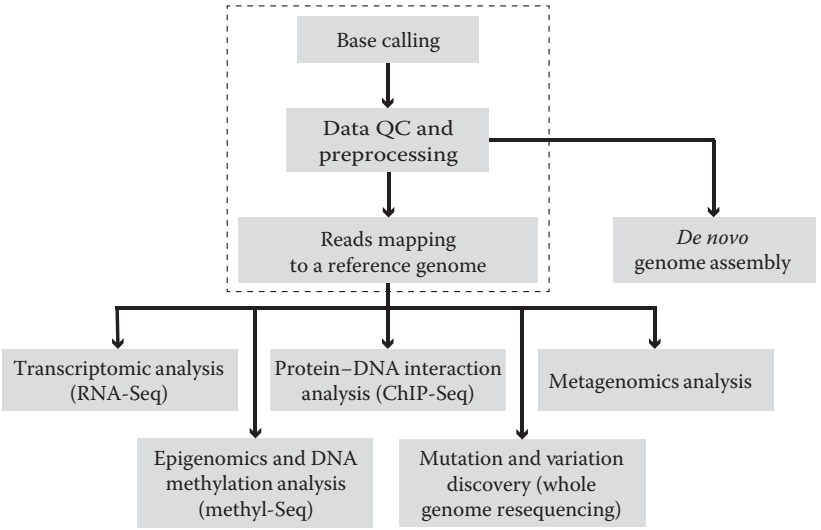

FIGURE 5.1
General overview of NGS data analysis. The steps in the dashed box are common steps conducted in primary and secondary analysis.

Regardless of the sequencing platform, base calling results are usually reported in the universally accepted FASTQ format. While there are other NGS file formats such as FASTA, CFASTA, SFF, and QUAL, FASTQ has become the *de facto* standard for reporting NGS reads data, and all the other formats can be converted to FASTQ using conversion tools (such as NGS QC Toolkit). In file size, a typical compressed FASTQ file is usually in the multigigabyte range and may contain 200 million or more reads. In a nutshell, the FASTQ format is a text-based format, containing the sequence of each read along with the confidence score of each base. Figure 5.2 shows an example of one such read sequence reported in the FASTQ format.

The confidence (or quality) score, as a measure of the probability of making an erroneous base call, is an essential component of the FASTQ format. The NGS base call quality score (Q-score) is similar to the Phred score used in Sanger sequencing and is calculated as

$$Q = -10 \times \log_{10} P_{Err}$$

where P_{Err} is the probability of making a base call error. Based on this equation, a 1% chance of incorrectly calling a base is equivalent to a Q-score of 20, and Q30 means a 1/1000 chance of making a wrong call. Usually for a base call to be reliable, it has to have a Q-score of at least 20. High-quality calls have Q-scores above 30, usually up to 40. For better visualization of

```
@HISEQ:131:C5NWFACXX:1:1101:3848:2428 1:N:0:CGAGGCTGCTCTCTAT
CTTTTATCAGACATATTTCTTAGGTTTGAGGGGGAATGCTGGAGATTGTAATGGGTATGGAGACATATCATATAAGTAATGCTAGG
GTGAGTGGTAGGAAG
+
BB7FFFFB<F<FBFBBFBFBFFFIFFFFIIIFF<FBFFFBFIFFBFFFIFFFBFB07<BFFF7BBFFFBFFFFFF<BFBFBBBBBB
B'77B<770<BBBBB
```

FIGURE 5.2
The FASTQ sequence read report format. Shown here is one read generated from an NGS experiment. A FASTQ file usually contains millions of such reads, with each containing several lines as shown here. Line 1, starting with the symbol "@," contains sequence ID and descriptor. Line 2 is the read sequence. Line 3 (optional) starts with the "+" symbol, which may be followed by the sequence ID and description. Line 4 lists confidence (or quality) scores for each corresponding base in the read sequence (Line 2). For Illumina-generated FASTQ files, the sequence ID in line 1 in its current version basically identifies where the sequence was generated. This information include the equipment ("HISEQ" in the example), sequence run ID ("131"), flow cell ID ("C5NWFACXX"), flow cell lane ("1"), tile number within the lane ("1101"), x-y coordinates of the sequence cluster within the tile ("3848" and "2848", respectively). The ensuing descriptor contains information about the read number ("1" is for the single read here; for the paired-end read it can be 1 or 2), whether the read is filtered ("N" here means it is not filtered), control number ("0"), and index (or sample barcode) sequence ("CGAGGCTGCTCTCTAT").

```
ASCII Character: ! " # $ % & ' ( ) * + , - . / 0 1 2 3 4 5 6 7 8 9 :
  Quality Score: 0 1 2 3 4 5 6 7 8 9 10 11 12 13 14 15 16 17 18 19 20 21 22 23 24 25

 ; < = > ? @ A B C D E F G H I J
26 27 28 29 30 31 32 33 34 35 36 37 38 39 40 41
```

FIGURE 5.3
Encoding of base call quality scores with ASCII characters. ASCII stands for American Standard Code for Information Interchange, and an ASCII code is the numerical representation of a character in computers (e.g., the ASCII code of the letter B is 66). In this encoding scheme, the ASCII character codes are equal to Q-scores plus 33. Current major NGS platforms, including Illumina (after version 1.8), use this encoding scheme for Q-score representation.

Q-scores associated with their corresponding base calls, they are usually encoded with ASCII characters. Although there have been different encoding scheme versions (e.g., Illumina 1.0, 1.3, and 1.5), the NGS field has mostly settled on the use of the same encoding scheme used by Sanger sequencing (Figure 5.3). In the FASTQ example shown in Figure 5.2, the first base, C, has an encoded Q-score of B (i.e., 33).

To come up with the P_{Err}, a control lane or spike control is usually used to generate a base call score calibration table in Illumina sequencing for lookup. A precomputed calibration table can also be used in the absence of a control lane and spike control. Because each platform calibrates its Q-scores differently, if they are to be compared with each other or analyzed in an integrated fashion, their Q-scores need to be recalibrated. To carry out the recalibration, a subset of reads is used that map to regions of the reference genome that contain no SNPs, and any mismatch between the reads and the reference sequence is considered a sequencing error. Based on the rate of mismatch at each base position of the reads, a new calibration table is constructed, which is then used for recalibration. Even without cross-platform NGS data comparison and integration, NGS data generated on the same platform can still be recalibrated postmapping (see Section 5.3) using the same approach, which often leads to improved base call quality scores.

5.2 NGS Data Quality Control and Preprocessing

After NGS data generation, the first step should be a data quality check. Although this step does not directly generate biological insights, it is nonetheless essential and should be carried out carefully. Doing so will avoid production of nonsensical or even erroneous results in later steps and unnecessary consumption of computational resources and time. In this process, the following metrics of data quality need to be examined:

Q-scores—Q-scores can be examined in different ways. On a per-base basis, this process can be conducted by examining quality scores across all base positions of all reads, from the first sequenced base to the last. As a general trend, for platforms based on sequencing-by-synthesis, base positions covered at early phases of a sequencing procedure tend to have higher Q-values than those sequenced later in the procedure. The Q-scores for even the late-phase base positions, however, should still have a median value of at least 20. If there is a significant Q-score drop in the late phase, the affected base positions need to be closely examined and low-quality bases should be trimmed from affected reads. In addition, increased percentage of N calls also helps determine loss of base call quality (an N is called when the base-calling algorithm cannot call any of the four bases with confidence). Another way of inspecting Q-scores is by plotting the average Q-score of each read and examining their distribution pattern. For a successful run, the majority of reads should have average Q-score of over 30, and only a very small percentage of reads have an average Q-score below 20.

Percentage of each base across base positions—If reads are obtained from a sequencing library constructed from randomly generated DNA fragments, the chance of observing each of the four bases at each base position should be constant. Therefore, when plotting the percentages of each base across all base positions, the plots for A, C, G, and T should be roughly parallel to each other, and the overall percentage shown in each plot should reflect the overall frequency of each base in the target library. If the plots deviate significantly from being parallel, this indicates problem(s) in the library construction process, such as existence of overrepresented sequences in the library (such as rRNA in an RNA-Seq library), or nonrandom fragmentation.

Read length distribution—For platforms that produce reads of varying length (such as the Pacific Biosciences platform), the distribution of read length should also be examined. In combination with the distribution of Q-scores, this determines the total amount of useful data a run generates. In addition, with data quality and total volume being equal, a run that produces longer reads is more advantageous in terms of sequence alignment or assembly than one with more relatively short reads.

Besides examining reads quality and length distribution, other QC metrics should also be examined, such as existence of artificial sequences including adapters and PCR primers, or duplicated sequences based on sequence identity (sequence duplication can also be checked based on reference genome mapping results). After inspecting sequence data quality,

filtering should be performed to remove low-quality reads. Furthermore, low-quality base calls (e.g., bases at the 3′ end that have Q-scores below 20), as well as artificial sequence contaminants, should also be trimmed off if they exist. While some platforms (e.g., Illumina) perform sequence filtering by default prior to FASTQ file generation, if the distribution of sequence Q-scores is found to be unsatisfactory after examination, additional filtering/trimming may need to be performed. Execution of these preprocessing tasks is a requirement for high-quality downstream analysis.

The most often used NGS data QC software include FastQC [73], FASTX-Toolkit [74], and NGS QC Toolkit [75]. These toolkits have functional modules to examine per-read and per-base Q-scores, base frequency distribution, read length distribution, and existence of duplicated sequences and artificial sequences. FastQC is written in Java and has a user-friendly interface on most operating systems including Windows. Preprocessing tasks such as filtering and trimming can be carried out by tools like ngsShoRT [76], sickle [77], Trimmomatic [78], and those contained in FASTX-Toolkit and NGS QC Toolkit.

5.3 Reads Mapping

After the data is cleaned up, the next step is to map, or align, the reads to a reference genome if it is available, or conduct *de novo* assembly. As shown in Figure 5.1, most NGS applications require reads mapping to a reference genome prior to conducting further analysis. The purpose of this mapping process is to locate origins of the reads in the genome. Compared to searching for the location(s) of a single or a small number of sequences in a genome by tools such as BLAST, simultaneous mapping of millions of NGS reads, sometimes very short, to a genome is not trivial. A further challenge comes from the fact that any particular genome from which NGS reads are derived deviates from the reference genome at many sites because of polymorphism and mutation. As a result any algorithm built for this task needs to accommodate such sequence deviations. To further complicate the situation, sequencing errors are often indistinguishable from true sequence deviations.

5.3.1 Mapping Approaches and Algorithms

The mapping of NGS reads to a reference genome is not a new task in itself. As indicated earlier, before the advent of NGS, a number of sequence alignment algorithms already existed, the best known of which is BLAST. These aligners use hash tables and seed-and-extend methods to perform the

computationally intensive process of aligning an individual query sequence against a sequence database (such as GenBank). To use these methods to align the millions of NGS reads to a reference genome, however, creates a scalability problem, as they simply cannot scale up to the data volume, and scale down to the short length of many NGS reads (as a short read carries less information). As a result, a new generation of algorithms has been devised for the mapping of NGS reads, either through optimizing the previous methods or introducing new approaches.

Aligners based on the optimization of the previous hash table and seed-and-extend methods include SOAP (Short Oligonucleotide Alignment Program) [79], MAQ (Mapping and Assembly with Qualities) [80], ELAND (Efficient Large-scale Alignment of Nucleotide Databases, Illumina's proprietary aligner), and Novoalign (Commercial). To boost searching speed, indexing of either reference genome sequence or read sequences in the computer memory is used. Among these aligners, SOAP and Novoalign are based on reference genome indexing, whereas MAQ and ELAND perform indexing on NGS reads.

In the seed-and-extend approach used by BLAST, if a match is found between short stretches of nucleotides (called words) in the query sequence and the reference sequence, the matched area is used as a seed to extend the alignment to nearby regions. The seed used by BLAST is a consecutive sequence, which is designed to locate near-exact matches but is not sensitive to sequence variations especially indels. To increase alignment sensitivity, NGS reads aligners have migrated from the use of consecutive exact-match seeds to nonconsecutive (or spaced) seeds (Figure 5.4a). By allowing space between seeds, the chance of finding a match is increased. In SOAP and Novoalign, to perform alignment using spaced seeds, the reference genome sequence is first cut into equal-sized small fragments and saved in a big hash table in memory. The NGS reads are then cut in a similar fashion into subsequences, which are searched against the reference genome. In MAQ and ELAND, the hash table is created from NGS reads, and subsequences extracted from the reference genome are used to look up matches in the reads. Computationally these aligners are memory and processor intensive and therefore not very fast.

To further increase speed and reduce demands on computational resources, a novel approach is developed on the basis of Burrows-Wheeler transform (or BWT) [81] and suffix trees (or arrays) (Figure 5.4b). BWT achieves better reference genome sequence compression to enable more efficient indexing and faster searching. For example, the human genome indexed with BWT only takes 2 to 3 GB of computer memory, whereas the spaced-seed indexing approach can take over 50 GB memory. This newer approach is employed by algorithms such as BWA (Burrows-Wheeler Alignment) [82], Bowtie/Bowtie 2 [83], and SOAP2 [84]. Through the use of BWT and suffix trees (or arrays), the run time needed for aligning million of reads to a large and complex genome, like the human genome, is cut from hours to minutes.

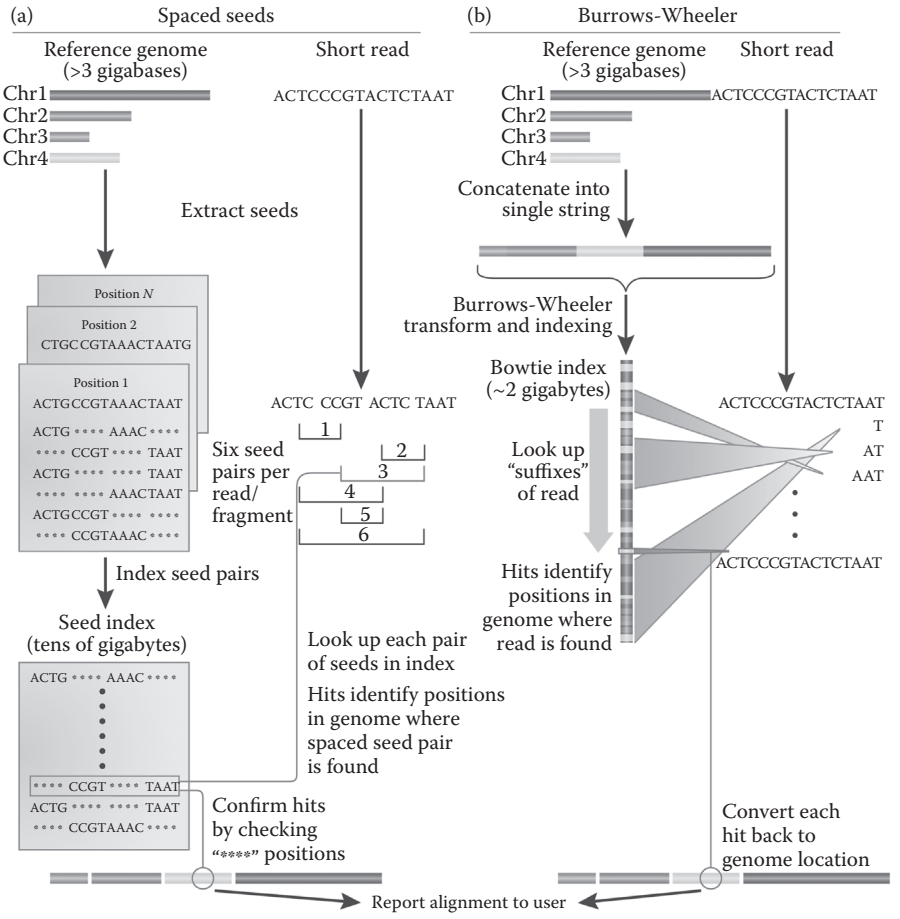

FIGURE 5.4
Two NGS reads mapping approaches. (a) The approach based on spaced seed indexing. In this illustration, spaced seeds extracted from the reference genome sequence are indexed. Some mapping algorithms like MAQ and ELAN index reads (usually in batches) instead. (b) A newer approach developed on the basis of the Burrows-Wheeler transform. In this example, the algorithm Bowtie performs mapping by looking up reads base by base, from right to left, against the transformed and indexed genome. (Modified from C Trapnell, SL Salzberg, How to map billions of short reads onto genomes, *Nature Biotechnology* 2009, 27:455–457. With permission.)

5.3.2 Selection of Mapping Algorithms and Reference Genome Sequences

When selecting aligners, factors including speed and sensitivity need to be considered. As these factors are usually conflicting, some aligners put more emphasis on speed while others stress sensitivity. If speed is a more important factor, Bowtie or SOAP2 is recommended. BWA strikes a balance between speed and sensitivity. If higher sensitivity is preferred, hash-table-based tools

such as Novoalign, Stampy [85], and SHRiMP2 [86] are often used. Most of these aligners were initially developed to map very short reads, such as those of 35 nucleotides from early Illumina sequencers. With the gradual increase in read length, these aligners have been adapted accordingly; for instance, BWA-MEM is a recent adaptation of the original BWA algorithm for aligning longer reads [87].

For aligning much longer reads such as those from the Pacific Biosciences SMRT platform, aligners designed to handle long sequences, such as BLASR [88], LAST [89], LASTZ [90], or BWA-MEM, should be used. Among these long reads aligners, LAST, which uses adaptive seeds instead of fixed-length seeds and suffix array indexing to achieve speed and sensitivity, and LASTZ, which uses the more traditional seed-and-extend approach like BLAST, are whole genome alignment tools originally designed for genome-scale comparisons and therefore can conduct pairwise alignment on very long sequences. BLASR is designed for aligning long reads generated from a single DNA molecule like those from the SMRT system. It conducts mapping of such reads through combining short read mapping data structures and alignment methods used by whole genome alignment tools.

Besides mapping algorithms, selection of reference genome sequences, when multiple reference genome sequences are available, also affects mapping result. By the design of most current mappers, reads that are more similar to the selected reference sequence align better than those that deviate more from the reference. If the deviation is sufficiently large, it might be discarded as a mismatch. As a result, the use of different reference genome sequences can introduce a "reference bias." The use of any one particular reference genome invariably introduces this bias, as a single reference genome simply cannot accommodate sequence variations and polymorphisms that are naturally present in a population or species. This bias should be kept in mind though, especially when the genetic background of the source organism is different from the reference genome. In this situation, comparison of mapping results from the use of different references can help select a reference that is more appropriate. Alternatively, some more recent mapping algorithms, such as GenomeMapper [91], have the capability of using multiple genome references simultaneously as a reference.

5.3.3 SAM/BAM as the Standard Mapping File Format

Mapping results generated from the various algorithms are usually stored in the SAM or BAM file format. SAM, standing for Sequence Alignment/Map, has a tab-delimited text format. It is human readable and easy to examine but relatively slow to parse. BAM, being the compressed binary version of SAM, is smaller in size and faster to parse. Due to their widespread use, SAM/BAM have become the *de facto* standard for storing reads mapping results. The basic structure of a SAM/BAM file is straightforward, containing a header section (optional) and an alignment section. The header section,

TABLE 5.1

Mandatory Fields in the SAM/BAM Alignment Section

Col	Field	Type	Description
1	QNAME	String	Query sequence read (or template) NAME
2	FLAG	Integer	Bitwise FLAG
3	RNAME	String	Reference sequence NAME
4	POS	Integer	Leftmost mapping POSition on the reference sequence
5	MAPQ	Integer	MAPping Quality
6	CIGAR	String	CIGAR string
7	RNEXT	String	Reference name of the NEXT read (For paired-end reads)
8	PNEXT	Integer	Position of the NEXT read (For paired-end reads)
9	TLEN	Integer	Observed Template LENgth
10	SEQ	String	Segment SEQuence
11	QUAL	String	ASCII of Phred-scaled base QUALity+33

if it exists, provides generic information about the SAM/BAM file and is placed above the alignment section. Each line in the header section starts with the symbol "@." For the alignment section there are 11 mandatory fields (listed in Table 5.1). An example of the SAM/BAM format is presented in Figure 5.5.

In the example shown in Figure 5.5, the header section contains two lines. The first line has the two-letter record type code HD, signifying it as the header line, which is always the first line if present. This record has two tags: VN, for format version, and SO, for sorting order (in this case the alignments are sorted by coordinate). The second line is for SQ, which is the reference sequence dictionary. It also has two tags SN and LN, for reference sequence name and reference sequence length, respectively. For the alignment section, while most of the fields listed on Table 5.1 are self-explanatory, some fields may not be so clear at first glance. The FLAG field uses a simple decimal number to track the status of 11 flags used in the mapping process, such as whether there are multiple segments in the sequencing (like r001 in the example) or if the SEQ is reverse complemented. To check on the status and meaning of these flags, the decimal number needs to be converted to its binary counterpart. For the POS field, SAM uses a 1-based coordinate system, that is, the first base of the reference sequence is counted as 1 (instead of 0). The MAPQ is the mapping quality score, which is calculated similarly to the Q-score introduced earlier ($MAPQ = -10 \times \log10(P_{MapErr})$). The CIGAR field describes in detail how the SEQ maps to the reference sequence, with the marking of additional bases in the SEQ that are not present in the reference, or missing reference bases in the SEQ. In the earlier example, the CIGAR field for r001/1 shows a value of "8M2I4M1D3M," which means the first 8 bases matching the reference, the next 2 bases being insertions, the next 4 matching the reference,

```
Coor         12345678901234  5678901234567890123456789012345
Ref          TACGATCGAAGGTA**ATGACATGCTGGCATGACCGATACCGCGACA

+r001/1            CGAAGGTACTATGA*ATG
+r002             cggAAGGTA*TATGA
+r003                        TGACAT..............TACCG
-r001/2                                          ACCGCGACA
```

(a)

```
@HD VN:1.5 SO:coordinate
@SQ SN:ref LN:45
r001   99 ref  7 30 8M2I4M1D3M = 37  39 CGAAGGTACTATGAATG *
r002    0 ref  9 30 3S6M1P1I4M *  0   0 CGGAAGGTATATGA    *
r003    0 ref 16 30 6M14N5M    *  0   0 TGACATTACCG       *
r001  147 ref 37 30 9M         =  7 -39 ACCGCGACA         * NM:i:1
```

(b)

FIGURE 5.5
The SAM/BAM format for storing NGS reads alignment results. The alignment shown in panel (a) is captured by the SAM format shown in panel (b). In panel (a), the reference sequence is shown on the top with the corresponding coordinates. Among the sequences derived from it, r001/1 and r001/2 are paired reads. The bases in lowercase in r002 do not match the reference and as a result are clipped in the alignment process. The read r003 represents a spliced alignment. In panel b, the SAM format contains 11 mandatory fields that are explained in more detail in Table 5.1.

the next 1 being a deletion, and the last 3 again being matches. For more details (such as those on the different FLAG status) and full specification of the SAM/BAM format, please refer to the documentation from the SAM/BAM Format Specification Working Group.

5.3.4 Mapping File Examination and Operation

After carrying out the mapping process, the mapping results reported in SAM/BAM files should be closely examined. First, summary statistics, such as the percentage of aligned reads, especially uniquely mapped reads, should be generated. Currently, the mapping rates are still far from 100%. Even under ideal conditions, most aligners find unique genomic position matches for 70%–75% of sequence reads. This inability to locate the genomic origin of a significant number of reads can be attributed to multiple factors, including the existence of repetitive sequences in most genomes, the relatively short length and therefore limited positioning information of most NGS reads, algorithmic limitation, sequencing error, and DNA sequence variation and polymorphism in a population. The mapping performance is expected to improve with increasing read length from newer NGS technologies and better-designed algorithms from active developments in this area.

Second, reads that map to multiple genomic locations, often called multireads, usually do not contribute to subsequent analysis and therefore are filtered out. The ambiguity in the mapping of multireads is due to the aforementioned sequence deviation caused by polymorphism and mutation, sequencing error, and the existence of highly similar sequences in the genome such as those from duplicated genes. Inclusion of these reads in downstream analysis may lead to biased or erroneous results. For most experiments, these reads should be excluded from further analysis. As filtering of multireads usually removes a significant number of reads, which may lead to potential loss of information, there are some algorithms (such as BM-Map [92]) that are designed to reuse multireads by probabilistically allocating them to competing genomic loci.

Third, besides multireads, duplicate reads should also be identified and filtered out for many experiments. In a diverse nonenriched sequencing library, because of the randomness of the fragmentation process, the chance of getting identical fragments is extremely low. Even with a PCR step to enrich DNA fragments, the chance of generating duplicate reads is still very low (usually <5%), as the number of cycles in the PCR process is limited and the subsequent sequencing process is a random sampling of the DNA library (to varying depth). The existence of excessive numbers of duplicate reads, therefore, suggests PCR overamplification. Duplicate reads can be detected based on sequence identity, but due to sequencing errors this tend to underestimate the amount of duplicate reads. It is more appropriate, therefore, to detect duplicate reads after the mapping step (Figure 5.6). Because technical duplicates caused by PCR overamplification and true biological duplicates are indistinguishable, researchers should exert caution when making decisions on whether to remove duplicate reads from further analysis. Although removing duplicate reads can lead to increased performance in subsequent analysis in many cases (such as variant discovery), in circumstances that involve less complex or mostly enriched sequencing targets, including those from an extremely small genome or those used in RNA-Seq or ChIP-Seq, removing them can lead to loss of true biological information.

Furthermore, a variety of other steps can also be conducted to operate SAM/BAM files. These steps are usually provided by SAMtools and Picard, two widely used packages for operating SAM/BAM files. These operations include

- SAM and BAM interconversion; SAMtools can also convert other alignment file formats to SAM/BAM
- Merging of multiple BAM files into a single BAM file
- Indexing of SAM/BAM files for fast random access
- Sorting reads alignment using various criteria, for example, genomic coordinates, lanes, libraries, or samples

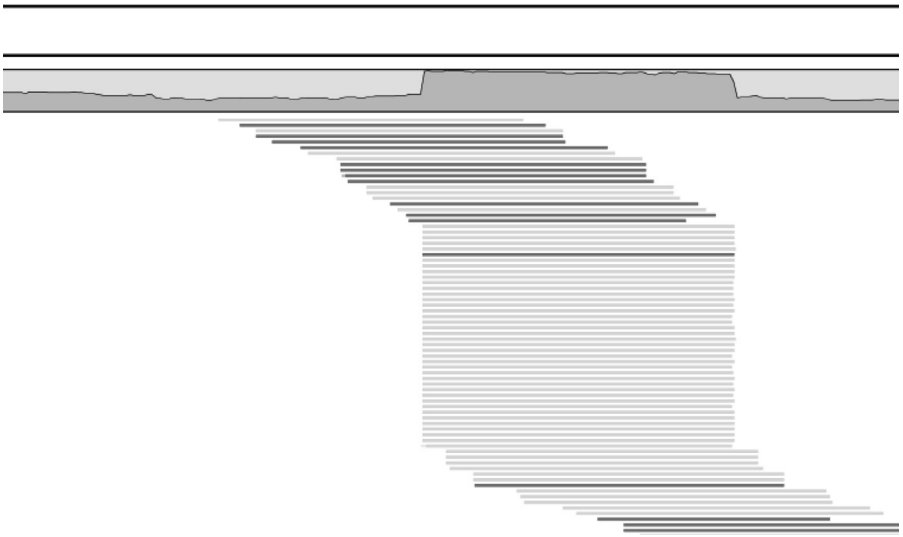

FIGURE 5.6
Detection of duplicate reads after the mapping process. Depth of coverage of the reference genomic region is shown on the top. Mapped reads, along with a set of duplicate reads that map to the same area, are shown underneath. The light and dark gray colors denote the two DNA strands. (Generated with CLC Genomics Workbench and used with permission from CLC bio.)

- Additional reads alignment filtering, such as removing paired reads that only one of the pair can map to the reference genome
- Generation of a pileup format file (Figure 5.7) to show matching (or mismatching) bases from different reads at each genomic coordinate (SAMtools)
- Simple visualization using a text-based viewer for close examination of read alignment in a small genomic region (SAMtools)

SAMtools and Picard are very versatile in handling and analyzing SAM/BAM files. In fact, the steps mentioned earlier, that is, generation of alignment summary statistics and removal of multireads and duplicate reads, can be directly conducted with these tools. For example, both SAMtools and Picard have utilities to detect and remove duplicate reads called rmdup and markduplicates, respectively. These utilities mark reads that are mapped to the same starting genomic locations as duplicates.

Last, in terms of examining mapping results, nothing can replace direct visualization of the mapped reads in the context of the reference genome. While a text-based alignment viewer, such as that provided by SAMtools, offers a simple way to examine a small genomic region, direct graphical

```
ref 181 A 24  ,.$.....,.,.,.,.,.,.,..^+. <<<+;<<<<<<<<<<=<;<;7<&
ref 182 C 23  ,.....,.,.,.,.,.,.,..A <<<;<<<<<<<<<3<=<<<;<<+
ref 183 A 23  ,.$.....,.,.,.,.,.,....    7<7;<;<<<<<<<<<=<;<;<<6
ref 184 G 23  ,$.....,.,.,.,.,.,.,...^l.  <+;9*<<<<<<<<<=<<:;<<<<
ref 185 G 22  ...T,,.,.,.,.,.,.,.,....  33;+<<7=7<<7<&<<1;<<6<
ref 186 C 22  ....,.,.,.A.,.,.,.,..G.  +7<;<<<<<<<&<=<<:;<<&<
ref 187 G 23  .....,.,.,.,.,.,.,.,....^k.   %38*<<;<7<<7<=<<<;<<<<<
ref 188 A 23  C..T,,.,.,.,.,.,.,.,.... ;75&<<<<<<<<<=<<<9<<:<<
```

FIGURE 5.7
The pileup file format as generated from SAMtools. A pileup file shows how sequenced bases in mapped reads align with the reference sequence at each genomic coordinate. The columns are (from left to right) chromosome (or reference name), genomic coordinate (1-based), reference base, total number of reads mapped to the base position, read bases, and their call qualities. In the read bases column, a dot signifies a match to the reference base, a comma to the complementary strand, and "AGCT" are mismatches. Additionally, the "$" symbol marks the end of a read, while "^" marks the start of a read, and the character after the "^" represents mapping quality.

visualization of mapping results by overlaying mapped read sequences against the reference genome provides a more intuitive way of examining the data and looking for patterns. This visualization process serves multiple purposes, including additional data QC, experimental procedure validation, and mapping pattern recognition. Commonly used visualization tools include the Integrative Genomics Viewer (IGV) [93], EagleView [94], and Tablet [95]. The UCSC and Ensembl genome browsers also provide visualization options by adding customized BAM tracks.

5.4 Tertiary Analysis

After the sequence reads mapping step, subsequent analyses vary greatly with application. For example, the workflow for RNA-Seq data analysis is different from that for mutation and variant discovery. Therefore, it is not possible to provide a "typical" workflow for all NGS data analyses in this chapter beyond the common steps of data QC, preprocessing, and reads mapping. Chapters in Section III provide details on application-specific tertiary analytic steps and commonly used tools.

6

Computing Needs for Next-Generation Sequencing (NGS) Data Management and Analysis

The gap between our ability to pump out next-generation sequencing (NGS) data and our capability to extract knowledge from these data is getting broader. To manage and process the torrent of NGS data for deep understanding of biological systems, significant investment in computational infrastructure and analytical power is needed. How to gauge computing needs and build a system to meet the needs, however, poses serious challenges to small research groups and even large research organizations. To meet this unprecedented challenge, the NGS field can borrow solutions from other "big data" fields such as high-energy particle physics, climatology, and social media. For biologists without much training in bioinformatics, while getting expert help is needed, having a good understanding of the various aspects of NGS data management and analysis will be beneficial for years to come.

6.1 NGS Data Storage, Transfer, and Sharing

NGS has itself become a major producer of big data in scientific research. With the continuous drop in sequencing cost, the speed at which NGS data is pumped out will only pick up. This translates into a concomitant increase in the demand for more data storage, access, and processing power. Compared to files generated from other biological assays, such as gel pictures or even microarray data files, NGS files are much larger. For an individual lab, a single typical run generates data at the level of tens or hundreds of gigabytes (GB) in compressed FASTQ format. After aligning to a reference genome, the processed files increase in size appreciably. Further analysis leads to the generation of more and more files and propagation in data volume. On average, analysis of the FASTQ files generates working data files at the level of 500 GB each month by current estimation. To accommodate raw and processed files from multiple runs, tens of terabytes (TB) of storage space is required. Storing and archiving these files are no trivial task. To make the situation even worse, the raw sequencing signal intensity files in formats

such as scanned images or movies are in the scale of terabytes from a single run (this amount is not counted in the data volume mentioned earlier). As these raw signal files accumulate, they can easily overwhelm most data storage systems. While these raw images files can be retained long term, newer sequencing systems process them on-the-fly and delete them by default once they have been analyzed to alleviate the burden of storing them. Oftentimes it is easier and more economical to rerun the samples in case of data loss rather than archiving these huge raw signal files.

Due to the huge size of most NGS files, transferring them from one place to another is nontrivial. For a small-sized project to transfer sequencing files from a production server to a local storage space, download via FTP or HTTP might be adequate if a fast network connection is available. As for network speed, a 1 Gbps network is essential, while a 10 Gbps network offers improved performance for high-traffic conditions. When the network speed is slow or the amount of data to be transferred is too large, the use of an external hard drive might be the only option. When the data reaches the lab, for fast local file reading, writing, and processing, they need to be stored in a hard drive array inside a dedicated workstation or server.

For a production environment, such as an NGS core facility or a large genome center, that generates NGS data for a large number of projects, enterprise-level data storage system, such as Directly Attached Storage (DAS), Storage Area Network (SAN), or Network Attached Storage (NAS), is required to provide centralized data repositories with high reliability, access speed, and security. To avoid accidental data loss, these data storage systems are usually backed up, mirrored, or synced to data servers distributed at separate locations. For large-scale collaborative projects that involve multiple sites and petabytes to exabytes of data, the processes of data transfer and sharing pose more challenges, which prompt the development of high-capacity and high-performance platforms such as Globus.

Data sharing among collaborating groups creates additional technical issues beyond those dealt with by individual labs. A centralized data repository might be preferred over simple data replication at multiple sites to foster effective collaboration and timely discussion. Associated with data sharing also comes the issues of data access control, and privacy for data generated from patient-oriented studies. In a broader sense, NGS data sharing with the entire life science community also increases the value of a research project. For this reason, many journals enforce a data sharing policy that requires deposition before publication of sequence read data and processed data into a publicly accessible database (such as the National Center for Biotechnology Information's [NCBI] Sequence Read Archive [SRA] or the European Nucleotide Archive [ENA]). To facilitate data interpretation and potential meta-analysis, relevant information about such an experiment must also be deposited with the data. Some organizations, such as the Functional Genomics Data Society, have developed guidelines on what information should be deposited with the data. For example, the Minimum

Information about a high-throughput Nucleotide SEQuencing Experiment (MINSEQE) guidelines specify the following information be provided with sequence read data and processed data: (1) description of the biological system, samples, and experimental variables; (2) experimental summary and sample-data relationships; and (3) essential experimental and data processing protocols. Archiving NGS data and associated information for the community is a huge undertaking and requires sizeable investment in maintaining and growing the requisite infrastructure and expert support. The NCBI SRA repository was shut down in 2011 due to high costs and government budgetary constraints. However, because its vital importance to the community, the National Institutes of Health (NIH) resumed its support to SRA later that year.

6.2 Computing Power Required for NGS Data Analysis

Processing the large volume of NGS data requires a lot of computing power. The question of how much computing power is needed is dependent on the type of analysis to be performed. For example, *de novo* assembly of a large genome requires much more computing power than resequencing for variant discovery, or transcriptomic analysis for the identification of differentially expressed genes. Therefore, to determine the computing power needed for a project, a lab, or an organization, the type(s) of NGS work to be performed need to be analyzed first. If the work will require intensive computation, or involve development and optimization of new algorithms and software tools, a high-performance cluster may be needed. On the other hand, if the work will use established workflow that does not require highly intensive computation, a powerful workstation may suffice. It is also advisable that the computer system to be built be scalable to accommodate increases in future computing needs due to unforeseeable change of future research projects or further development of high-throughput genomics technologies.

For a small-sized project, the most basic system needed for NGS data analysis can be simply a 64-bit computer with 8 GB of RAM and two 2 GHz quad-core processors. With such a computer, basic mapping to a reference genome can be performed on obtained sequence reads. This basic setup allows handling of one data set at a time. For simultaneous processing of multiple data sets or projects, high-performance computing (HPC) systems with more memory and CPU cores are needed. The number of cores that an HPC system needs is based on the number of simultaneous tasks to be run at one time. For each task, the number of cores that is needed depends on the nature of the task and the algorithm that carries it out.

Besides the number of CPU cores, the amount of memory a system has also heavily affects its performance. Again memory needs depend on the

number and complexity of jobs to be processed, for example, reads mapping to a small genome may need only a few gigabytes of memory, whereas *de novo* assembly of a large genome may require hundreds of gigabytes or even terabyte-level memory. The current estimation is that for each CPU core the amount of memory needed should not be less than 3 GB. In an earlier implementation of *de novo* assembly of the human genome using the SOAPdenovo pipeline (to be detailed in Chapter 10), a standard supercomputer with 32 cores (eight AMD quad-core 2.3 GHz CPUs) and 512 GB memory was used [96]. As a more recent example of the computing power needed for *de novo* genome assembly, a server with 64 cores (eight Intel Xeon X6550 8-core 2.00 GHz CPUs) and 2 TB RAM is used by a Swedish team [97]. For *de novo* assembly of small genomes such as those of microbes, a machine that contains at least 8 CPU cores, 256 GB of RAM, and a fast data storage system can get a job completed in a reasonable time frame. By current estimation, an 8-core workstation with 32 GB RAM and 10 TB storage can work for many projects that do not conduct *de novo* genome assembly.

The amount of time needed to complete a job varies greatly with the complexity of the job and accessible computing power. As a more concrete example, on a computer with 32 cores and 128 GB RAM, it took <2 hours to map an RNA-Seq data set of 80 million 75 bp reads to the human genome using Bowtie, and less time in subsequent steps including normalization and differential expression statistical tests [98]. In a small RNA NGS study, with a 32-core and 132 GB memory workstation, processing 20 multiplex barcoded samples with a total of 160 million reads took a little over 2 hours for sample demultiplexing, and about the same amount of time for read mapping to the host genome and small RNA annotation databases [99].

6.3 Software Needs for NGS Data Analysis

After a workstation or server is put together from requisite hardware, the operating system and software need to be installed. While some NGS analysis software (such as CLC Genomics Workbench) can operate in the Windows environment, most tools only operate in the Unix (or Linux) environment. Therefore, Unix or Linux is usually the operating system installed on such a machine. Installing software in Unix or Linux is not as straightforward as in Windows, as uncompiled software source code downloaded from a developer site needs to be compiled first before being installed to a particular distribution of the operating system. If the reader is not familiar with the Unix/Linux environment and the command line interface it uses, an introductory book or web-based tutorial is suggested.

One approach to reducing the barrier of using tools developed for the Unix/Linux environment is to access them through a "bridging" system, such as

Galaxy [100], that provides a more user-friendly interface to the command line tools. Developed by the Nekrutenko lab at Penn State and the Taylor lab at Johns Hopkins University, the Galaxy system provides a mechanism to deploy these tools via the familiar web browser interface, making them accessible to users regardless of the operating systems they use. The Galaxy system is highly extensible, with the latest tools being constantly wrapped for execution through the web interface. Besides providing a user-friendly interface, such a system also allows creation of data analysis workflow from different tools, which enables fast deployment of multiple tools in tandem, achievement of consistency and reproducibility, and sharing of analytical procedures with other researchers. Galaxy can be accessed through a publicly available server (e.g., usegalaxy.org), installed on a local instance or in the cloud. With a public server, the user does not need to maintain a local server, but the usable storage space assigned to each account is usually limited and the computing resource is shared with many other users. Creating a local Galaxy instance in Unix/Linux or Mac OS takes some effort and the user does need to provide maintenance, but the user has more control on storage space, computing power, and selection and installation of tools from the entire collection of genomics tools that are made available through the Galaxy Tool Shed. The Galaxy team has made it very easy to install a local instance by offering detailed and easy-to-follow instructions. An instance of Galaxy on the cloud, instantiated via CloudMan [101] on the Amazon Elastic Compute Cloud (EC2), behaves like a local setup but with the flexibility of configuring on a needed basis (see more on cloud computing in Chapter 14).

There are also other community projects that provide alternative platforms to facilitate user access to various NGS and other genomics analysis tools. Bioconductor, an open-source and open-development software project, is among the best known of these projects. This large-scale project is based on R, a programming language and software environment designed for statistical computing and graphics. With the goal of providing tools for the analysis and comprehension of high-throughput genomics data, the recent release (version 3.1) of the Bioconductor software library contains more than one thousand software packages, many of which are designed or can be used to process NGS data. The R environment and the Bioconductor library can be installed in all major operating systems including Windows. The Bioconductor project web portal (www.bioconductor.org) and the R project site (www.r-project.org) provide detailed information and tutorials for the installation and use of these packages. Each tool is well documented with actual use examples provided.

Identifying, installing, and maintaining suitable NGS analysis software from an ever-growing number of tools for a local Unix/Linux workstation, a local Galaxy instance, or a local Bioconductor R library are not trivial. New software tools are constantly being developed and introduced, while many existing ones are updated from time to time. To evaluate candidate packages and identify appropriate tools for installation, it is better to use multiple test

datasets, not just using those from computer simulation but also those from real-world biological samples. In addition, almost all tools have adjustable parameters, which should be set equivalently to facilitate performance comparison. Also in terms of performance, earlier NGS software usually does not take advantage of high-performance parallel computing (more details on parallel computer Chapter 14). To increase performance and take full advantage of the multiple cores or nodes in an HPC system, more recent algorithms tend to use threading or a message passing interface (MPI) to spread the work across multiple processes. Therefore, when evaluating NGS tools, it also helps to examine if these types of parallel processing are employed to take advantage of the power of multicore computing architecture.

6.4 Bioinformatics Skills Required for NGS Data Analysis

For biologists and students in life sciences, acquiring basic bioinformatics skills is greatly advantageous, as biology has become more data rich and data driven. Understanding the basics of bioinformatics also facilitates communication with bioinformaticians on the conduct of more advanced tasks. In general, these skills include use of common computing environments, bioinformatic algorithms, and software packages. Following is a short list of bioinformatics skills required of biologists for NGS data handling.

- Familiarity with Unix/Linux, and the most commonly used commands in the Unix/Linux computing environment. This is essential to operate a local Unix/Linux-based machine or log into a remote server to initiate and monitor jobs, as most genomics servers are Unix/Linux based.
- Basic knowledge of programming languages that are commonly used for NGS data analysis. These languages include R and Perl, both of which are open source, easy to learn, and have a large user base for help and support. While programming is not required of biologists, understanding how an algorithm is executed step-by-step can be helpful, especially when a preexisting tool does not work ideally for a special case and needs modification.
- Knowledge of key concepts in computational biology and biostatistics. Some computational methodologies developed in the field of computer science, especially machine learning and data mining, have been widely applied to high-throughput biological data processing. Artificial neural networks (ANNs), hidden Markov models (HMMs), and support vector machines (SVMs) serve as good examples in this domain. Statistical approaches such as linear and

nonlinear regression are integrated into many genomics data analysis tools and should also be integrated into our knowledge base.

- Basic understanding of a relational database. Most of the information currently available for the annotation and interpretation of NGS data is captured in various databases. Knowledge of database design and structure is the basis to extract, manipulate, and process the information stored in these databases for generation of new biological knowledge. Knowing how to interact with the databases via Standard Query Language (SQL) or Application Programmer Interfaces (APIs) is also beneficial. This knowledge on relational databases and their operation also determines our ability to curate, organize, and disseminate the tremendous amount of information generated from NGS projects.
- Basic understanding and handling of computer hardware such as CPU, RAM, and storage. Although strictly speaking computer hardware is not in the realm of bioinformatics, it is nevertheless advantageous and economical to know how to put together a data server and have it up and running. It is also beneficial to understand how an HPC cluster, or a heterogeneous computing system, works through parallel processing, as NGS tools that are designed to take advantage of these computing systems usually function better and this knowledge can help evaluate and select those that maximize performance built in a server system.

For bioinformaticians who deal with NGS data, on the other hand, the following skills and knowledge are expected:

- Proficiency with Unix-based operating systems
- Familiarity with a programming language such as Python, Perl, Java, or Ruby
- Familiarity with statistical software such as R, MATLAB, or Mathematica
- Understanding of supercomputing, HPC (including parallel computing), and network-based storage
- Knowledge of database management languages such as MySQL or Oracle
- Familiarity with web authoring and web-based user interface implementation technologies
- Understanding of molecular biology, cell biology, and biochemistry

Section III

Application-Specific NGS Data Analysis

7

Transcriptomics by RNA-Seq

7.1 Principle of RNA-Seq

Transcriptomic analysis deals with the questions of which parts of the genome are transcribed and how actively they are transcribed. In the past, these questions were mostly answered with microarray, which is based on hybridization of RNA samples to DNA probes that are specific to individual gene-coding regions. With this hybridization-based approach, the repertoire of hybridization probes, which are designed based on the current annotation of the genome, determines what genes in the genome or which parts of the genome are analyzed, and genomic regions that have no probe coverage are invisible. A next-generation sequencing (NGS)-based approach, on the other hand, does not depend on the current annotation of the genome. Because it relies on sequencing of the entire RNA population, hence the term RNA-Seq, this approach makes no assumption as to which parts of the genome are transcribed. After sequencing, the generated reads are mapped to the reference genome in order to search for their origin in the genome. The total number of reads mapped to a particular genomic region represents the level of transcriptional activity at the region. The more transcriptionally active a genomic region is, the more copies of RNA transcripts it produces, and the more reads it will generate. RNA-Seq data analysis is essentially based on counting reads generated from different regions of the genome.

By counting the number of reads from transcripts and therefore being digital in nature, RNA-Seq does not suffer from the problem of signal saturation that is observed with microarrays at very high values. RNA-Seq also offers a native capability to differentiate alternative splicing variants, which is basically achieved by detecting reads that fall on different splice junctions. Whereas some specially designed microarrays, like the Affymetrix Exon Arrays, can be used to analyze alternative splicing events, standard microarrays usually cannot make distinctions between different splicing isoforms. Also different from microarray signals, which are continuous, raw RNA-Seq signals (i.e., read counts) are discrete. Because of this difference, distribution model and methods of differential expression analysis designed for microarray data cannot be directly applied to RNA-Seq data without modification.

7.2 Experimental Design

7.2.1 Factorial Design

Before carrying out an RNA-Seq experiment, the biological question to be answered must be clear and well defined. This will guide experimental design and subsequent experimental workflow from sample preparation to data analysis. For experimental design, factorial design is usually used. Many experiments compare the transcriptomic profile of two conditions, for example, cancer versus normal cells. This is a straightforward design, involving only one biological factor (i.e., cell type). Experiments involving a single factor may also have more than two conditions, for example, comparison of samples collected from multiple tissues in the body in order to detect tissue-specific gene expression.

If a second biological factor (e.g., treatment of a drug) is added to the example of cancer versus normal cell comparison, the experiment will have a total of four (2 × 2) groups of samples (Table 7.1). In this two-factor design, besides detecting the effects of each individual factor—cell type and drug treatment—the interacting effects between the two factors are also detected, for example, drug treatment may have a larger effect on cancer cells than normal cells. If the factors contain more conditions, there will be a total of $m \times n$ groups of samples, with m and n representing the total number of conditions for each factor. Experiments involving more than two factors, such as adding a time factor to the aforementioned example to detect time-dependent drug effects on the two cell types, are inherently more complex and therefore more challenging to interpret, because in this circumstance it is not easy to attribute a particular gene expression change to a certain factor, or especially, to the interaction of these factors due to the existence of multiple interactions (three factors involve four different types of interactions).

7.2.2 Replication and Randomization

As with any experiment that requires proper statistical analysis, replication and randomization is an essential component of RNA-Seq experimental design. Randomization refers to the random assignment of experimental subjects or targets into each group. This is to avoid introducing unwanted biases to the sample collection process. To generalize the gene expression

TABLE 7.1

Experimental Design Involving Two Biological Factors

	Cancer Cells	**Normal Cells**
Drug treated	Cancer + Drug	Normal + Drug
Vehicle treated	Cancer + Vehicle	Normal + Vehicle

differences observed from groups of samples to the respective populations, within-group variability in the expression of each gene has to be estimated, which requires replication. To meet this requirement at least three replicates need to be included within each group. The more replicates each group has, the more accuracy there is in within-group biological variability estimation, and therefore more certainty to call a gene differentially expressed. While differential gene expression can be detected from unreplicated data, the results are limited to the tested samples and not easily generalizable. Due to the lack of knowledge on biological variation within each group, it is unrealistic to draw conclusions on the population from an unreplicated experiment.

7.2.3 Sample Preparation

Since gene expression is highly plastic and varies greatly with internal (such as tissue and cell type, developmental stage, circadian rhythm, etc.) and external (such as environmental stress) conditions, samples should be collected in a way that minimizes the effects of irrelevant factors. If the influence of such factors cannot be totally avoided, they should be balanced across groups. As many biological samples contain different cell types, this heterogeneity in cell composition is another factor that may confound data interpretation. The use of homogeneous target cells is preferred whenever possible, as this will greatly improve data quality and experimental reproducibility.

To prepare samples for RNA-Seq, total RNA (or messenger RNA [mRNA]) is first extracted from samples of contrasting conditions. As ribosomal RNAs (rRNAs) are usually the predominant but uninformative component in total RNA extractions, rRNA species are usually depleted prior to sequencing. Approaches for rRNA depletion include enrichment of eukaryotic mRNAs that have poly(A) tails with poly(T) primer-based capturing; the Ribo-Zero method based on hybridization and then removal of rRNAs with rRNA-specific RNA probes; degradation by duplex-specific nuclease (DSN), which relies on denaturation-reassociation kinetics to remove extremely abundant RNA species including rRNAs [102]; and RNase H selective depletion based on binding rRNAs to rRNA-specific DNA probes and then using RNase H to digest bound rRNAs. Without rRNA depletion using one of these approaches, signals from low-abundance mRNA transcripts might be masked.

Besides rRNA depletion, degradation of RNA molecules in an extracted sample may also lead to artifactual results. To detect the intactness of RNA molecules in samples, some quality metrics, such as the RNA integrity number (or RIN), are often used. It is recommended to use high-quality RNA samples with no or low levels of degradation, as indicated by high RIN scores, whenever possible. One prerequisite to extracting high-quality RNA is to snap-freeze tissue samples whenever possible to avoid potential RNA degradation. Under circumstances where this is not possible (e.g., sample collection in the field), RNA stabilizing reagent (such as RNAlater) can be used.

For RNA samples prepared under certain circumstances such as those from historical samples or formalin-fixed paraffin-embedded (FFPE) clinical tissues, RNA degradation can be unavoidable. But even from highly degraded RNA samples such as these, useful data may still be generated [103].

Other experimental factors may also have impacts on RNA-Seq data generation and subsequent analysis. Contamination of RNA samples with genomic DNA is one such factor. To remove DNA contaminants, DNase treatment of extracted RNA samples is recommended. In addition, many RNA extraction protocols do not retain small RNA species including microRNAs (miRNAs). If these species are also of interest (more on small RNA sequencing in Chapter 8), alternative protocols (such as the TRIzol method) need to be used. After RNA extraction, an RNA sequencing library needs to be constructed, which basically involves reverse transcription to cDNA and sequencing adapters ligation. This sequencing library construction process may also introduce bias to the subsequent sequencing and data generation. For example, using poly(T) oligonucleotides to enrich for mRNA or prime reverse transcription during this process introduces 3′ end bias, as these procedures are based on the poly-A tail located at the 3′ end of the vast majority of eukaryotic mRNAs. This bias precludes analysis of those mRNAs and other noncoding RNAs that do not have this tail structure [104]. If these RNA species are of interest, the use of alternative rRNA depletion methods and random primers in the reverse transcription step can be employed.

7.2.4 Sequencing Strategy

To facilitate subsequent read alignment to identify their origins in the genome, although single-end long reads will definitely help, use of paired-end but shorter reads works equally well. Besides read length, how to arrange samples on a sequencer in terms of lane assignment can also affect the outcome of an RNA-Seq experiment. On sequencer lane assignment, a balanced block design [105] should be used to minimize technical variation due to lane-to-lane or flow cell-to-flow cell difference. In such a design, samples from different conditions are multiplexed on the same lanes, instead of running different samples or conditions on separate lanes.

An often-asked question on the conduct of RNA-Seq is how many reads should be obtained for an RNA-Seq experiment. The answer on this issue of coverage depth is based on a number of factors, such as the size of the organism's genome, the purpose of the study (quantification of low-abundance genes and alternative splicing variants vs. quick survey of majorly expressed genes), and ultimately statistical rigor (effect size and statistical power). Some RNA-Seq power analysis tools, such as Scotty [106], can be used to help decide on sequencing depth as well as sample size. To start on a species or cell type that has not yet been studied, it might be useful to try out a small number of samples first to get a general idea on the composition of the target transcriptome and the variability between biological replicates.

In general, for human studies, 100 million reads (after filtering, see Section 7.3.1) are needed to detect ~80% of expressed genes; significantly more reads (300 million) are needed, however, to find 80% of differentially expressed genes between conditions. Studying alternative splicing requires more reads due to the increased resolution. It is estimated that 150 million reads are needed to detect 80% of splicing events and 400 million reads for finding 80% of differential splicing events between conditions [107–109]. It should also be mentioned that the detection power of an RNA-Seq study is not only affected by sequencing depth but also the number of sample replicates. Sequencing depth and sample replication provide estimation on gene expression variation at two different levels, with the former sampling RNA fragments in the sequencing library (not every RNA fragment is sequenced) and the latter sampling biological subjects. For projects on budget, it has been reported that increasing the number of biological replicates is more effective in boosting detection power than increasing sequencing depth [110].

7.3 RNA-Seq Data Analysis

7.3.1 Data Quality Control and Reads Mapping

The first step after an RNA sequencing run is to examine the run summary with regard to the total number of reads generated, quality score distribution, GC content, and other indices of the sequencing run, as detailed in Chapter 5. Besides the standard NGS quality control (QC) packages mentioned in Chapter 5, RNA-Seq data QC can also be conducted with those specially designed for RNA-Seq data, including RNA-SeQC [111] and RSeQC [112]. Based on QC results generated from these packages, reads filtering and base trimming can be conducted to remove low-quality reads or base calls. Some other data quality metrics, including percentage of total aligned reads, percentage of rRNA reads, rates of duplicate reads, and genomic coverage, should be examined after reads mapping.

Mapping RNA-Seq reads to a reference genome is more complex than the general reads mapping procedure described in Chapter 5. Because mRNAs are generated from the splicing out of introns and joining of exons, many RNA-Seq reads may not map continuously to the reference genomic sequence. Mapping of these reads, therefore, creates a challenge to the mapping algorithms that are designed to map reads to a reference genome continuously. Two approaches have been developed to meet this challenge. One is to use the current gene exonic annotation in the reference genome to build a database of reference transcript sequences that join currently annotated exons. RNA-Seq reads are then searched against this reference transcripts database using standard nongapped read aligners such as BWA or Bowtie. Examples

of annotation-guided mappers include PASTA [113], RNASEQR [114], RUM [115], SAMMate [116], and SpliceSeq [117]. These mappers may produce better outcome when high accuracy and reliability are emphasized.

The other approach conducts *ab initio* splice junction detection, and therefore does not depend on genome annotation. Depending on their methodology, *ab initio* spliced mappers can be classified into two categories: methods using "exon-first" and those using "seed-and-extend." The exon-first methods include TopHat/TopHat2 [118,119], MapSplice [120], SpliceMap [121], HMMSplicer [122], and GEM [123]. They first align reads to a reference genome to identify unspliced continuous reads (i.e., exonic reads first), and then predict splice junctions out of the initially unmapped reads based on the initial mapping results. Taking TopHat/TopHat2 as an example, they first use Bowtie/Bowtie2 to align reads to the reference genome. Reads that map to the reference continuously without interruption are then clustered based on their mapping position. The clusters, supposedly representing exonic regions, are used to search for splicing junctions from the remaining reads. The seed-and-extend methods, on the other hand, use part of reads as substrings (or k-mers) to initiate the mapping process, followed by extension of candidate hits to locate splicing sites. Examples of methods in this category include GSNAP [124], MapNext [125], SplitSeek [126], and STAR [127]. A hybrid strategy combining the two is also used sometimes, with the exon-first approach employed for mapping unspliced reads and the seed-and-extend approach for spliced reads. As they do not rely on current genomic annotations, these *ab initio* methods are suitable to identify new splicing events and variants.

The percentage of reads that are mapped to the genome is an important QC parameter. Although it is variable depending on a number of factors such as aligning method and species, this number usually falls within the range of 70% to 90%. The percentage of reads that map to rRNA regions is dependent on and a measure of the efficiency of the rRNA depletion step. Due to technical and biological reasons, it is usually impossible to remove all rRNA molecules. The percentage of rRNA reads can vary greatly, from 1%–2% to 35% or more. For downstream analysis, rRNA reads are filtered out so they do not usually affect subsequent normalization. Duplicate reads, a common occurrence in an RNA-Seq experiment, can be caused by biological factors, such as overpresentation of a small number of highly expressed genes, and/or technical reasons, such as PCR overamplification. It is possible to have a high percentage of duplicate reads (e.g., 40% to 60%) in a run. While it is still debatable as to how to treat duplicate reads, because of the biological factors involved in their formation they should not be simply removed. Some experimental approaches, such as removing some of the highly expressed genes prior to library construction, or using paired-end reads, can help reduce the amount of duplicate reads. With regard to genomic coverage, RNA-Seq QC tools often report on the percentage of reads that are intragenic, that is, those

that map within genes (including exons or introns), or intergenic, for those that map to genomic space between genes.

If the species under study does not have a sequenced reference genome against which to map RNA-Seq reads, two approaches exist. One is to map the reads to a related species that has a reference genome, while the alternative is to assemble the target transcriptome *de novo*. The *de novo* assembly approach is more computationally intensive, but it does not rely on reference genomic sequence. Currently available *de novo* transcriptome assemblers include Oases [128], SOAPdenovo-Trans [129], Trans-ABySS [130], and Trinity [131]. These *de novo* assemblers are suited when no related species or only very distantly related species with a reference genome exists, or the target genome, despite the available reference sequence, is heavily fragmented or altered (such as in tumor cells). It should also be noted that if a related reference genome exists with 85% or higher sequence similarity with the species under study, mapping to the related genome may work equally well, or even better, compared to the *de novo* assembly approach. This is especially true when studying alternative splicing variants.

7.3.2 RNA-Seq Data Normalization

As previously mentioned, the basic principle of determining gene expression levels through RNA-Seq is that the more active a gene is transcribed, the more reads we should be able to observe from it. To apply this basic principle to gene expression quantification and cross-condition comparison, at least two factors must be taken into consideration. The first is sequencing depth. If a sample is split into two halves, and one half is sequenced to a depth that is twice of that of the other, for the same gene the former will generate twice as many reads as the latter although both are from the same sample. The other factor is the length of gene transcript. If one gene transcript is twice the length of another gene transcript, the longer transcript will also produce twice as many reads as the shorter one. Because of these confounding factors, prior to comparing abundance of reads from different genes across samples in different conditions, the number of reads for each gene needs to be normalized against both factors using the following formula to ensure different samples and genes can be directly compared:

$$e_{i,j} = \frac{g_{i,j} \times SF}{a_i \times l_j}$$

where $e_{i,j}$ is the normalized expression level of gene j in sample i, $g_{i,j}$ is the number of reads mapped to the gene in the same sample, a_i is the total number of mapped reads (depth) in sample i, and l_j is the length of gene j. SF is a scaling factor and equals to 10^9 when $e_{i,j}$ is presented as RPKM or FPKM

(reads, or fragments [for paired-end reads], per kilobase of transcript per million mapped reads).

The calculation of RPKM or FPKM is the simplest form of RNA-Seq data normalization. In a nutshell, normalization deals with unintended factors and/or technical bias, such as those that lead to unwanted variation in total read counts in different samples. By correcting for the unwanted effects of these factors or bias, the normalization process puts the focus on the biological difference of interest and makes samples comparable. Since the introduction of RKPM or FKPM as an early normalization approach for RNA-Seq data, other methods of normalization have also been developed. Some of these methods employ a similar strategy to adjust for sequencing depth. This group of methods normalize RNA-Seq data through dividing gene read counts by either (1) the total number of mapped reads (i.e., the total count approach), (2) the total read count in the upper quantile (the upper quantile approach) [132], or (3) the median read count (the median approach). These methods do not normalize against gene length, as it is not needed if the goal is to detect relative expression changes of the same genes between groups rather than compare relative abundance levels of different genes in the same samples.

Further normalization approaches are based on the assumption that the majority of genes are not differentially expressed, and for those that show differential expression, the proportion of up- and downregulation is about equal. These include the normalization approaches employed in two commonly used RNA-Seq analysis tools: DESeq and edgeR. In DESeq, normalization is carried out by dividing the read count of each gene in each sample by a scaling factor. To compute the scaling factor for each sample, the ratio of each gene's read count over its geometric mean across all samples is first calculated. After calculating this ratio for all genes in the sample, the median of this ratio is used as the scaling factor. The edgeR package employs a different approach called TMM (trimmed means of M values). In this approach, one sample is used as the reference and others as test samples. TMM is computed as the weighted mean of gene count log ratios between a test sample and the reference, excluding genes of highest expression and those with the highest expression log ratios. Based on the assumption of no differential expression in the majority of genes, the TMMs should be 1 (or very close to 1). If not, a scaling factor should be applied to each sample to adjust their TMMs to the target value of 1. Multiplying the scaling factor with the total number of mapped reads generates effective library size. The normalization is then carried out by dividing the raw reads count by the effective library size, that is,

$$\text{Normalized read count} = \text{Raw read count}/(\text{Scaling factor} \times \text{Total number of mapped reads})$$

The quantile normalization method, originally developed for microarray data, has also been adopted for RNA-Seq data. This method sorts gene read count levels and adjusts quantile means to be equal across all samples, thus

ensuring that all samples have the same empirical distribution. This method is used by the package limma, originally designed for microarray data analysis and now revised for RNA-Seq data [133]. Among other normalization methods are those that use a list of housekeeping genes or spike-in controls as the normalization standard. The use of housekeeping genes or spike-in controls is for conditions in which the assumption that the majority of genes are not differentially expressed might be violated. In this approach, a set of constitutively expressed housekeeping genes that are known to stay unchanged in expression under the study conditions, or a panel of artificial spike-in controls that mimic natural mRNA and are added to biological samples at known concentrations, is used as the basis against which other genes are normalized. Additional methods include those that adjust for putative bias associated with sample-specific GC content [134].

7.3.3 Identification of Differentially Expressed Genes

To compare normalized RNA-Seq gene expression data in different groups and identify differentially expressed genes, the distribution model of the data has to be established first in order to decide on the appropriate statistical tests to be used. While microarray data can be treated as normally distributed variables after log transformation, the RNA-Seq read count values, being discrete in nature, cannot be approximated by continuous distributions even after transformation. In general, count data, including the RNA-Seq data, follows the Poisson distribution, which is characterized by the mean of the distribution being equal to the variance. While this distribution can be and has been used to model RNA-Seq data [132,135], it has also been observed that in RNA-Seq data genes with larger mean counts tend to have greater variance, causing the over-dispersion problem [136] (see Figure 7.1). To deal with this problem, an overdispersed Poisson process, or as an approximation the negative binomial distribution, is often applied. Other distribution models that have been used in RNA-Seq data analysis tools, including the Poisson log-linear model used by PoissonSeq [137] and the normal linear model used by limma [133], have also been found to perform well under many circumstances.

On the identification of differentially expressed genes based on these models, there is a growing list of methods to choose from, among which the commonly used ones are baySeq [138], Cuffdiff/Cuffdiff 2 [139,140], DEGSeq [141], DESeq/DESeq2 [136,142], and edgeR [143]. While DEGSeq has been developed based on the Poisson distribution, baySeq, Cuffdiff/Cuffdiff 2, DESeq/DESeq2, and edgeR have been designed on the negative binomial distribution. To detect differentially expressed transcripts, these packages use different approaches. For example, baySeq employs an empirical Bayesian-based approach, in which two alternative models are proposed for each gene, with one assuming differential expression and another assuming null. Given the observed read counts, the posterior likelihood for the differential expression model is used to identify differentially expressed genes.

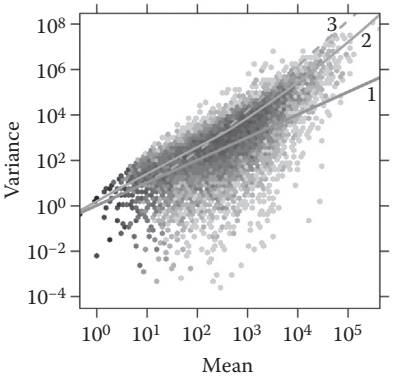

FIGURE 7.1
The overdispersion problem in RNA-Seq data. Poisson distribution is often used to model RNA-Seq data, but instead of the variance/dispersion being approximately equal to the mean as assumed by the distribution, the variance in RNA-Seq data is often dependent on the mean. Line 1 represents the relationship between variance and mean based on the Poisson distribution, and lines 2 and 3 (dashed) represent local regressions used by DESeq and edgeR, respectively, based on negative binomial distribution. (Modified from S Anders, W Huber, Differential expression analysis for sequence count data, *Genome Biology* 2010, 11:R106.)

Cuffdiff/Cuffdiff 2 uses the T statistic, which equals the ratio of mean(log[y]) over variance(log[y]), with y representing the expression ratio of a gene between two groups. Since this statistic approximately follows a normal distribution, a *t*-test is used to identify differentially expressed genes. DEGSeq employs several methods to identify differentially expressed genes, including methods based on the MA-plot, Fisher's exact test, likelihood ratio test, and samrWrapper (a wrapper of functions in SAM, which was originally designed for identifying differential gene expression from microarray data). DESeq identifies differentially expressed genes using a method that is similar to Fisher's exact test for single-factor experiments and a generalized linear model (GLM) based test for multifactor experiments (DESeq2 uses the GLM model for both single- and multifactor experiments). Similarly, edgeR also tests for differential gene expression using an exact test that is highly parallel to Fisher's for experiments with one factor, and the GLM likelihood ratio test for multifactorial experiments.

Packages that are not based on the Poisson distribution or negative binomial distribution are also used for differential expression analysis. For example, limma uses a moderated *t*-statistic to find differentially expressed genes. PoissonSeq conducts differential expression analysis based on tests of a correlation term between gene and experimental conditions, which follows a chi-squared distribution model. The adaptation of SAM for RNA-Seq data analysis has led to the development of SAMseq, which, different from the original SAM, is based on a nonparametric approach [144].

Most of the currently available methods are designed to handle samples with biological replicates. For RNA-Seq without replicates, the method developed by Audic and Claverie, which is also based on the Poisson distribution, is particularly sensitive and suitable [145]. Although it was originally implemented for analyzing relatively small data sets (<10 K reads), the A-C statistic is equally applicable to the much larger NGS data sets that contain millions of reads.

7.3.4 Differential Splicing Analysis

Besides overall expression level changes, eukaryotic genes also undergo alternative splicing to produce different forms of transcripts (see Chapter 3). As differential splicing may exist even in the absence of overall gene expression level changes, analysis of differential splicing adds another dimension to transcriptomic profiling. This analysis involves a number of steps, including reads mapping, inference of splicing events/variants, detection of splicing pattern changes between groups, and identification of differentially expressed splicing variants.

At the reads mapping step, many of the mappers introduced earlier in this chapter, especially the *ab initio* spliced mappers, can be used to map the reads. Inference of individual splicing events, such as skipped exons, alternative 3′/5′ splicing sites, or retained introns, can be performed on the mapped reads using methods such as MISO [146], SpliceTrap [147], and RUM [115]. As an example of how these methods work, RUM generates read counts and RPKM of exons and splicing junctions as well as the entire gene. Assembly and quantification of individual splicing variants are less straightforward because of the uncertainty associated with assigning shared reads to individual variants. There are a number of methods that attempt to achieve this task. ERANGE, for example, assigns reads mapped to known splice junctions to different variants based on gene structure coverage and reports the expression level of each variant as RPKM. Other methods, such as RABT [148], SLIDE [149], and DRUT [150], do not rely on gene annotations, but instead predict and quantify novel splicing variants.

To detect splicing pattern change, methods such as SpliceSeq quantify and compare reads covering exons and splicing junctions to determine splicing pattern changes. Most of these methods carry out gene splicing analysis at the level of splicing events rather than full-length splicing variants, again due to the uncertainty in assigning reads to splicing variants. There are, however, an increasing number of methods, including MISO, ALEXA-Seq [151], FDM [152], rDiff [153], and rSeqDiff [154], that attempt to deal with this uncertainty. These methods assign reads, especially those shared by variants, to different variants by using probabilistic models. For differential analysis of the identified splicing variants, available methods include BASIS [155], BitSeq [156], Cuffdiff2 [140], and EBSeq [157].

For species without a sequenced reference genome, or in cases where RNA transcripts are expected to contain much variation from the reference genome (such as those produced under cancerous conditions), splicing variants can be analyzed using genome-independent methods. Some of these methods are based on mapping to a transcriptome preassembled from RNA-Seq reads. Examples of this transcriptome-based approach include RSEM [158], IsoEM [159], and BitSeq [156]. Additionally, most recently developed *de novo* transcriptome assemblers, including SOAPdenovo-Trans, Oases, Trinity, Trans-ABySS, Rnnotator [160], and KisSplice [161], can also assemble and differentiate splicing variants. These assemblers gather reads into the transcription unit, that is, the set of RNA sequences transcribed from the same gene locus that contain different splicing variants. While these genome-independent methods do not depend on a sequenced reference genome, one major challenge is in distinguishing splicing variants from transcripts derived from closely related genes. For this reason, mapping assembled transcripts to a related reference genome or transcriptome, even from a not-so-closely-related species, often improves accuracy [108].

7.3.5 Visualization of RNA-Seq Data

RNA-Seq data visualization is often needed to appreciate the complexity in gene transcription, including alternative splicing. A growing list of visualization tools has been used to meet this need. Among the most used are the Integrative Genomics Viewer (IGV) and the Integrated Genome Browser (IGB). In addition, RNA-Seq data can be exported as custom tracks for display in a genome browser such as the UCSC Genome Browser. For visualization of alternative splicing, tools like Alexa-Seq, SpliceGrapher, SpliceSeq, and SplicingViewer have their own built-in visualization capabilities. DiffSplice generates GFF-style files that can be visualized in the genomic browser GBrowse.

7.3.6 Functional Analysis of Identified Genes

Once a list of differentially expressed genes is identified, data interpretation is necessary to connect the genes, usually in large numbers, to the biological question under study. Functional analysis of the identified genes is at the core of this process. This analysis can be conducted at multiple levels, including Gene Ontology (GO), biological pathway, and gene network. There are many tools available for analyses at these levels. DAVID [162] is among the best known, which detects enrichment of biological terms in the identified genes, including GO terms and biological pathways. The statistical significance of this enrichment is usually calculated using the hypergeometric distribution, or one-tailed Fisher's exact test. An alternative approach is the Gene Set Enrichment Analysis (GSEA), which instead of using a filtered list of genes, uses the entire gene set for functional analysis [163]. Not relying on

a somewhat arbitrary cutoff for gene selection, the GSEA approach increases sensitivity of the analysis and can pick up weaker signals that might be otherwise missed. For gene network analysis, tools like Cytoscape or Ingenuity Pathway Analysis (IPA, commercial) are often used. Gene network can be reconstructed on the basis of currently available experimental evidence, or coexpression patterns.

7.4 RNA-Seq as a Discovery Tool

Besides interrogating currently cataloged genes, RNA-Seq, being an unbiased approach, is a powerful technology for discovering novel transcripts, splicing events, and other transcription-related phenomena. RNA-Seq studies of the transcriptional landscape of the genome have found that besides protein-coding regions, the majority of the genome produces RNA transcripts. The finding that 75% of the human genome is transcribed (see Chapter 3), made with extensive use of RNA-Seq, shows the power of this technology in discovering currently unknown transcripts. The aforementioned *de novo* alternative splicing variant analysis has also shown its potential in uncovering currently unknown splicing variants. For example, the discovery of circular RNAs (also see Chapter 3), which are formed as a result of noncanonical RNA splicing, is also due to the application of RNA-Seq [164]. RNA-Seq has also been applied to uncover other transcription-related phenomena, such as gene fusion. Gene fusion is caused by genomic rearrangement and is a common occurrence under certain conditions such as cancer. Because RNA-Seq has the capability to locate transcripts generated from a fused gene, detection of gene fusion events has been greatly facilitated by this powerful technology [165].

8

Small RNA Sequencing

Small RNAs play an important role in regulating gene expression in both the cytoplasm and the nucleus through inducing both posttranscriptional and transcriptional gene silencing mechanisms. In additional to RNA interference (RNAi), some studies also show that some small RNAs can increase gene expression via a mechanism called RNA activation (RNAa) [166]. Through these regulatory activities, small RNAs are involved in many cellular processes, affect growth and development, and if their own expression goes awry, lead to diseases such as cancer and Alzheimer's disease.

As introduced in Chapter 3, the major categories of small RNAs in cells include microRNAs (miRNAs), small interfering RNA (siRNAs), and Piwi-interacting RNAs (piRNAs). Among these three types of small RNAs, miRNAs are so far the most studied. A total of 24,521 miRNA loci have been cataloged in 206 species in a recent release of miRBase (version 20), the gold-standard database for miRNAs. It has been estimated that a typical mammalian cell contains hundreds of miRNA species, each of which regulates transcripts from multiple genes. The expression of these miRNAs is cell- and tissue-specific, and dynamically regulated based on cellular state. Mutations or methylations in miRNA genes often lead to dysregulation in their expression. Studying the expression of miRNAs and other small RNAs is an important aspect of studying their roles in biological processes and diseases. Compared to other small RNA expression analysis methods, such as microarray and qPCR, next-generation sequencing (NGS) has a broader dynamic range for measuring small RNAs even at extremely high or low levels, a single-base resolution to differentiate closely related small RNA molecules, the ability to study organisms without a currently available genome assembly, and the capability to discover novel small RNA species.

Concerning new small RNA discovery, although from human and other model organisms the community has cataloged thousands of miRNAs and other small RNA species, more remain to be found. For less studied species, the number of known small RNAs is still low. Many *in silico* miRNA prediction algorithms have been developed, but their predictions have to be validated with experimental evidence. Small-RNA sequencing, through interrogating the entire pool of small RNAs, provides an excellent tool for novel miRNA discovery and experimental validation of computational predictions. Furthermore, small RNA sequencing offers an assumption-free, comprehensive analysis of the small RNA transcriptome in biological targets, including differential expression between conditions. In general, small

RNA sequencing data analysis shares much commonality with the analysis of RNA-Seq data (Chapter 7). In the meantime, some aspects of small RNA sequencing data analysis are unique and mostly focused on in this chapter.

8.1 Small RNA Next-Generation Sequencing (NGS) Data Generation and Upstream Processing

8.1.1 Data Generation

Because sequencing analysis of small RNAs in the transcriptome is similar to messenger RNA (mRNA) analysis, the experimental aspects detailed in Chapter 7 on factorial design, replication and randomization, and sample collection apply equally here and are therefore not repeated. Mature small RNA species, generated as a result of Dicer and Argonaute processing (Figure 8.1, also see Chapter 3, Section 3.4.4.1), have a size range of 18 to 31 nucleotides. Small RNA molecules can be purified from cells or tissues, while total RNA extracts that retain small RNA species works equally well and are often recommended. A size selection step in the sequencing library construction process removes larger RNA molecules in total RNA extracts. Furthermore, the small RNA sequencing library construction process takes advantage of the particular end structure on small RNAs, which are absent on mRNAs. Canonical mature small RNAs have a monophosphate group at the 5′ end and a hydroxyl group at the 3′ end, which is derived from the action of small RNA processing enzymes such as Dicer.

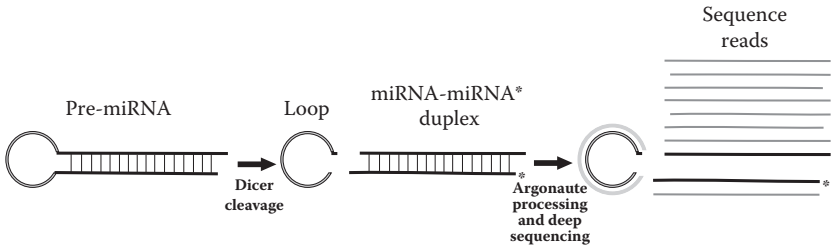

FIGURE 8.1
Deep sequencing of mature small RNAs after Dicer and Argonaute processing. Dicer cleaves a short stem-loop structure out of pre-miRNA to form the miRNA:miRNA* duplex. Upon loading into RISC, Argonaute unwinds the duplex and uses one strand as a guide for gene silencing and discards the other strand (the star strand). Although the short stem-loop and star strand sequences are usually degraded, they may still generate sequencing signals, because of undegraded residues or the fact that they may exist to perform other functions (e.g., the star strand is sometimes functional).

The small RNA sequencing library construction process starts with ligation of adapter sequences to their 3′ and 5′ ends. The universal adapter sequences provide anchoring for subsequent reverse transcription, and then PCR amplification. In the polymerase chain reaction (PCR) step, the number of cycles should be limited to less than 15 (even when the amount of starting material is limited), otherwise library complexity may be reduced leading to a biased result. For multiplexed sequencing, indexing sequences should also be incorporated during the PCR step as part of the PCR primers. Alternatively incorporating indexing sequences during adapter ligation as part of adapter sequence has been found to lead to serious ligation bias [167,168]. After the PCR step, size selection is conducted to purify constructs that carry only small RNAs. Although the library construction process may vary with different sequencing platforms in technical detail such as the use of different adapters and PCR primers, this general workflow is usually followed. It should also be noted that some biases, sometimes unavoidable like in other NGS applications, can be introduced in the library preparation process, for example, some miRNA sequences may be preferentially captured over others, leading to sequence-specific biases [168–170].

Because of their short length, constructed small RNA sequencing libraries do not need to be sequenced for very long. The actual read length depends on the configuration of library constructs and whether the index sequences are read in the same pass or as a separate reading step. In the current version of the Illumina small RNA sequencing protocol that reads index sequences in a second pass, 50 cycles of sequencing can be enough. Sequencing depth is another key factor in the data generation process that determines the power of differential expression analysis and novel small RNA discovery. While this depends on the sample source, as small RNA amount and composition vary greatly with cell type and species, in general 4 million to 5 million raw unmapped reads should offer enough confidence for most studies. A study has shown that coverage higher than 5 million reads contributes little to the detection of new small RNA species [171].

8.1.2 Preprocessing

After obtaining sequencing reads and demultiplexing (if the samples are multiplexed), the reads generated from each sample need to be checked for base call quality using the quality control (QC) tools introduced in Chapter 5 such as FastQC and FASTX-Toolkit. Because small RNA libraries are usually sequenced longer than the actual lengths of the small RNA inserts, the 3′ adapter sequence is often part of the generated sequence reads and therefore should also be trimmed off. The trimming can be carried out with standalone tools such as Cutadapt and Trimmomatic, or utilities in the FASTX-Toolkit and NGS QC Toolkit. Adapter trimming can also be conducted contemporaneously with mapping, as some mappers provide such an option, or using data preprocessing modules within some small RNA data analysis tools (to be covered next).

8.1.3 Mapping

For mapping small RNA sequencing reads to a reference genome, short read aligners introduced in Chapter 5, such as Bowtie/Bowtie2, BWA, Novoalign, or SOAP/SOAP2, can be used. Among these aligners, Novoalign offers the option of stripping off adapter sequences in the mapping command. As for the reference genome, the most recent assembly should always be used. Because of the short target read length, the number of allowed mismatches should be set as 1. To speed up the mapping process, a multithreading parameter, which enables the use of multiple CPU cores, can be used if the aligner supports it. After mapping, reads that are aligned to unique regions are then searched against small RNA databases to establish their identities (see Section 8.1.4), while those that are mapped to a large number (e.g., >5000) of genomic locations should be removed from further analysis.

Besides the aforementioned general tools for small RNA reads preprocessing and mapping, tools have also been developed specially for small RNA analysis, including DSAP [172], miRanalyzer [173], miRDeep/miRDeep2 [174], miRExpress [175], miRNAKey [176], and mirTools [177]. Among these tools, miRanalyzer was among the first developed and is currently one of the most widely used methods. It provides functions for data preprocessing, including 3′ adapter sequence removal, and uses Bowtie for mapping. Both miRDeep2 and mirTools also have modules for data preprocessing, and mapping with the use of Bowtie and SOAP, respectively. DSAP, miRNAKey, and miRExpress all have preprocessing functionalities, but instead of mapping to a reference genome, they map the reads to noncoding RNA (ncRNA) databases including miRBase and Rfam [178]. Rfam is an annotated database for ncRNA families with each family containing a series of RNA sequences that share a common ancestor.

While the mapping of small RNA reads to a reference genome is similar to the mapping in RNA-Seq, as covered in Chapter 7, some characteristics of small RNAs, mostly their short length and posttranscriptional editing, present different challenges from the small RNA reads mapping process. Because of their short length, sizeable numbers of small RNA reads are usually mapped to more than one genomic region. In comparison, this issue is minimal for RNA-Seq data, as longer and sometimes paired-end reads greatly increase specificity. The easiest way to deal with multimapped small RNA reads is to simply ignore them, but this leads to the loss of great amounts of data. A more commonly used approach is to randomly assign them to one of the mapped positions, while an alternative approach is to report them to all possible positions. More sophisticated algorithms have also been developed in an effort to avoid the precision or sensitivity pitfalls of these approaches. For example, one package called Butter (Bowtie UTilizing iTerative placEment of Repetitive small RNAs) [179] makes assignment to one of the possible positions based on the relative local densities of other more confidently assigned reads.

Posttranscriptional editing, on the other hand, leads to the generation of isomiRs [180], which are isoforms of canonical miRNAs that resemble but nevertheless vary from the reference miRNA annotated in miRBase. The isomiRs have various forms of variations from the canonical sequence, including alternative 3′ (more often) and 5′ termini, and nucleotide substitutions in the body sequence. Since their discovery, which itself is attributed to small RNA sequencing, isomiRs have been shown to have physiological significance [181,182]. Because the discovery of isomiRs is very recent, most small RNA mappers still only count miRNAs with exact matches or small variations to the miRBase-cataloged reference of mature miRNA sequences. More recently developed tools, such as miRSeq [183] and SeqBuster [184], have begun to cover isomiRs.

8.1.4 Identification of Known and Putative Small RNA Species

To identify currently known small RNA species, the mapped reads need to be searched against the most recent version of the miRBase or other small RNA databases (such as piRNABank). Reads with no matches in these databases can then be searched against other databases (Rfam, repeat, and mRNA) to determine if they are degradation products of ncRNAs, genomic repeats, and mRNAs. The previously mentioned tools, that is, DSAP, miRanalyzer, miRDeep/miRDeep2, miRExpress, miRNAKey, and mirTools, all provide these database search capabilities.

To discover potentially novel miRNA species, mapped reads that do not match known miRNAs and sequences in the other databases are submitted to algorithms such as miRanalyzer and miRDeep2, which are designed to search for putative miRNAs. A machine learning approach based on the random forest classifier is used by miRanalyzer to classify the reads and make predictions. The approach used by miRDeep2 takes into consideration the biogenic process of miRNAs. It first identifies potential miRNA precursor coding regions out of the genomic regions that are clustered with the mapped reads. RNA secondary structures are then predicted on these identified regions using RNA folding software, and examined to see if they resemble a typical miRNA hairpin structure seen in pri-miRNA molecules and if they are thermodynamically stable. Putative miRNA species are called if the reads fall into stable hairpins in an expected manner, along with other evidence such as reads from the star strand.

8.1.5 Normalization

Before identifying differentially expressed small RNAs, read counts for each small RNA species in the samples need to be normalized. The goal of normalization is to make the samples directly comparable by removing unwanted sample-specific variations, which are usually due to differences in library size and therefore sequencing depth. The normalization approaches used in

RNA-Seq as detailed in Chapter 7 can be similarly applied here. The general assumption for most of the normalization approaches, that the majority of small RNAs stay constant between conditions, seems to hold. For the total read-count-based normalization, since all small RNAs are similarly short in size, the RPKM (reads per kilobase of transcript per million mapped reads) normalization can be simplified as RPM (reads per million). This popular method, however, has been found to be inadequate in some benchmark studies [185,186]. Other normalization approaches, including the DESeq, quantile, or LOWESS methods, were found to have better performance in these studies.

8.2 Identification of Differentially Expressed Small RNAs

The packages and tests introduced for RNA-Seq differential expression analysis in Chapter 7 can also be directly used for small RNA analysis. For experiments without replicates, the Audic-Claverie methods can be used. For those with replicates, DESeq and edgeR, along with the other tools introduced earlier, work well on the identification of differentially expressed small RNAs. Because of their good performance, these tools are also often used by packages particularly designed for miRNA-Seq data analysis. For example, miRanalyzer applies DESeq for its differential expression analysis, and mirTools uses the Audic-Claverie method.

8.3 Functional Analysis of Identified Small RNAs

To perform functional analysis of differentially expressed small RNAs, their gene targets need to be predicted first. A number of tools are available for this task, including miRanda [187], mirSVR [188], PicTar [189], PITA [190], RNA22 [191], RNAhybrid [192], TargetScan [193], and the DNA intelligent analysis (DIANA) application microT-CDS [194] or microT [195]. These tools predict target genes based on base-pairing pattern, thermodynamic stability, and sequence conservation. For example, miRanda makes predictions based on the miRNA–mRNA complementarity pattern, location of the binding site in the mRNA, binding energy, and miRNA evolutionary conservation. On miRNA target gene prediction, it should also be noted that the predictions generated from the aforementioned tools have certain levels of false positives, as well as false negatives, as miRNA target gene prediction is no easy task because of the small size of the miRNA-mRNA binding area, often imperfect complementarity of the binding, and sometimes lack of conservation [196].

Once a list of potential target genes are generated, functional analysis, such as Gene Ontology (GO) and pathway analysis, can be conducted using the approaches detailed in Chapter 7. In addition, for pathway analysis, a list of miRNAs can also be uploaded directly to the DIANA miRPath web server to a generate a list of biological pathways that are significantly enriched with the miRNAs' target genes, which are predicted with DIANA-microT-CDS or documented with existing experimental evidence [197].

9

Genotyping and Genomic Variation Discovery by Whole Genome Resequencing

Detection of genomic variation among individuals of a population is among the most frequent applications of next-generation sequencing (NGS). Genome sequence heterogeneity is prevalent in a naturally occurring population, which cannot be captured by the current use of a single reference genome for a species. Genomic variant cataloging projects, such as the 1000 Genomes Project and the 100,000 Genomes Project, underscore the importance of genomic variation discovery. Locating genomic sequence variations that correlate with disease predisposition or drug response, and establishing a genotypic basis of various phenotypes have become common focuses of many NGS studies in biomedical and life science research. Besides variations carried through the germline for generations, NGS has also been applied to identify *de novo* germline and somatic mutations, which occur more frequently than previously expected and underlie numerous human diseases including various types of cancer [198,199].

Detecting the various forms of genomic variations/mutations from NGS data, as detailed in Chapter 2, including single nucleotide variations (SNVs), indels, and structural variations (SVs), is not an easy task. The primary challenge is to differentiate true sequence variations/mutations from false positives caused by sequencing errors and artifacts generated in base calling and sequence alignment. It is, therefore, important to generate high-quality sequencing data before performing data analysis. Equally important, sensitive and yet specific variant/mutant calling algorithms are required to achieve high accuracy in genomic variation and mutation discovery. This chapter first provides details on data preprocessing, alignment, realignment, and recalibration. It then focuses on methods for the detection of SNVs/indels and SVs, followed by variant annotation, and finally testing of variant association with diseases or phenotypic traits. Figure 9.1 shows an overview of the data analysis pipeline.

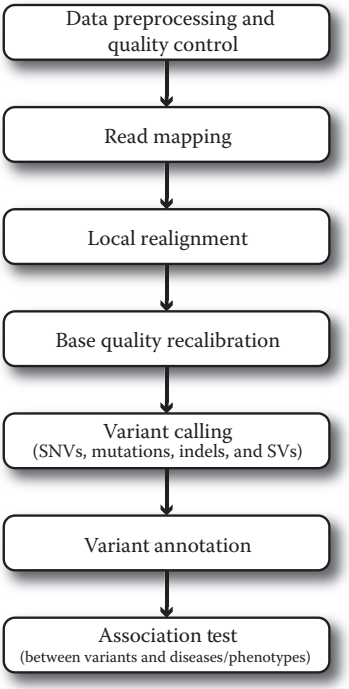

FIGURE 9.1
General workflow for genotyping and variation discovery from resequencing data.

9.1 Data Preprocessing, Mapping, Realignment, and Recalibration

Besides the general data preprocessing and quality control steps introduced in Chapter 5, such as examining sequencing data quality, removing low-quality and duplicate reads, additional steps are needed for variant calling. The reads mapping step requires the use of a highly sensitive alignment algorithm, such as BWA, Novoalign, Stampy, MOSAIK, or BFAST (particularly good for aligning SOLiD-generated color space sequence reads). After examining mapping quality, reads with low-quality mapping scores need to be filtered out. For paired-end reads, they should map to the reference genome as pairs at the expected interval and those that do not show the expected pattern should be filtered out as well.

After the initial alignment, realignment around indels usually leads to improvement in mapping results. This is usually due to the fact that short indels, especially those at the ends of reads, often cause problems in the initial alignment process. To realign around the indel regions, the original BAM

file is first processed to identify where realignment is needed using tools such as the GATK RealignerTargetCreator. In this process, using a known set of indels (such as those in dbSNP, or those cataloged by the 1000 Genomes Project) can speed the process and improve accuracy. After the target regions for realignment are identified, programs such as the GATK IndelRealigner can be employed to conduct the realignment. At the end of this process, a new BAM file is generated containing realigned reads.

Prior to variant calling, the original base-call quality scores should also be recalibrated to further improve data quality. This base quality score recalibration can be conducted with tools such as the GATK BaseRecalibrator, which recalibrates raw quality values using a covariate-aware base quality recalibration algorithm. This algorithm adjusts for covariates, such as the machine sequencing cycle and local sequence context, that are known to affect sequencing signal and base-call quality. To carry out the recalibration, the covariation pattern is first analyzed, which is examined and then applied to recalibrate the data. Variant calling based on the recalibrated data has higher accuracy and cuts down on the number of false positives.

9.2 Single Nucleotide Variant (SNV) and Indel Calling

9.2.1 SNV Calling

In general, variant calls are made based on a number of factors (Figure 9.2). These factors include (1) base call quality, (2) mapping quality, (3) single versus paired-end sequencing, (4) read length, (5) depth of coverage, and (6) sequence context. Because of errors or uncertainties that occur in the steps of sequencing, base calling, and mapping, there are almost always certain levels of uncertainty associated with each variant call. To minimize this uncertainty,

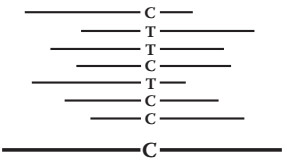

FIGURE 9.2
The variant calling process is usually affected by various factors. In this illustration, a number of reads are aligned against a reference sequence (bottom). At the illustrated site, the reference sequence has a C, while the reads have C and T. Depending on the factors mentioned in the text and prior information, this site can be called heterozygous (C/T), or no variation (C/C) if the T's are treated as errors. It is also possible to be called a homozygous T/T, if the C's are regarded as errors.

base-calling algorithms use statistical models or heuristics. By modeling the errors and biases, and sometimes incorporating other related prior information, variant callers that use statistical models significantly reduce the probability of miscalling variants. For methods that are based on the heuristic approach, on the other hand, the call variants are based on a number of heuristic factors, such as minimum read depth, base quality, and allele frequency. Algorithms based on statistical models are currently more widely used than those based on heuristics. It should be noted, however, that statistical models are usually based on certain assumptions. Under circumstances when the assumptions are violated, the heuristic methods can be more robust.

Among the tools that are based on statistical models, GATK [200] and Samtools [201] are currently among the most widely used. In its current version, GATK offers two variant callers, UnifiedGenotyper and HaplotypeCaller. UnifiedGenotyper uses a Bayesian genotype likelihood model to call variants (SNPs and indels separately) and genotypes (i.e., as A/A, A/B, or B/B). This caller is fast and considers each locus independently. HaplotypeCaller, as the name suggests, considers the linkage between nearby variants, and calls SNPs and indels simultaneously. It performs local *de novo* assembly of haplotypes and is therefore more computationally intensive and slower. Samtools uses the same genotype likelihood model for variant calling, which is achieved in two steps, namely, mpileup and Bcftools. In the mpileup step, it collects summary information from input BAM files and computes the likelihoods of possible genotypes, which are stored in BCF files. The subsequent Bcftools step uses the likelihood information in the BCF files to conduct variant calling.

Besides GATK and Samtools, other model-based variant callers include SOAPsnp [202], a component of the SOAP tool package, and Atlas 2 for variant analysis of exome sequencing data [203]. SOAPsnp takes a similar Bayesian modeling approach to identify SNPs. Atlas 2 is based on logistic regression models that are validated with whole exome sequencing data. A commonly used heuristics-based variant caller is VarScan/VarScan2 [204], which works more robust on data confounded by factors such as extreme read depth, pooled samples, and contaminated or impure samples. Since these different types of tools use different approaches for variant calling, the variants they identify are usually only partially overlapping. It is advisable, therefore, to closely examine the specifics of an experiment to decide on more appropriate variant caller(s). If more than one method can be used, it is advisable to compare their outputs and analyze how they intersect. Use of convergent variants is an effective way to reduce rates of miscalled variants.

Some of these tools, such as GATK, work for both single- and multiple-sample data. Multiple-sample analysis usually has more detection power than single-sample analysis. This is because with multiple samples it is more likely to call a variant when more than one sample shows the same variation. Therefore, to improve variant call quality it is usually better to conduct the calling on multiple samples.

9.2.2 Identification of *de novo* Mutations

Most of the currently available variant-calling methods are designed to identify variations that are passed from generation to generation. Although these variants are a major target of genomic variation studies, *de novo* mutations in somatic and germline cells (Figure 9.3) also play important roles in many diseases and altered phenotypes. To identify these *de novo* mutations, though some of the variant callers mentioned earlier such as VarScan can be used, there exist some specifically designed algorithms, including MuTect [205], SomaticSniper [206], Strelka [207], and JointSNVmix [208], on the comparison of parent–offspring or normal–diseased samples. Mechanistically, while some of these algorithms (such as MuTect and VarScan) carry out mutation calling on each of the contrasting samples separately against a reference genome, others (such as JointSNVmix, Strelka, and SomaticSniper) directly compare the contrasting samples. In the former approach, sequence reads generated from contrasting samples (e.g., normal versus cancer tissues

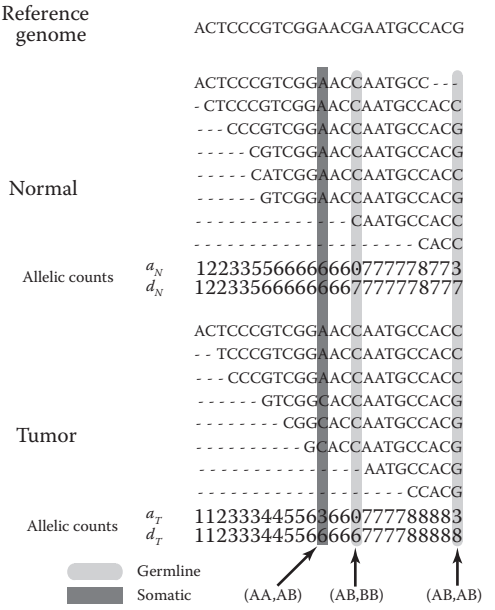

FIGURE 9.3
De novo somatic mutations versus inherited germline variations. In this example, sequence reads from normal and tumor tissues are aligned to the reference genome (shown at the top). The allelic counts, that is, the number of matches (a_N and a_T) and depth of reads (d_N and d_T), at each base position are shown. The light gray sites indicate germline variants, while the dark gray indicates a *de novo* somatic variant acquired in some tumor cells. Also shown at the bottom are the predicted genotypes for the normal and tumor tissues. (Modified from A Roth, J Ding, R Morin, A Crisan, G Ha, R Giuliany, M Hirst et al., JointSNVMix: A probabilistic model for accurate detection of somatic mutations in normal/tumour paired next-generation sequencing data, *Bioinformatics* 2012, 28 (7):907–913. With permission.)

carrying somatic mutations from the same patient) are independently aligned to and variants called against a reference genome. The called variants in the contrasting samples are then compared to each other to locate somatic mutations in the cancer tissue. In the latter approach, the samples are directly compared to each other using statistical tests on the basis of joint probability.

9.2.3 Indel Calling

Calling small indels (large indels are covered in Section 9.3 on SV calling), which occur at a frequency of about 1 in 8000 bp in the genome, is more challenging than calling SNVs. This is because the existence of indels in a read could interfere with the read's accurate mapping. The mapping for small indel calling, therefore, should allow insertions or deletions that involve a few bases. After mapping, a simple approach to indel calling is to extract insertion and deletion information from the sorted BAM file using Samtools (varFilter). This approach, while simple, often shows high false-positive and false-negative rates. More complex approaches, such as Dindel [209] or GATK, can be used for improved performance. The basic workflow of these approaches is (1) scan the input BAM file for insertions and deletions; (2) for each indel site, build a new haplotype based on the indel event; (3) realign all sequence reads to the newly created alternative haplotype; (4) count the number of reads that support the indel in the alternative haplotype; and, finally, (5) make indel calls.

Another approach that addresses the challenge of calling indels is based on local *de novo* assembly. SOAPindel [210] is an example of this approach. With the use of paired-end reads, this approach first identifies mapped reads that have unmapped mates, and then positions the unmapped reads at their expected genomic locations. A local *de novo* assembly is subsequently built with high density of such unmapped reads and aligned to the reference to identify indels. This approach is computationally more intensive, especially when the reference genome is large.

9.2.4 Variant Calling from RNA-Seq Data

While variant calling is mostly carried out from DNA sequencing data, RNA-Seq can also be used to call variants from transcriptionally active regions of the genome. RNA-Seq-based variant calling is more challenging due to the inherent heterogeneity in the abundance of reads transcribed from different regions and the splicing of exons. Variant calling from RNA-Seq data offers certain advantages, however, as it does not incur additional cost beyond collecting the original transcriptomic data, and it directly interrogates transcriptionally active regions of the genome. In addition, RNA-Seq-based variant discovery can be used to validate variants called from whole-genome or whole-exome sequencing. Methods for RNA-Seq-based variant calling are still limited. Currently available tools, including eSNV-Detect [211], SNPiR [212], and SNVMix [213], employ different models for variant calling. For

example, SNVMix employs a probabilistic binomial mixture model to call variants from premapped RNA-Seq reads.

9.2.5 Variant Call Format (VCF) File

Variant call format (VCF) is a standard file format for storing major types of sequence variation, including SNVs, indels, and SVs [214]. This format is designed to be scalable to encompass millions of sites from thousands of samples. Originally developed for the 1000 Genomes Project, it is designed for fast data retrieval. Besides reporting variants and their genomic positions, it allows fields to store additional information such as variant-call quality score and allows users to add their own custom tags to describe new sequence variations thereby offering flexibility.

Figure 9.4 provides an example of the VCF. It contains metainformation lines at the front, a header line, and data lines, each of which describes a variant position. The metainfo lines start with "##" and describe related analysis information, such as species, file date, and assembly version. In addition, abbreviations used in the user definable data columns are also defined in the metainfo lines. The subsequent header line lists the names of the eight mandatory columns (Table 9.1). In the QUAL column, a Phred-like quality score for the alternative allele (ALT) call is given (e.g., a QUAL value of 30 means the probability of the ALT call being wrong is 0.001). In the FILTER column, "PASS" means this position has passed all filters, while a value of "q10" as shown in Figure 9.4 indicates that the variant-call quality at this site is below 10. The data lines, containing variant calls for a list of genomic positions, make the body of a VCF file.

VCF files can be parsed and manipulated using tools such VCFtools [214] or vcflib [215]. VCFtools, for example, is a tool kit containing various utilities for VCF file parsing, analysis, and manipulation. It consists of two modules: a general Perl API and a C++ binary executable. The Perl module can be used for

```
##fileformat=VCFv4.2
##fileDate=20090805
##source=myImputationProgramV3.1
##reference=file:///seq/references/1000GenomesPilot-NCBI36.fasta
##contig=<ID=20,length=62435964,assembly=B36,md5=f126cdf8a6e0c7f379d618ff66beb2da,species="Homo sapiens",taxonomy=x>
##phasing=partial
##INFO=<ID=NS,Number=1,Type=Integer,Description="Number of Samples With Data">
##INFO=<ID=DP,Number=1,Type=Integer,Description="Total Depth">
##INFO=<ID=AF,Number=A,Type=Float,Description="Allele Frequency">
##INFO=<ID=AA,Number=1,Type=String,Description="Ancestral Allele">
##INFO=<ID=DB,Number=0,Type=Flag,Description="dbSNP membership, build 129">
##INFO=<ID=H2,Number=0,Type=Flag,Description="HapMap2 membership">
##FILTER=<ID=q10,Description="Quality below 10">
##FILTER=<ID=s50,Description="Less than 50% of samples have data">
##FORMAT=<ID=GT,Number=1,Type=String,Description="Genotype">
##FORMAT=<ID=GQ,Number=1,Type=Integer,Description="Genotype Quality">
##FORMAT=<ID=DP,Number=1,Type=Integer,Description="Read Depth">
##FORMAT=<ID=HQ,Number=2,Type=Integer,Description="Haplotype Quality">
#CHROM POS     ID        REF    ALT     QUAL FILTER INFO                              FORMAT      NA00001        NA00002        NA00003
20     14370   rs6054257 G      A       29   PASS   NS=3;DP=14;AF=0.5;DB;H2           GT:GQ:DP:HQ 0|0:48:1:51,51 1|0:48:8:51,51 1/1:43:5:.
20     17330   .         T      A       3    q10    NS=3;DP=11;AF=0.017               GT:GQ:DP:HQ 0|0:49:3:58,50 0|1:3:5:65,3   0/0:41:3
20     1110696 rs6040355 A      G,T     67   PASS   NS=2;DP=10;AF=0.333,0.667;AA=T;DB GT:GQ:DP:HQ 1|2:21:6:23,27 2|1:2:0:18,2   2/2:35:4
20     1230237 .         T      .       47   PASS   NS=3;DP=13;AA=T                   GT:GQ:DP:HQ 0|0:54:7:56,60 0|0:48:4:51,51 0/0:61:2
20     1234567 microsat1 GTC    G,GTCT  50   PASS   NS=3;DP=9;AA=G                    GT:GQ:DP    0/1:35:4       0/2:17:2       1/1:40:3
```

FIGURE 9.4
The VCF format (version 4.2). (From http://samtools.github.io/hts-specs/; the format is currently managed by the Global Alliance Data Working Group File Formats Task Team.)

TABLE 9.1

Mandatory Fields in a VCF File

Col	Field	Type	Description
1	#CHROM	String	Chromosome number
2	POS	Integer	Start position of the variation
3	ID	String	Database identifier
4	REF	String	Reference allele
5	ALT	String	Alternate allele(s)
6	QUAL	Numeric	Quality score (Phred-style)
7	FILTER	String	Filter status
8	INFO	String	User extensible information

routine tasks such as VCF file validation, merging, intersecting, complements, and so forth. The binary executable provides tools for generating various QC metrics, filtering out specific variants, summarizing the variants, estimating allele frequencies, calculating levels of linkage disequilibrium, and so on.

9.2.6 Evaluating VCF Results

SNVs and indels reported in VCF files need to be evaluated in order to identify false positives. Visualization of called variants and supporting reads in a genome browser, such as IGV or Savant, provides an initial examination of the variant call result. Further evaluation should be based on criteria such as deviation from Hardy–Weinberg equilibrium, systematic call quality difference between major and minor alleles, extreme depth of coverage, or strand bias. The ratio of transitions and transversions (Ti/Tv) is an additional indicator of variant call specificity and quality. The theoretical ratio of Ti/Tv is 0.5, because purely from the point of statistical probability the chance of producing transitions is half that of transversions. However, due to biochemical mechanisms involved in these nucleotide substitution processes, the frequency of having transitions is higher than that of transversions. Based on existing NGS data from multiple species, the expected values of Ti/Tv for whole genome and exome data sets are usually in the ranges of 2.0 to 2.1 and 3.0 to 3.5, respectively [216].

9.3 Structural Variant (SV) Calling

9.3.1 Read-Pair-Based SV Calling

Earlier experimental methods on the detection of SVs were mostly based on comparative genome hybridization and SNP whole genome arrays. The advent

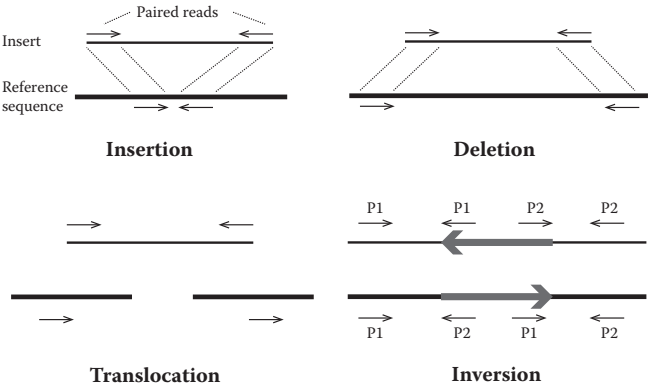

FIGURE 9.5
Common SVs and the basic approach to detect them using paired reads.

of NGS, especially the use of paired-end reads, has greatly pushed SV detection forward. As illustrated in Figure 9.5, the basic approach to locate large indels, inversions, and translocations is based on changes in orientation or distance between paired reads. SV detection algorithms that employ this general approach include BreakDancer [217], GASV [218], HYDRA [219], PEMer (Paired-End Mapper) [220], and SVDetect [221]. Figure 9.6 shows the general algorithmic procedure for calling SVs using this approach. The first step is to separate read pairs into concordant and discordant groups, defined by the distance between a read pair matching or deviating from the expected distance based on the reference genome. The discordant read pairs are then assembled into different clusters based on the genomic region they cover to generate candidate SV calling regions. In the last step, the candidate SV clusters are filtered based on statistical assessment so that only clusters that are covered by multiple read pairs are reported as SVs. The bounds on possible breakpoints in the region are also identified in this step (indicated by the shaded area in Figure 9.6d).

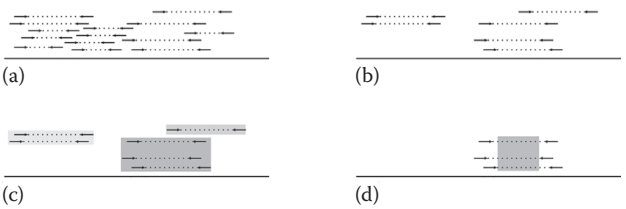

FIGURE 9.6
General steps of calling SVs using paired-end reads. (a) Paired reads are mapped to the reference genome. (b) Discordant read pairs are identified. (c) Discordant read pairs are assembled into clusters. (d) Candidate clusters of discordant read pairs are filtered to identify SVs, and bounds on possible breakpoints identified. (From C Whelan, Detecting and analyzing genomic structural variation using distributed computing, 2014, *Scholar Archive*, Paper 3482. With permission.)

9.3.2 Breakpoint Determination

Although the aforementioned read-pair based approach can be used to locate most SV events (except multiple copy number duplications), they cannot be used to locate exactly where the breakpoints are in the genome. This is due to the fact that the distance between paired reads is dependent on the size of the fragment of their origin, which is not exact even under the best experimental conditions. To locate the breakpoints in these events, a split-read based approach may be used, which locates breakpoints by splitting some reads into subsequences that map to different genomic regions. Algorithms that use this approach include CREST [222], Pindel [223], SplazerS [224], and SRiC [225]. Pindel, for example, first searches for read pairs in which one read aligns to the reference genome but the other does not. Based on the assumption that the second read contains a breakpoint, it uses the aligned read as anchor to scan the surrounding regions for split mapping of the second read. Although it can locate breakpoints at single base resolution, this approach is computationally expensive because of the challenge associated with aligning read subsequences to different genomic regions with gaps in between.

9.3.3 *De novo* Assembly-Based SV Detection

Both the read-pair mapping and the split-read analyses are based on alignment to a reference genome. A different approach to SV detection is to use *de novo* assembly. This approach tries to assemble much of the genome directly from the reads, and then the assembled genome is compared to the reference genome searching for SVs. Cortex [226] is an SV detection algorithm that employs this *de novo* approach. While this approach has the advantage of being unbiased, it is computationally more intensive and demanding on computer hardware than the read-pair mapping approach. Because of the computational complexity involved in the process, it is less used compared to the other approaches.

9.3.4 CNV Detection

Detection of variation in segmental copy numbers is usually conducted with algorithms that detect abnormal changes in regional read frequency. These algorithms are based on the assumption that the number of reads obtained from a region is proportional to its copy number in the genome. If a genomic segment is repeated multiple times, a significantly higher number of reads will be observed from the segment compared to other nonrepeated regions. If a segment is deleted, on the other hand, there will be no read coverage for it. Examples of these algorithms include CNAseg [227], CNV-Seq [228], CNVnator [229], Event-Wise Testing (EWT) [230], JointSLM [231], mrFAST, and SegSeq [232]. As other factors, such as GC content, may also affect local

read density, a normalization step is often conducted in these methods to account for the compounding factors. In studies that involve comparison of samples from the same genetic background, for example a diseased versus a healthy tissue from the same patient, these compounding factors are often canceled out.

9.3.5 Integrated SV Analysis

The different software tools introduced earlier are usually tailored for detecting particular types (or aspects) of SVs. In order to make calls for the full range of SVs, there have been efforts to take an integrated approach toward comprehensive SV calling using the different but often complementary tools. SVMerge, being one of these efforts, integrates SV calling results from different callers [233]. It first feeds BAM files into a number of SV callers such as those introduced earlier to generate BED files, and then the SV calls in the BED files are merged. A comprehensive list of SVs is generated after computational validation with breakpoints refined by local *de novo* alignment. Other efforts that take a similarly integrated approach include GASVPro [234], SVSeq [235], and CNVer [236].

9.4 Annotation of Called Variants

To gain biological insights from identified SNVs, indels, or SVs, annotation of the variants is needed. For example, if an SNV is annotated to be nonsynonymous in a gene, it may impair protein function if the affected amino acid is located within the active site of the protein. Through examination of their annotations, called variants can be filtered and prioritized for more in-depth analysis. Because of the large number of variants usually called from an experiment, an automatic pipeline is usually preferred. To meet this demand, a number of variant annotation tools have been developed. ANNOVAR [237] is one such tool among the most widely used. It takes SNVs, indels, and CNVs as input, and as output, it reports their functional impacts and provides significance scores to help with filtering and prioritization. Its TABLE_ANNOVAR script can quickly turn a variant list into an Excel-compatible file containing many annotation fields that can help the researcher evaluate the function importance of the variants. ANNOVAR offers flexibility and extensibility; for example, it can identify variants located in conserved genomic regions, or find variants that overlap with those from the 1000 Genomes Project or dbSNP. Other variant annotation tools include SeattleSeq [238], SnpEff [239], and VEP (Variant Effect Predictor) [240]. For easy access, SeattleSeq and VEP provide a web interface. For local deployment, ANNOVAR, SnpEff, and VEP provide scripts for download.

9.5 Testing of Variant Association with Diseases or Traits

To identify polymorphic variants significantly associated with a disease or trait of interest, an association test needs to be carried out. For common polymorphic variants with frequency of occurrence >5%, the test is usually conducted at the level of single variants, which examine each variant individually for association with the disease or trait. Commonly used statistical methods include the chi-squared test, Fisher's exact test, Cochran-Armitage test for trend, or logistic regression for disease incidence and qualitative traits. For quantitative traits, such as blood pressure or body mass index, linear regression is often used. Because of the large number of individual tests involved in such an analysis, the significance level of each variant needs to be adjusted for multiple testing (such as false discovery rate [FDR]). Many of the aforementioned statistical methods are implemented in software tools such as PSEQ [241].

For the detection of rare polymorphic variants, that is, those with frequency of occurrence <5%, the single-variant level association test is often underpowered. To improve detection power, multiple variants, such as those located in a gene or a sliding window of predefined size, can be grouped together for association testing. In this approach, the different variants in the group are often tested individually first and then the individual test results combined to represent the group. To further improve detection power for rare variants, all variants across a genomic region can be aggregated and collapsed into a single unit for subsequent test. For such a test, multiple logistic or linear regression models can be used to combine the effects of these variants. For the collapsing method, statistical tests such as CAST (cohort allelic sums test) [242] or CMC (combined multivariate and collapsing) [243] can be used to determine if the aggregated burden of rare variants is significantly different between two conditions.

10

De novo Genome Assembly from Next-Generation Sequencing (NGS) Reads

Sanger sequencing was considered the golden standard for *de novo* genome assembly. However, it is prohibitively expensive and time-consuming to assemble a genome using this first-generation technology, as it took $3 billion and 13 years to generate the human genome draft assembly. The demand for low-cost and fast genome sequencing provides the very impetus for the development of next-generation sequencing (NGS) technologies. The dramatically reduced cost of NGS makes whole-genome shotgun sequencing much more affordable and accessible to individual labs. *De novo* genome assembly from the relatively short and enormous number of reads generated from most NGS platforms, however, poses serious challenges to assembling algorithms that were designed for Sanger sequences. The short length of NGS reads means that they carry less information and as a result lead to more uncertainties in the assembling process. To remedy this situation, higher coverage is required, which significantly increases the number of reads required and therefore the computational complexity. For example, using Sanger sequences with lengths up to 800 bp, assembling the human genome used approximately 8× coverage; for NGS reads of 35 to 100 bp, the same task needs 50× to 100× coverage [244].

Since Sanger sequence assemblers cannot deal effectively with these challenges, new *de novo* genome assemblers have been developed for NGS data. The development of Velvet [245] and ABySS [246] in 2008 and 2009 showed that *de novo* high-quality genome assembly can be achieved, even for large genomes, using massive numbers of ultrashort (as short as 30 bp) reads. The first *de novo* assembly of a human genome with the use of only short NGS reads was accomplished in 2010 with the development of SOAPdenovo [96]. With the recent rapid algorithmic developments in this direction, along with the gradual increase in read length, *de novo* genome assembly from NGS sequences has been becoming more and more robust.

10.1 Genomic Factors and Sequencing Strategies for *de novo* Assembly

10.1.1 Genomic Factors That Affect *de novo* Assembly

The size of a target genome to a large degree determines the difficulty of assembling it. All NGS *de novo* assemblers (see Section 10.2) can handle small genomes (<10 Mb), such as those of bacteria, without difficulty. For genomes of medium size (10 Mb–1 Gb), such as those of lower plants and insects, most of the assemblers should still work without much problem. For large genomes (>1 Gb), while some assemblers, such as the aforementioned SOAPdenovo, have been shown to have the capability to assemble the human or other mammalian genome, in general it is still not an easy task to put them together with only short reads (e.g., those from Illumina sequencers). In addition, assembling a large genome *de novo* is the most computationally demanding among all NGS applications.

The amount of repetitive sequences in a genome is another major factor that affects *de novo* genome assembly. Some genomes are inherently more difficult to assemble than others because they contain more repetitive sequences. Because they produce reads that are not unique due to their repetitive nature, repetitive regions create serious challenges in the genome assembly process. The challenges come from the inability to assemble reads from these regions into contiguous segments (contigs) or scaffolds, and the inability to determine the locations of these reads in relation to contigs or scaffolds assembled from reads from nonrepetitive regions. As a result, these regions become gaps in a draft assembly. Besides repetitive elements, genomic heterozygosity is another factor that may affect *de novo* assembly. Genomic heterozygosity is a measure of allelic differences in a genome, and allelic differences in a diploid or polyploid genome lead to uncertainty in assembling their reads together. In addition, other genome features, such as local GC content, may also affect *de novo* genome assembly.

10.1.2 Sequencing Strategies for *de novo* Assembly

Filling the gaps caused by repetitive regions is important for most *de novo* genome assembly projects, and how to fill them should be a major consideration when devising an appropriate sequencing strategy. The basic approach to connect contigs or scaffolds across the gaps is to use read pairs that span a distance longer than the gaps. These read pairs have to be generated from paired-end or mate-pair sequencing, and the known distances between the read pairs provide guidance to align the contigs or scaffolds over the gaps. Mate-pair sequencing differs from paired-end sequencing (see Chapter 4) in that the mate-pair approach is designed to "jump" sequence two ends of a larger DNA fragment. To conduct mate-pair sequencing, a DNA fragment

is first circularized to have the two ends joined. This circular DNA is then fragmented, and the segment that contains the junction of the two ends is selected and sequenced with paired-end sequencing. To span repetitive regions in different sizes, sequencing reads generated from mate-pair libraries of varying insert sizes (e.g., from 2 to 40 Kb), as well as regular paired-end reads, are often used [247,248].

The combined use of paired-end and mate-pair libraries of different insert sizes is a key strategy in assembling a genome from NGS reads. The paired-end sequencing generates reads at the shorter size range (e.g., 180 bp) for assembling of nonrepeat sequences as well as resolving short repeat sequences, whereas the mate-pair jump sequencing produces reads at the larger size range for resolving intermediate and long-range repeat regions and fill the corresponding gaps. Gaps of substantial sizes that are beyond the covering range of mate-pair libraries cannot be filled.

Besides the use of paired-end and mate-pair sequencing, read length is also a key parameter for *de novo* genome assembly. Although mammalian genomes have been assembled from reads shorter than 75 bp [96,248], longer reads are always better. To obtain long reads, some sequencing platforms, such as that from Pacific Biosciences, can be used. To balance read length with cost and error rate, NGS systems that usually do not read long sequences, such as the Illumina system, can also be used ingeniously to produce longer reads. For example, the current Illumina system can sequence 250 nucleotides from one end using the rapid run mode. If using this mode to conduct paired-end sequencing on libraries that contain DNA inserts of 450 bp, each generated read pair will overlap, and with software they can be merged to form a single long read covering the entire length of 450 bp. This strategy, combined with the use of mate-pair libraries of different sizes, or different sequencing technologies, can to a large degree overcome the limitation imposed by a short read length. Read length will become a lesser issue with the emergence of long-read NGS technologies.

Sequencing depth is another important factor to consider for a *de novo* assembly project. While it varies by project and is dependent on the other factors (including the amount of repeats and level of heterozygosity in the genome as well as read length and error rate), a coverage that is too low will undoubtedly result in a highly fragmented assembly. As a rough guide, in the combined use of paired-end and mate-pair libraries of various insert sizes, 45× to 50× coverage is needed for the short-insert-size paired-end and intermediate-size (3 to 10 Kb) mate-pair libraries, and 1× to 5× coverage for the long-insert (10 to 40 Kb) mate-pair libraries [249,250]. It should also be noted that while higher coverage may lead to improvement in the final assembly quality, additional increase in coverage also means increased data volume, computational complexity, and processing time. There are also studies showing that beyond a certain level of coverage, further increase in sequencing depth does not necessarily lead to an increase in assembly quality in terms of the size of assembled contigs [96].

10.2 Assembly of Contigs

10.2.1 Sequence Data Preprocessing, Error Correction, and Assessment of Genome Characteristics

The *de novo* assembly of a genome from NGS reads is a multistep process (Figure 10.1). As the first step, sequence data quality needs to be inspected. The data quality control (QC) steps described in Chapter 5 can be performed here to examine per-base error rate, quality score distribution, read size distribution, contamination of adaptor sequences, and so on. Low-quality reads need to be filtered out, and portions of reads that contain low-quality base calls (usually the 3′ end), ambiguities (reported as N's), or adaptor sequences should be trimmed off. As part of data preprocessing, paired-end reads with part of their sequences overlapped need to be merged to generate longer reads. The read merging can also correct errors if discrepancy at some base positions are observed, in which case the higher quality base call is used. The merging process can be handled by tools such as FLASH [251] or PANDAseq [252].

Sequencing error correction is an important step for *de novo* read assembly, more so than for most other NGS applications due to the fact that the

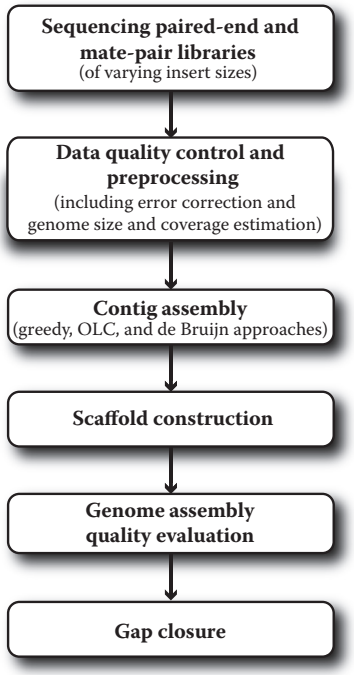

FIGURE 10.1
General workflow for *de novo* genome assembly.

assembly process is much more sensitive to these errors. The data QC measures mentioned earlier cannot totally remove sequencing errors, as high base-call quality scores alone cannot guarantee a read is free of sequencing errors. If left uncorrected, the errors will lead to prolonged computational time, erroneous contigs, and low-quality genome assembly. While it can be time consuming, an additional error correction step can improve final assembly quality. There are multiple options to carry out this step. For example, the Quake error corrector [253] can be used as a standalone tool, while some assemblers (see Section 10.2.2) have their own error correction modules, such as ALLPATHS-LG [254]. Most error correction algorithms are based on k-mer filtering [255]. K-mer refers to all the possible subsequences of length k in a read, and breaking reads to k-mers makes the complicated task of genome assembly more tractable. When all reads are converted to k-mers, most k-mers in the pool are represented multiple times. Having a k-mer that appears only once or twice is an indication of sequencing error (Figure 10.2). The general error correction approach is to find the smallest number of base changes to make all k-mers contained in a read "strong," that is, with the frequency of these k-mers from all reads above a threshold level. To determine the appropriate threshold level for error correction, the distribution of the frequency of k-mers can be plotted using data from a k-mer counting software such as Jellyfish [256]. From the distribution pattern, the size of the to-be-assembled genome, as well as coverage, can also be estimated. For example, tools such as Kmergenie [257], SGA [258], and VelvetOptimiser [259] provide reports on genome size and coverage from the k-mer distribution pattern. Some of these tools, like SGA, also report on other characteristics of the genome, such as repeat content and the level of heterozygosity.

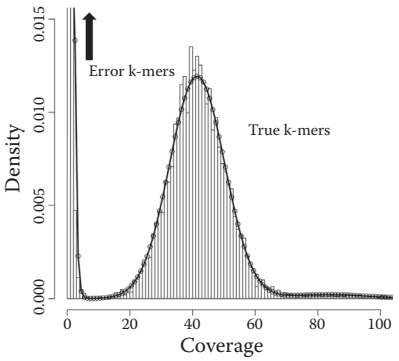

FIGURE 10.2
The coverage profile of true k-mers and those with sequencing errors. (From DR Kelley, MS Schatz, SL Salzberg, Quake: Quality-aware detection and correction of sequencing errors, *Genome Biology* 2010, 11:R116. Used under the terms of the Creative Commons Attribution License, http://creativecommons.org/licenses/by/2.0, © 2010 Kelley et al.)

10.2.2 Contig Assembly Algorithms

Fundamentally different from the reference-based alignment process, which is used by most of the other NGS applications in this book, *de novo* genome assembly attempts to construct a superstring (or superstrings) of DNA letters based on the overlapping of sequence reads. This assembly process was previously modeled by Lander and Waterman with the use of ideal (error- and repeat-free) sequence data [260]. In this model, if two reads overlap and the overlap is above a cutoff level, the two reads are merged into a contig and this process reiterates until the contig cannot be further extended. Although this guiding model is straightforward, finding all possible overlaps between millions of short reads and assembling them into contigs are computationally intensive and challenging. Added to this challenge are other complicating factors such as sequencing errors, heterozygosity, and repetitive sequences. To deal with these challenges, a number of assemblers that employ different methodologies have been developed.

The currently available *de novo* genome assemblers can be classified into three major categories: those using (1) the Greedy approach, (2) the overlap–layout–census (OLC) approach, and (3) the de Bruijn graph. Although all of them are based on graphs, the Greedy approach is the one that is based on the maximization of local sequence similarity. It was used by Sanger sequence assemblers, such as phrap and the TIGR assembler, and early NGS reads assemblers, such as SSAKE [261], SHARCGS [262], and VCAKE [263]. Since it is a local approach, the Greedy approach does not consider the global relationship between reads. Therefore, more recent NGS-based assemblers no longer use this approach, as it cannot take advantage of the global relationship offered by paired-end and mate-pair reads.

The OLC and the de Bruijn graph approaches are global by design, and both assemble reads into contigs using reads overlapping information based on the Lander-Waterman model. They approach the task, however, in different ways (Figure 10.3). The OLC approach, as the name suggests, involves three steps: (1) detecting potential overlaps between all reads; (2) laying out all reads with their overlaps in a graph; and (3) constructing a consensus sequence superstring. The first step is computationally intensive and the run time increases quadratically with the increase in the total number of reads. The graph created in the second step consists of vertices (or nodes) representing reads, and edges between them representing their overlaps. The construction of a consensus sequence superstring equals to finding a path in the graph that visits each node exactly once, which is known as a Hamiltonian path in graph theory. Currently available OLC-based short read assemblers include CABOG [264], Edena [265], Fermi [266], Forge [267], and Newbler [66]. The OLC approach is widely used to assemble longer reads generated from 454 and Sanger sequencers, but relatively less used to assemble short reads (such as those from Illumina sequencers) due to the demand for significantly higher depth and consequently quadratic increase in computational

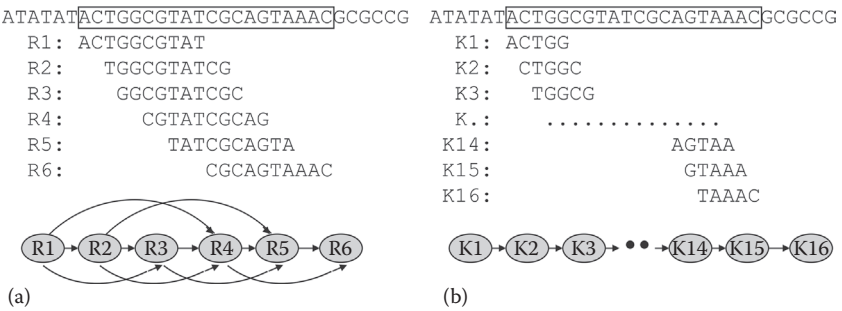

FIGURE 10.3
Comparison of the (a) OLC and the (b) de Bruijn graph approaches for global *de novo* genome assembly. In the OLC example, six sequence reads (R1–R6) are shown for the illustrated genomic region, with each read being 10 bp in length and the overlap between them set at ≥5 bp. The reads are laid out in order based on how they overlap. The OLC graph is shown at the bottom, with many nodes having more than one incoming or outgoing connection. In the de Bruijn graph example, the reads are cut into a series of k-mers (k = 5). In total there are 16 such k-mers, many of which occur in more than one read. The k-mers are arranged sequentially based on how they overlap, and the de Bruijn graph built from this approach is shown at the bottom. Different from those in the OLC graph, the majority of the nodes in this graph have only one incoming and one outgoing connection. (From Z Li, Y Chen, D Mu, J Yuan, Y Shi, H Zhang, J Gan, et al., Comparison of the two major classes of assembly algorithms: Overlap–layout–consensus and de-Bruijn-graph, *Briefings in Functional Genomics* 2012, 11(1):25–37. With permission.)

complexity. To reduce the high computing demand imposed by this approach, a simplified version of the OLC graph called the String graph has been employed to merge and reduce redundant vertices and edges, along with identification and removal of false vertices and edges [268]. The implementation of a string indexing data structure called FM-index has improved the performance of assemblers such as SGA and ReadJoiner [269].

Compared to the OLC approach, the de Bruijn graph-based approach takes an alternative, computationally more tractable route. This approach does not involve a step to find all possible overlaps between reads. Instead, the reads are first cut into k-mers. For instance, the sequence read ATTACGTCGA can be cut into a series of k-mers, for example, ATT, TTA, TAC, ACG, CGT, GTC, TCG, and CGA, when k = 3. These k-mers are then used as vertices in the de Bruijn graph. An edge that connects two nodes represents a convergence of the two nodes, for example, the edge that connects ATT and TTA is ATTA. Using the de Bruijn graph, the assembly process is equivalent to finding a shortest path that visits each node at least once, which is known as the Chinese postman problem in graph theory. An Eulerian path, if it exists, represents the solution to this problem. Computationally, finding an Eulerian path is much easier than finding a Hamiltonian path for the OLC approach. The major drawback of this approach, however, is that it is highly sensitive to

sequencing errors. Therefore, to use assemblers in this category, error correction is mandatory. Assemblers that use this approach include ABySS [246], ALLPATHS-LG [254,270,271], Euler-SR [255], IDBA-UD [272], SOAPdenovo [96,273], SparseAssembler [274], and Velvet [245]. Some assemblers, such as MaSuRCA [275], combine the de Bruijn graph and OLC approaches in an attempt to increase efficiency.

10.3 Scaffolding

After assembly of contigs, the next step is to organize the contigs into a "scaffold" structure to improve continuity rather than leave them disjointed. This scaffolding process orders and orients the contigs, and estimates the lengths of the gaps between them (Figure 10.4). To establish positional relationship between contigs, scaffolding algorithms use mate-pair reads that span different contigs.

For input, scaffolding algorithms take preassembled contigs, mate-pair reads, and sometimes long reads generated from other sequencing technologies (such as 454 or PacBio). The first and also an important step in the scaffolding process is to map the input read pairs or long reads to the contigs. To improve mapping results, sequencing errors in the reads should be corrected prior to mapping. To assemble the contigs into scaffolds using the guiding information in the mate-pair or long reads, scaffolders usually take a graph-based approach similar to the contig assembly process, but here with contigs as nodes and connecting read pairs (or long reads) as edges. The quality

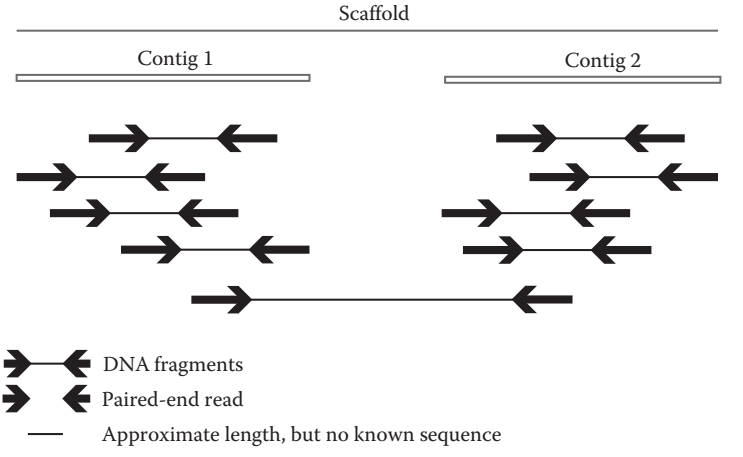

FIGURE 10.4
Assembling contigs into a scaffold.

of the assembled scaffolds is dependent on the quality of input contigs, the complexity of the genome, and the quality of mate-pair or long-read libraries. The sizes of the scaffolds are limited by the insert size of mate-pair libraries or the length of long reads, as the scaffolds cannot span repetitive regions larger than the insert size or read length.

Currently available standalone scaffolders include Bambus2 [276,277], Opera [278], SOPRA [279], and SSPACE [280]. Many contig assemblers, including ABySS, SGA, and SOAPdenovo2, also have built-in scaffolding modules. The performance of different scaffolders varies with data sets and analysis parameters. Therefore, before deciding on an appropriate scaffolder for a project, it is helpful to first try different scaffolders using different parameters and then evaluate the results (see Section 10.4). As of this writing, SGA, SOAPdenovo2, SOPRA, and SSPACE seem to perform well based on benchmark tests [281].

10.4 Assembly Quality Evaluation

Contiguity, completeness, and accuracy are key indices of the quality of an assembly. Contiguity is reflected by the total number of assembled contigs or scaffolds and their size distribution, that is, whether the assembly is composed of a small number of large fragments or a large number of small fragments. It can be measured by statistics such as mean or median length, but the most commonly used statistic is N50, which is the weighted mean of assembled contigs (or scaffolds). To calculate the N50, all contigs (or scaffolds) are first ranked based on length from the largest to the smallest. Their lengths are then summed up from the largest contig (or scaffold) downward. N50 refers to the size of the contig (or scaffold) at which the summed length becomes greater or equal to 50% of the total assembly size.

The total assembly size, however, does not measure the completeness of the assembly. To determine completeness, the original DNA reads are aligned to the assembly and the percentage of reads aligned is calculated. Other sequence data from the same species, such as RNA-Seq data, may also be used for the alignment and rough estimation of completeness. On the measurement of accuracy, the assembly can be compared to a high-quality reference genome of the species, if such a reference is available. This comparison can be carried out on two aspects of the assembly: base accuracy and alignment accuracy. Base accuracy determines if the right base is called in the assembly at a given position, while alignment accuracy examines the probability of placing a sequence at the right position and orientation. In many cases, however, a reference map is not available and instead is the very goal of the assembling process. For these cases, a measurement on internal consistency, through aligning original reads to the assembly and checking

for evenness and congruence in coverage across the assembly, provides an indicator of the assembly quality. Comparison of the assembly with independently acquired sequences from the same species, such as gene or cDNA sequences, can also be used to estimate assembly accuracy. With regard to software implementation on evaluating assembly quality, only a limited number of tools are currently available, such as QUAST [282], to help perform the aforementioned measurements and compare different contig and scaffold assembly algorithms and settings.

10.5 Gap Closure

The final stage of finishing a *de novo* genome assembly is to close the gaps between contigs. A standard approach to achieving this is to employ PCR, first to amplify the gapped regions using primers specific to the ends of the two contigs bordering the gaps, followed by sequencing of the amplicons. If the number of gaps is high, however, this approach can be laborious and expensive. Alternatively, gap filling software, such as IMAGE [283], GapFiller [284], or gap filling modules in some assemblers (such as SOAPdenovo) can be used to close the gaps using paired reads generated from the gapped regions. For example, IMAGE uses a targeted reassembly process in the gap region to create new contigs to gradually fill the entire gap (Figure 10.5). It first collects read pairs that align to contig ends and uses them to create new contigs that extend into the gap region. After incorporating the new contigs into the scaffold, this process is reiterated until the entire gap is filled.

10.6 Limitations and Future Development

The short read length of most current NGS systems poses a limit on *de novo* genome assembly. This, combined with other factors including sequencing errors, repetitive elements, and uneven regional coverage, leads to ambiguities, false positive and branched paths in the assembly graph, and early terminations in contig extension, limiting the completeness of assembled sequences. As a result, the assembled sequences are usually fragmented and exist in the suboptimal form of large numbers of contigs. Among the contigs, there are also certain (sometimes high) levels of falsely assembled contigs, due to chimeric joining. In addition, the gapped regions between the assembled contigs may not be completely filled. To overcome some of these limitations and increase assembly quality, the use of a reference genome, even from a remotely related species, can be very helpful. This reference-assisted

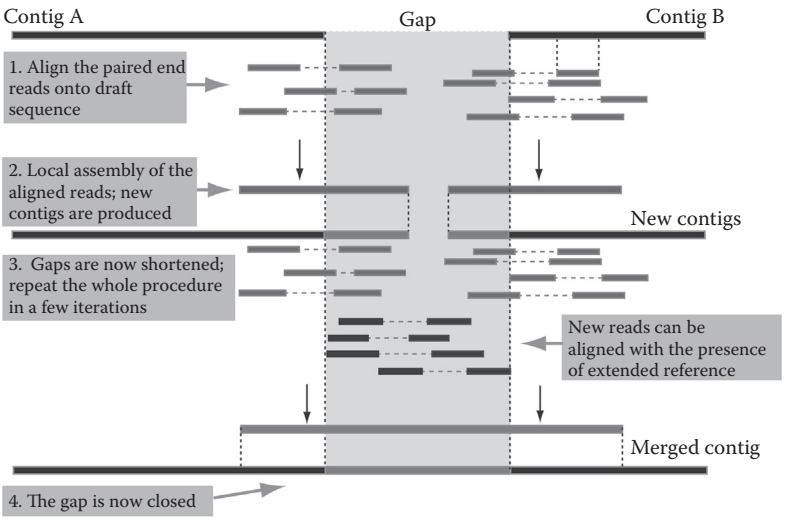

FIGURE 10.5
Gap closing with the IMAGE process. (From IJ Tsai, TD Otto, M Berriman, Improving draft assemblies by iterative mapping and assembly of short reads to eliminate gaps, *Genome Biology* 2010, 11:R41. Used under the terms of the Creative Commons Attribution License, http://creativecommons.org/licenses/by/2.0, © 2010 Tsai et al.)

assembly approach works especially well when scaffolding information from paired reads is not available or exhausted. With the quickly increasing number of sequenced genomes, improving assembly quality with this reference-assisted approach becomes more feasible. Some tools have recently been developed to provide this functionality, either as dedicated packages such as AlignGraph [285] and RACA [286], or components of existing assemblers including ALLPATHS-LG, IDBA-Hybrid, and Velvet.

With the development of third-generation sequencing technologies that generate increasingly long reads, the landscape of *de novo* genome assembly will be bound to change. In the meantime, to further overcome the limitations caused by short reads, the community has devised innovative workaround approaches. For example, a hierarchical sequencing approach has been used to increase the assembly quality of large complex genomes. In this approach, a genome is first divided into a small number of large overlapping fragments, each of which is made into a sequencing library. After independent sequencing of the libraries, the reads from each library are assembled into contigs. Subsequently, all contigs assembled from the different libraries are merged to supercontigs, which are then connected with scaffolders. This hierarchical approach leads to a significant decrease in sequence complexity within each library and an increase in final assembly quality. Ingenious workaround approaches like this overcome the challenges resulting from the current short-read-based shotgun approach.

11

Mapping Protein–DNA Interactions with ChIP-Seq

11.1 Principle of ChIP-Seq

Without the involvement of DNA-interacting proteins, the information coded in DNA could not be accessed, transcribed, and maintained. Besides a large number of transcription factors and coactivators, key DNA-interacting proteins include histones, DNA and RNA polymerases, and enzymes for DNA repair and modification (e.g., methylation). Through their DNA-interacting domains, such as helix-turn-helix, zinc finger, and leucine zipper domains, these proteins interact with their DNA targets by hydrogen bonding, hydrophobic interactions, or base stacking. Because the intimate relationship between DNA and these proteins plays an important role in the functioning of the genome, studying how proteins and DNA interact and where DNA-interacting proteins bind across the genome provides key insights into the many roles these proteins play in various aspects of genomic function, including information exposure, transcription, and maintenance.

ChIP-Seq is a next-generation sequencing (NGS)-based technology to locate binding sites of a DNA-interacting protein in the genome. An exemplary scenario for using ChIP-Seq is to study transcription factor binding profiles in the genome under different conditions, such as development stages or pathological conditions. To achieve this, the protein of interest is first cross-linked covalently to DNA in cells with a chemical agent, usually formaldehyde (Figure 11.1). Then the cells are disrupted, and subsequently sonicated or enzymatically digested to shear chromatin into fragments that contain 100 to 300 bp DNA, followed by enrichment of the target protein with its bound DNA by immunoprecipitation using an antibody specific for the protein. Subsequently, the enriched protein-DNA complex is dissociated by reversing the cross-links previously formed between them, and the released DNA fragments are subjected to NGS. One key experimental factor in the ChIP-Seq process is the quality of the antibody used in the enrichment step, as the use of a poor-quality antibody can lead to high experimental noise due to nonspecific precipitation of DNA fragments.

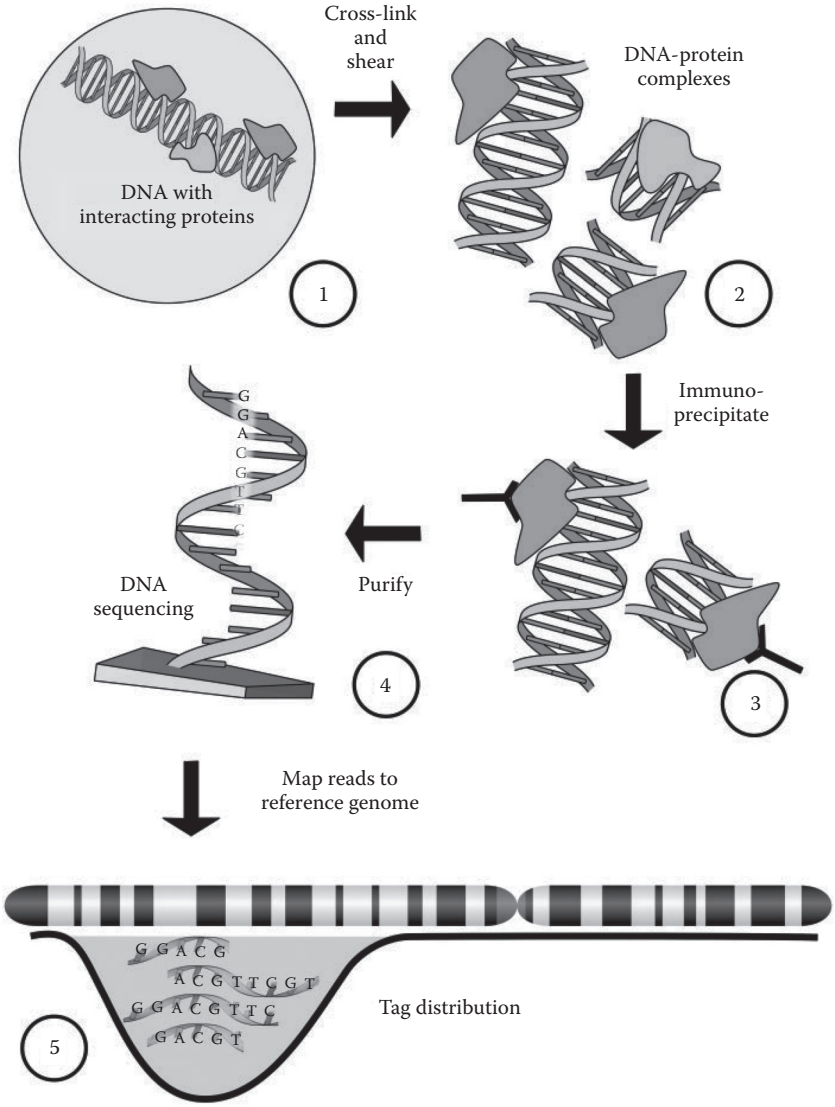

FIGURE 11.1
The basic steps of ChIP-Seq. (From AM Szalkowski, CD Schmid, Rapid innovation in ChIP-seq peak-calling algorithms is outdistancing benchmarking efforts, *Briefings in Bioinformatics* 2011, 12(6):626–633. With permission.)

Based on the size of the region(s) that they bind, DNA-interacting proteins can be divided into three groups:

1. Punctate binding—These proteins, usually transcription factors, bind to a genomic region that is a few hundred base pairs or less in size.
2. Broad binding—Chemically modified histones, or other proteins associated with chromatin domains, bind to a much larger area of the genome up to several hundred thousand base pairs.
3. Mixed or intermediate binding—These include proteins such as RNA polymerase II, which bind to regions of the genome that are a few thousand base pairs in size.

11.2 Experimental Design

11.2.1 Experimental Control

Appropriate control for a ChIP-Seq experiment is the key to account for artifacts or biases that might be introduced into the experimental process. These artifacts and biases may include potential antibody cross-reactivity with nonspecific protein factors, higher signal from open chromatin regions (since they are easier to fragment than closed regions [287]), and uneven sequencing of captured genomic regions due to variations in base composition. Two major types of controls are usually set up for ChIP-Seq signal adjustment. One is input control, that is, chromatins extracted from cells or tissues, which are subjected to the same cross-linking and fragmentation procedure but without the immunoprecipitation process. The other is "mock" control, which is processed by the same procedure including immunoprecipitation; the immunoprecipitation, however, is carried out using an irrelevant antibody (e.g., IgG). While it may seem to serve as the better control between the two, the mock control often produces much less DNA for sequencing than real experimental ChIP samples. Although sequencing can be carried out on amplified DNA in this circumstance, the amplification process adds additional artifacts and bias to the sequencing data, which justifies the use of input DNA as the experimental control in many cases.

11.2.2 Sequencing Depth

How many reads to obtain for a ChIP-Seq experiment depends on the size of the genome and how many binding sites the protein of interest has in the genome. A good indication of having reached sufficient sequence depth is when the number of protein binding sites reaches plateau with the increasing

number of reads. As a practical guide, for analyzing a transcription factor that has thousands of binding sites in the mammalian genome, 20 million reads may be sufficient. Fewer reads may suffice for a smaller genome, while more reads are required for proteins that bind to the genome at a higher frequency or with larger "footprint." To locate binding regions of these proteins, including histone marks, 60 million reads might be needed for a genome at the scale of the human genome [287]. Higher sequencing depth is required for control samples in order to obtain background signals from most regions of the genome.

11.2.3 Replication

To examine the reproducibility of a ChIP-Seq experiment and to reduce the false discovery rate (FDR), replicate samples should be used. If a protein of interest binds to regions of the genome with high affinity, the bound regions should be identified in replicate samples. Regions that are not identified in replicates are most possibly due to experimental noise. The Pearson correlation coefficient (PCC) between biological replicates serves as a measurement of experimental reproducibility, and the irreproducible discovery rate (IDR) is another such metric. The calculation and usage of the PCC and IDR will be detailed later in this chapter.

11.3 Read Mapping, Peak Calling, and Peak Visualization

11.3.1 Data Quality Control and Read Mapping

The first step in ChIP-Seq data analysis (Figure 11.2) is to evaluate reads quality. The quality control (QC) metrics detailed in Chapter 5 need to be examined. If necessary, low-quality reads should be filtered out and low quality bases trimmed off. Other aspects of determining ChIP-Seq data quality include assessing library complexity and experimental reproducibility between replicates. Assessment of library complexity is important, as low-complexity libraries, caused by limited starting material, over-crosslinking, low antibody quality, or polymerase chain reaction (PCR) overamplification, can lead to skewed reads distribution. Library complexity can be examined with tools such as Preseq [288], or using the PCR bottleneck coefficient (PBC), which is defined as the ratio of N1/Nd, with N1 being the number of non-redundant, uniquely mapped reads, and Nd the number of uniquely mapped reads. PBC is calculated by a component of ENCODE Software Tools (http://www.encodeproject.org/software/) called phantompeakqualtools, which, besides PBC, also calculates other quality metrics, such as normalized strand cross-correlation (NSC) and relative strand cross-correlation coefficients (RSC),

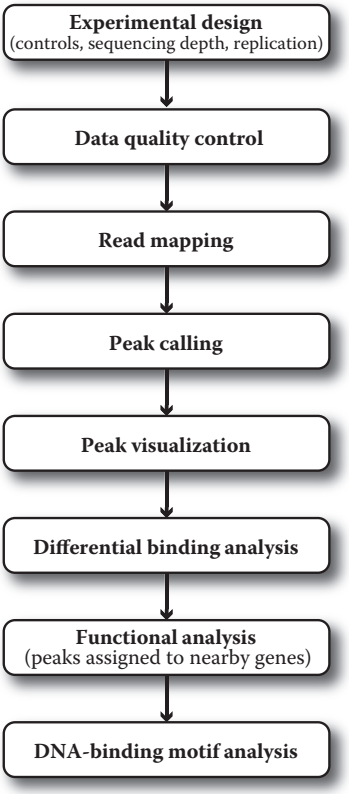

FIGURE 11.2
Basic ChIP-Seq data analysis workflow.

as measures of sequence enrichment (NSC and RSC will be discussed more in Section 11.3.2). The assessment of experimental reproducibility is usually performed by analyzing IDR, which can be calculated using another component of ENCODE Software Tools called Irreproducible Discovery Rate (IDR) (http://www.encodeproject.org/software/idr/).

The assessment of library complexity and experimental reproducibility by the ENCODE Software Tools, or the use of other ChIP-Seq QC tools such as CHANCE [289], requires mapping the filtered/trimmed reads to a reference genome. For this mapping, the mappers introduced in Chapter 5, including Bowtie, BWA, or SOAP, can be used. One mapping parameter that directly affects subsequent binding site detection sensitivity and specificity is whether to use multireads, which are reads that map to multiple genomic regions. Multireads may represent background noise and, if this is the case, should be excluded from further analysis, but they may also represent true signals located in repeats or duplicated regions. Including them increases

sensitivity but at the expense of higher FDR, while excluding them improves specificity but at the risk of losing true signals. The choice for their inclusion or not, therefore, is dependent on whether sensitivity or specificity is a priority. Independent of whether multireads are used, the percentage of uniquely mapped reads reported from the mappers is indicative of data quality. If this value is below 50%, it may indicate a potential problem with the experimental procedure and caution should be used in the interpretation of the data. ChIP-Seq involving proteins that bind to repetitive regions of the genome may also generate a low percentage of uniquely mapped reads.

For ChIP-Seq reads mapping, it is also worth mentioning that ChIP is an enrichment, not purification, of protein-bound DNA sequences. As a result, more reads are usually generated from background noise than from bound regions. The background noise can be determined empirically with the use of control samples. The distribution of observed background noise is not random as many would expect (Figure 11.3). Instead, it is affected by the density of mappable reads in different genomic regions and the local chromatin structure (e.g., as previously mentioned, an open chromatin structure generates more background reads). True binding signals in ChIP-Seq samples are

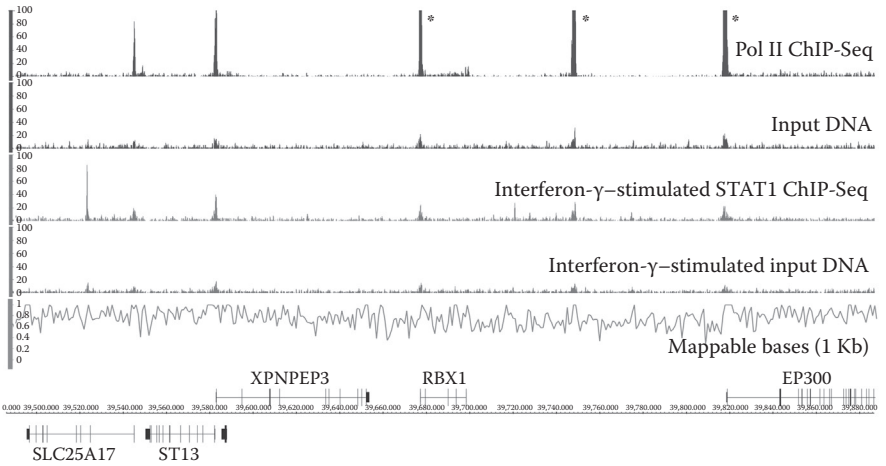

FIGURE 11.3
Background noise and signal profiles in a ChIP-Seq experiment. Shown here is the density of mapped reads in one region of the human chromosome 22 for RNA polymerase II and the transcription factor STAT1 (tracks 1 and 3, count from the top), respectively. Genes coded by the two DNA strands in this region are displayed at the bottom. Tracks 2 and 4 show the distribution of mapped reads for the respective input DNA controls for the two proteins. It should be noted that some of the peaks in the protein tracks are also present in their input controls. Track 5 displays the fraction of uniquely mappable bases. (From J Rozowsky, G Euskirchen, RK Auerbach, ZD Zhang, T Gibson, R Bjornson, N Carriero et al., PeakSeq enables systematic scoring of ChIP-seq experiments relative to controls, *Nature Biotechnology* 2009, 27:66–75. With permission.)

usually superimposed on the background noise. In the absence of control samples, although the background noise could be estimated from modeling of the ChIP-Seq data itself, the estimation cannot fully capture the inherent complexity of the background noise and therefore experimental controls should always be run. To further complicate the situation, the degree of protein-binding sequence enrichment may also vary from location to location, with some having strong signals and others more modest signals. The degree of enrichment at each location is not necessarily a reflection of their biological importance, as those with more modest enrichment may be equally important as those at the top of the enrichment list.

After mapping, reproducibility between replicate samples and overall similarity between different samples can be examined with the PCC. The PCC can be calculated with tools such as GMD [290] using sample read counts at each genomic location. In this calculation, regions that have no signals in both samples should be excluded, as they lead to overestimation of the PCC. For replicate samples in experiments of high reproducibility, the level of PCC is expected to be >0.9. For unrelated samples, it is typically in the range of 0.3 to 0.4. For a successful run, the PCCs between replicate samples should be much higher than those between ChIP and their control samples. Besides PCC and the other aforementioned QC measures such as PBC, additional QC analyses can also be performed. For example, visualization of the distribution of mapped reads in the genome, using the visualization tools introduced in Chapter 5, can offer further clues on data quality. This is especially true when some specific binding regions have already been known for the protein of interest. In comparison to those from control samples, sequence reads from ChIP samples should show strong clustering in these regions.

11.3.2 Peak Calling

Peak calling, the process of finding regions of the genome to which the protein of interest binds, is a key step in ChIP-Seq data analysis. It is basically achieved through locating regions where reads are mapped at levels significantly above the background. The simplest way of peak calling is to count the total number of reads mapped along the genome and call each location with the number of mapped reads over a threshold as a peak. Due to the inherent complexity in ChIP-Seq signal generation, including uneven background noise and other confounding experimental factors, this approach is overly simplistic. Among the experimental factors, the way the immunoprecipitated DNA fragments are sequenced on most platforms has a direct influence on how peaks are called. Since the reads are usually short, only one end or both ends of a fragment, instead of the entire fragment, are sequenced. To locate a target protein's binding regions, which are represented by the immunoprecipitated DNA fragments and not just the generated reads, peak calling algorithms need to either extend or shift the reads to cover the actual binding areas. For example, PeakSeq extends each mapped read in the 3′

direction to reach the average length of DNA fragments [292]. Alternatively, Kharchenko et al. [300] used a strategy to shift reads mapped to the two opposite strands relative to each other (Figure 11.4).

The reads shift approach and the strand cross-correlation profile shown in Figure 11.4 can also be used to evaluate ChIP-Seq data quality. When using short reads (usually less than 100 bases) to analyze large target genomes, which usually results in a significant number of reads being mapped to multiple genomic locations, a "phantom" peak also exists at a shift that equals to the read length [301] (Figure 11.5). If a run is successful, the fragment-length ChIP peak should be significantly higher than the read-length "phantom" peak, as well as the background signal. The aforementioned ENCODE software phantompeakqualtools provides two indices for the examination of strand cross-correlation: (1) NSC, the ratio of the cross-correlation coefficient at the fragment-length peak over that of the background; and (2) RSC, the ratio of background-adjusted cross-correlation coefficient at the fragment-length peak over that at the phantom peak.

Shifting reads mapped to the positive and negative strands toward the center, or extending reads to reach the average fragment length, in order to count the number of aggregated reads at each base-pair position is the first substep to peak calling. As illustrated in Figure 11.6, peak calling involves multiple substeps. First, a signal profile is created through smoothing of aggregated read count across each chromosome. Subsequently, background noise needs to be defined and the signals along the genome need be adjusted for the background. One simple approach is to subtract read counts in the control sample, if available, from the signal across the genome, or use the signal-to-noise ratio. In the absence of a control sample, the background noise can be modeled using Poisson or negative binomial distributions. Some peak callers also use modeling to simulate background in their initial pass even when control data is available. For example, PeakSeq uses background modeling in its first pass to identify potential binding regions. In the second pass, to more accurately adjust for the background using control data, the fraction of reads located in the initially identified potential binding regions are excluded and the reads in the remainder of the genome in the ChIP-Seq sample is normalized to the control data by linear regression [292]. Some other peak calling packages, such as MACS [293] and CisGenome [294], use similar approaches for background adjustment using control data.

To call peaks from the background-adjusted ChIP-Seq signal, often-used approaches include using absolute signal strength, signal enrichment in relation to background noise (shown in Figure 11.6), or a combination of both. To facilitate determination of the signal enrichment, the statistical significance is often computed using Poisson or negative binomial distributions. Empirical estimation of the FDR can be carried out by first calling peaks using control data (i.e., false positives), and subsequently calculating the ratio of peaks called from the control to those called from the ChIP sample.

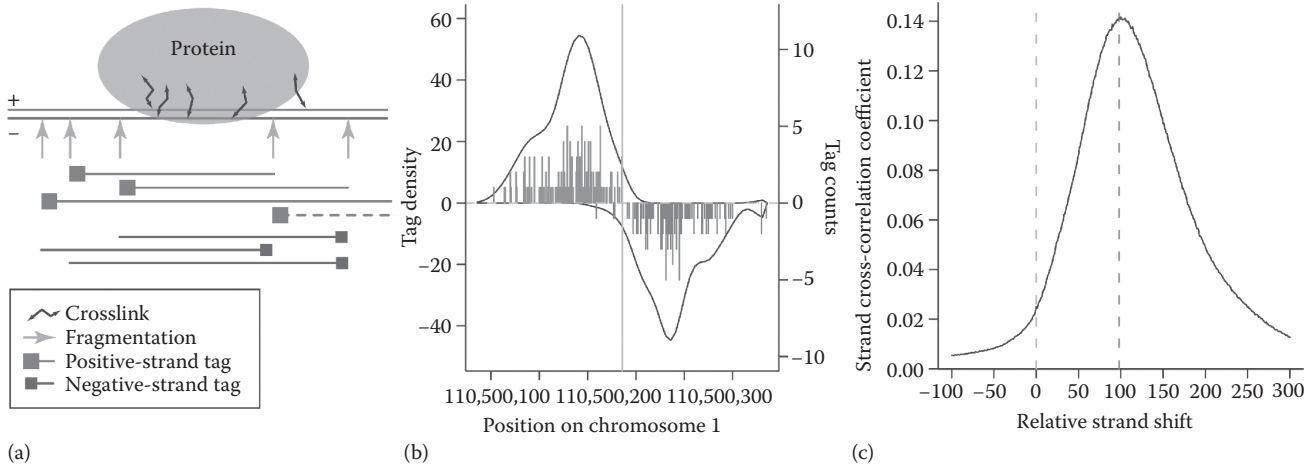

FIGURE 11.4
The distribution of ChIP-Seq reads around the actual binding region and their positional shift on the two DNA strands. (a) How ChIP-Seq reads are produced from cross-linked and fragmented DNA. The cross-linking between the protein and the bound DNA can occur at different sites, as does the fragmentation of the DNA. Each fragment is read at its 5′ end (indicated by the squares). These reads, serving as sequence tags of each fragment, are clustered around the actual binding region from the two sides depending on which strand they come from. The dashed line depicts a fragment from a long cross-link. (b) The distribution of sequence tag signal around the binding region. Vertical lines represent counts of sequence tags whose 5′ end maps to each nucleotide position on the positive and negative strands (displayed as positive and negative values, respectively). The solid curves represent smoothed tag density along each strand. Since the two curves approach the binding site from the two sides, there is a gap between their peaks. (c) Strand cross-correlation associated with shifting the strands across the gap. Before shifting the strands, the Pearson correlation coefficient is calculated between the tag density of the two strands. When sequence tags mapped to the two strands are shifted relative to each other (shown on the x-axis), the Pearson correlation coefficient gradually changes (y-axis). The dashed line at x = 0 corresponds to the strand cross-correlation before the shift, while the one at the peak corresponds to the highest cross-correlation coefficient at the strand shift that equals to the average length of the DNA fragments. (From PV Kharchenko, MY Tolstorukov, PJ Park, Design and analysis of ChIP-seq experiments for DNA-binding proteins, *Nature Biotechnology* 2008, 26:1351–1359. With permission.)

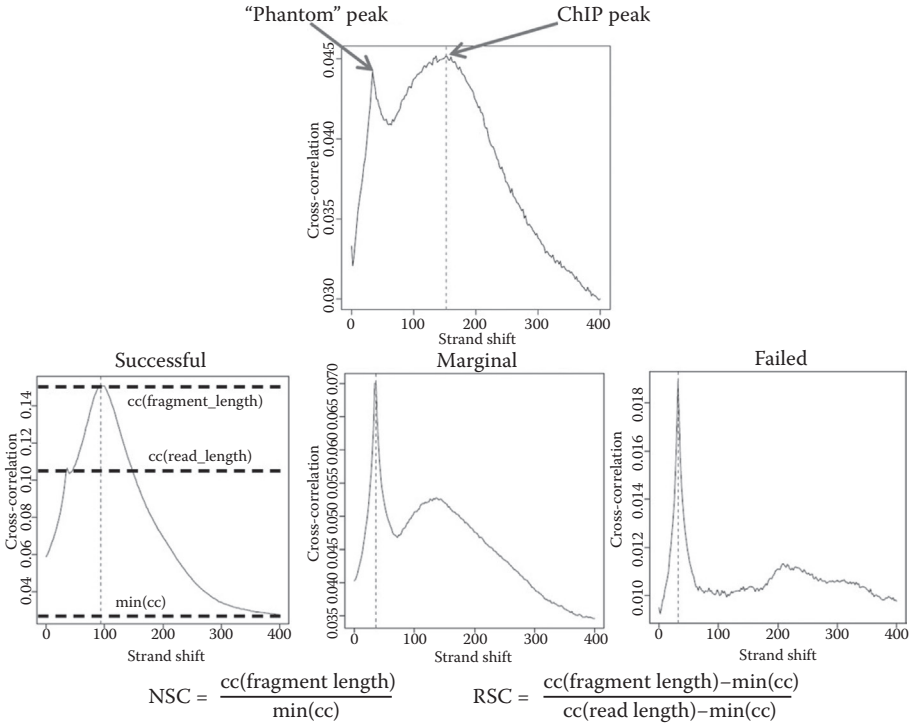

FIGURE 11.5
The "phantom" peak and its use in determining ChIP-Seq data quality. The phantom peak corresponds to the cross-correlation at the strand shift that equals to the read length, while the ChIP peak corresponds to the cross-correlation at the shift of the average DNA fragment length. A successful run is characterized by the existence of a predominant ChIP peak and a much weaker phantom peak. In marginally passed or failed runs, the former diminishes while the latter relatively becomes much stronger. (Adapted from SG Landt, GK Marinov, A Kundaje, P Kheradpour, F Pauli, S Batzoglou, BE Bernstein et al., ChIP-seq guidelines and practices of the ENCODE and modENCODE consortia, *Genome Research* 2012, 22(9):1813–1831. With permission. ©2012 Cold Spring Harbor Laboratory Press.)

After peak calling, artifactual peaks need to be filtered out, including those that contain only one or a few reads that are most possibly due to PCR artifacts, or those that involve significantly imbalanced numbers of reads on the two strands (see Figure 11.6).

For implementation of this peak calling process, different peak callers use different methods, which can lead to differences in final outcomes. Table 11.1 shows some of the currently available peak callers. PeakSeq, MACS/MACS2, HOMER (findPeaks module) [302], and SPP [300] are among some of the popular ones. As previously mentioned, the peak calling employed in PeakSeq is a two-pass process. In the second pass, peaks are called by scoring reads-enriched target regions based on calculation of the fold enrichment in the

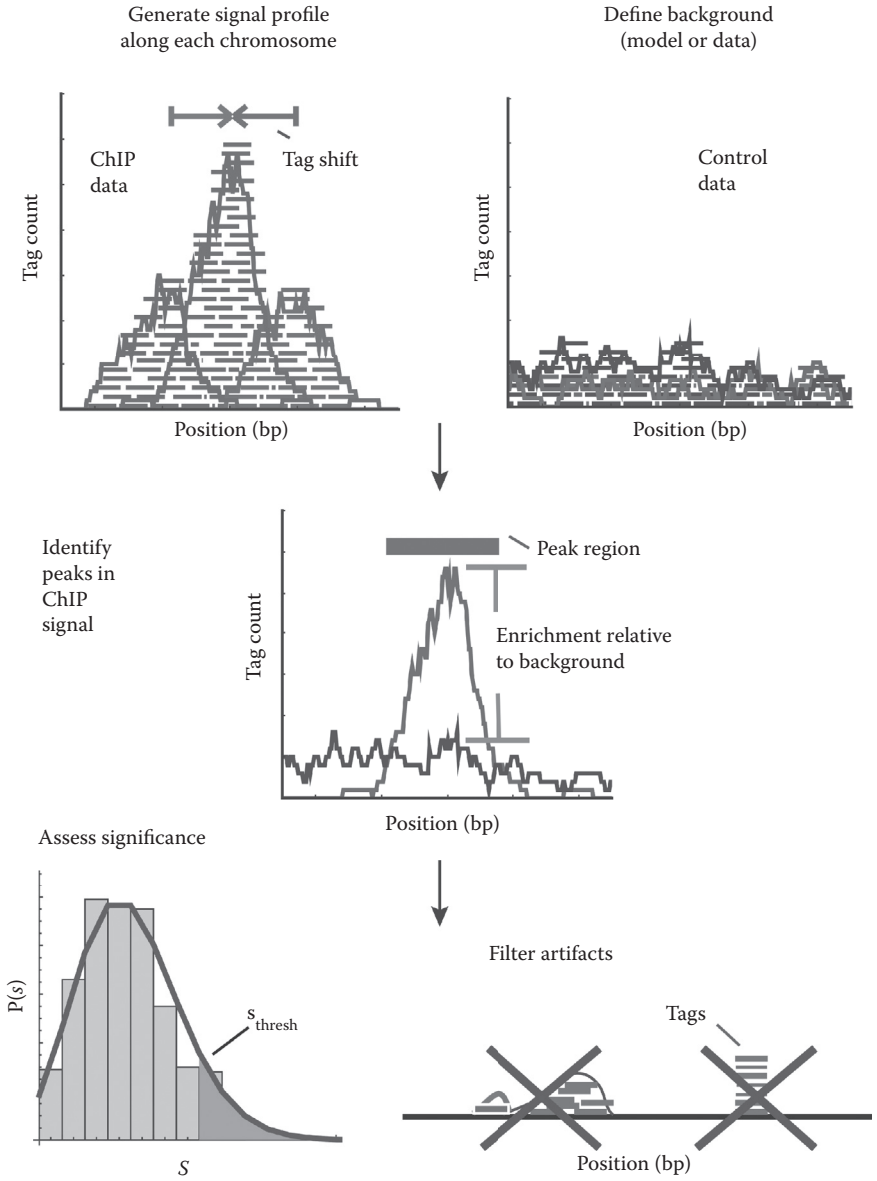

FIGURE 11.6
Basic substeps of calling peaks from ChIP-Seq data. The P(*s*) at bottom left signifies the probability of observing a location covered by *S* mapped reads, and the s_{thresh} marks the threshold for calling a peak significant. Bottom right shows two types of possible artifactual peaks: single strand peaks and those based on mostly duplicate reads. (From S Pepke, B Wold, A Mortazavi, Computation for ChIP-seq and RNA-seq studies, *Nature Methods* 6, 2009:S22–S32. With permission.)

TABLE 11.1

ChIP-Seq Peak Calling Algorithms

Name	Description	Reference
CCAT	Designed to identify weak ChIP signals	303
CisGenome	Features multifaceted interactive analysis and customized batch-mode computation	294
E-RANGE	A Python package for both ChIP-Seq and RNA-Seq data analysis	304
F-Seq	Generates continuous genomic sequence density data for easier visualization and interpretation	305
GLITR	Uses classification to identify regions that have peak height and fold-change not resembling those in control	306
HOMER (findPeaks module)	Identifies peaks based on the principle that more sequencing reads are found in these regions than expected by chance	302
MACS/MACS2	Empirically models ChIP-Seq read length to improve peak prediction, uses a dynamic Poisson distribution	293
PeakSeq	Based on a two-pass strategy to compensate for open chromatin signal	292
PeakRanger	Uses a staged algorithm to discover enriched regions and the summits within them	307
QuEST	A statistical framework based on kernel density estimation	308
RSEG	Especially developed for locating genomic regions associated with histone marks	309
SICER	Uses a clustering approach to identify enriched domains from histone modification ChIP-Seq data	310
SiSSRs	Uses the direction and density of reads and the average DNA fragment length to identify binding sites	311
SPP	Includes binding profile normalization, peak detection, and estimation of read depth to achieve peak saturation	300
USeq	Empirical algorithms to reduce false positives and estimate confidence in ChIP-Seq peaks	312
ZINBA	Models and accounts for factors covarying with background or true signals	313

ChIP-Seq sample versus the control, and the statistical significance associated with each enriched target region is calculated from binomial distribution. The MACS/MACS2 method was one of the earliest developed methods and a good overall peak caller. It reduces analysis bias through the use of control data and local statistics and generates an empirical FDR. The findPeaks module in HOMER identifies peaks based on the principle that more sequencing reads are found in these regions than expected by chance. SPP is an R package designed for analyzing Illumina-generated ChIP-Seq data. It calculates a smoothed read enrichment profile along the genome and identifies significantly enriched sites compared to input control.

To ensure the robustness of analysis results, it is recommended to use more than one method for peak calling. Although IDR is usually used to measure the rate of irreproducible discoveries, which are peaks that are called in one replicate sample but not in another, it can also be used to compare peak calling results generated from different methods. The original use of IDR in assessing replicate reproducibility is based on the rationale that peaks of high significance are more consistently ranked across replicates and therefore have better reproducibility than those of low significance. As shown in Figure 11.7, to compare a pair of ranked lists of peaks identified in two replicates, IDR are plotted against the total numbers of ranked peaks. Since IDR computation relies on the use of both high significance (more reproducible) and low significance (less reproducible) peaks, peak calling stringency needs to be relaxed to allow generation of both high and low confidence calls. The transition in this plot from reliable signal gradually to noise is an index of overall experimental reproducibility. Because IDR is independent of any particular peak-calling method, it can be applied to compare the performance of different peak calling methods on a particular data set and therefore help pick the most appropriate method(s) (Figure 11.8). IDR can also be used to evaluate reproducibility across experiments and labs.

For proteins that bind to specific genomic sites, the fraction of reads in peaks (or FRiP) is an index of immunoenrichment and ChIP-Seq data quality. Usually only a small percentage of reads map to peak regions, and the majority of reads only represent background. As a general guideline, the ENCODE consortium sets 1% as the minimum for an acceptable FRiP with

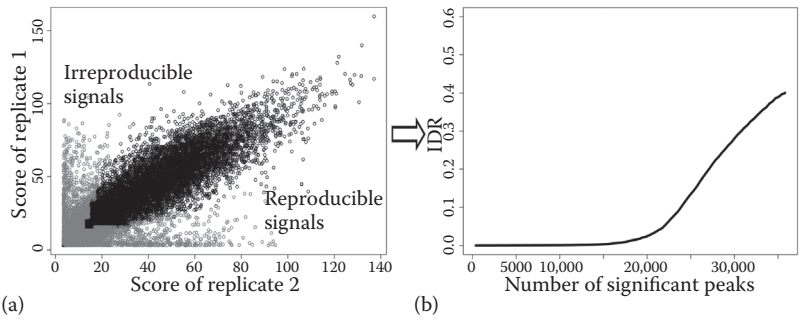

FIGURE 11.7
Use of irreproducible discovery rate (IDR) in assessing replicate reproducibility. (a) The distribution of the significance scores of the peaks identified in two replicate experiments. The IDR method computes the probability of being irreproducible for each peak, and classifies them as being reproducible (black) or irreproducible (gray). (b) The IDR at different rank thresholds when the peaks are sorted by the original significance score. (From T Bailey, P Krajewski, I Ladunga, C Lefebvre, Q Li, T Liu, P Madrigal, C Taslim, J Zhang, Practical guidelines for the comprehensive analysis of chip-seq data, *PLoS Computational Biology* 9, 2013:e1003326. Used under the terms of the Creative Commons Attribution License, http://creativecommons.org/licenses/by/3.0/, ©2013 Bailey et al.)

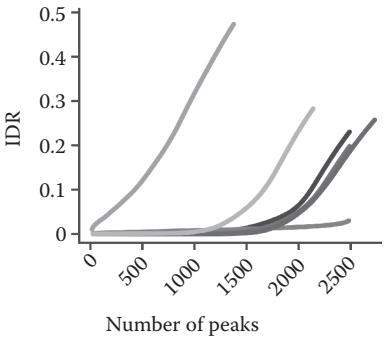

FIGURE 11.8
Evaluation of the performance of six peak callers using IDR. (Original study from Y Chen, N Negre, Q Li, JO Mieczkowska, M Slattery, T Liu, Y Zhang et al., Systematic evaluation of factors influencing ChIP-seq fidelity, *Nature Methods* 9, 2012:609–614. With permission.)

MACS as the peak caller using default parameters. As they can vary with the use of different peak callers and parameters, FRiP values must be derived from the same algorithm using same parameters in order for them to be comparable across samples or experiments.

11.3.3 Peak Visualization

Visualizing peaks in their genomic context allows identification of overlapping or nearby functional elements, and thereby facilitates peak annotation and data interpretation. Many peak callers generate BED files containing peak chromosomal locations, along with WIG and bedGraph track files, all of which can be uploaded to a genome browser for peak visualization. Examination of peak regions in a genome browser and comparison with other data/annotation tracks allow identification of associated genomic features, such as promoters, enhancers, and other regulatory regions. BEDTools can also be applied to explore relationships between peaks and other genomic landmarks such as nearby protein-coding or noncoding genes.

11.4 Differential Binding Analysis

Binding of DNA-interacting proteins to their target genomic regions is a quantitative process, that is, they occupy these regions at different rates under different conditions. This is due to regional accessibility, presence/absence of other protein partners, and/or other factors that regulate their binding. Differential binding analysis answers the question of how a target

protein changes its DNA-binding pattern under different conditions. There are two different approaches for this analysis, with one qualitative and the other quantitative. The qualitative approach compares peaks called in different conditions, and divides them into "shared" and "unique" [316]. This approach is simple but it does not use the quantitative information generated in the peak calling process, so it is best used to produce a rough initial estimation of differential binding. The quantitative approach, based on analysis of read counts or read densities in peak regions, generates statistical assessment of the degree of differential binding between conditions. As this is similar to the RNA-Seq-based differential expression analysis, data normalization is required to adjust for systematic biases that are unrelated to biological factors. For the comparison of two or more ChIP-Seq samples, such biases include immunoprecipitation efficiency and sequencing depth.

Similar to normalizing RNA-Seq data, adjusting for sequencing depth is the simplest approach. In this approach, the total numbers of reads in different samples are adjusted by multiplying a scaling factor to each sample to the same target level, for example, the median or lowest total read count among the samples. The basic assumption for this approach is that the overall number of binding sites for a target protein does not change across different experimental conditions. Although this approach is simple and straightforward, it does not take into consideration the differences in the signal-to-noise ratios that are often observed in different samples. If one sample library is noisier and contains more background reads, these reads, while not representing true signals, are still counted in the total read number. This situation will therefore lead to bias in the normalized data.

There are several currently available normalization approaches that consider this issue of signal-to-noise ratio variation among samples. For example, the normalization procedure used in diffReps first identifies and removes regions with low read count (mostly background noise) [295]. The subsequent normalization is then based on the remaining regions, using a linear procedure similar to that used by DESeq. Another similar approach uses only reads mapped to peaks. In this modified sequencing depth-based normalization approach, the total number of reads mapped to the peak regions are used as the basis for calculating scaling factor for each sample [296]. Using this approach, the normalized peak signal is computed as the original peak sequence read count being scaled by the sum of read counts of all peaks, that is,

$$Z_{i,j} = \frac{X_{i,j}}{\sum_{j=1}^{N} X_{i,j}}$$

where $Z_{i,j}$ and $X_{i,j}$ are the normalized and original peak signal for sample i and peak j, and N is the total number of called peaks.

Normalization methods that were previously developed for microarray data have also been adapted for ChIP-Seq data. MAnorm uses a nonlinear normalization process [297] that is similar to the MA plot approach used for microarray data. ChIPnorm uses a modified version of quantile normalization [298]. A locally weighted regression (LOESS) normalization approach for ChIP-Seq data [299] is similar to the LOESS procedure applied to cDNA microarray data normalization. All these approaches assume that the overall binding profile of the target protein does not vary across different conditions.

Besides all the normalization approaches introduced earlier, good experimental design and consistent experimental procedure can minimize data variability in different samples and groups, thereby alleviating the burden on posterior normalization. For example, processing all samples side by side using the same experimental procedure and parameters, such as the same antibody, by the sample operator, will minimize sample-to-sample variability. When conducting an experiment in this way, the simpler normalization approach based on total library read count can be sufficient.

Since the ChIP-Seq-based quantitative analysis of differential binding is similar to the RNA-Seq-based differential expression analysis, packages such as edgeR and DESeq can be applied here. Table 11.2 lists some of the packages that are designed for ChIP-Seq differential binding analysis. Like those devised for RNA-Seq-based differential expression analysis, these

TABLE 11.2

Packages Developed for ChIP-Seq Differential Binding Analysis

Name	Description	Reference
ChIPComp	Differential binding analysis taking into consideration controls, signal-to-noise ratios, replicates, and multifactor experimental design	317
ChIPDIff	Differential histone mark analysis based on Hidden Markov model	318
ChIPnorm	Carries out quantile normalization for differential binding sites identification	298
ChromaSig	Performs unsupervised learning to determine significant patterns of chromatin modifications across multiple experiments	319
DBChIP	Identifies differentially bound punctate binding sites in multiple conditions using RNA-Seq differential expression approaches and accommodates controls	320
DiffBind	Uses statistical tests used in RNA-Seq packages edgeR and DESeq to process peak sets and identify differentially bound regions	321
diffReps	Detects and annotates differential chromatin modification hotspots	295
DIME	Differential binding analysis using a finite exponential-normal mixture model	322
MAnorm	Conducts an MA-plot-based normalization prior to quantitative comparison	297
MMDiff	Takes a multivariate nonparametric approach to testing differential binding	323

packages are designed on certain assumptions and therefore the user needs to be aware of these assumptions and ensure they are fulfilled prior to using them. For example, DIME is based on the assumption that a large proportion of peaks are common across the conditions under comparison.

11.5 Functional Analysis

Often the data gathered from a ChIP-Seq study is used to understand gene expression regulation and associated biological functions. To conduct functional analysis, peaks are first assigned to nearby genes. While it is debatable on what genes a peak should be assigned to, a straightforward approach is to assign it to the closest gene. Once peaks are assigned to target genes, an integrated analysis of ChIP-Seq and gene expression data (more on this later) can be carried out. Furthermore, Gene Ontology (GO), biological pathway, gene network, or gene set enrichment analyses can be conducted using similar approaches as described in Chapter 7. Prior to carrying out these gene functional analyses, one should also bear in mind that the peak-to-gene assignment process is biased by gene size, as the presence of peak(s) has a positive correlation with the length of a gene. In addition, the distribution of gene size in different functional annotations such as GO categories is not uniform, with some categories having an excess number of long genes and others having more short genes. To solve the problems caused by different gene size, methods that adjust for the effects of gene size should be used, such as ChIP-Enrich [324].

11.6 Motif Analysis

One of the goals of ChIP-Seq data analysis is to identify DNA-binding motifs for the protein of interest. A DNA-binding motif is usually represented by a consensus sequence, or more accurately, a position-specific frequency matrix. Figure 11.9a shows an example of such a DNA-binding motif, the one bound by a previously introduced transcription factor NRF2 (see Chapter 2). To identify motifs from ChIP-Seq data, all peak sequences need to be assembled and fed into multiple motif discovery tools. Some of the commonly used motif discovery tools are Cistrome [325], Gibbs motif sampler (part of CisGenome), HOMER (findMotifs module), MEME-ChIP [326], QuEST [308], RSAT peak-motifs [327], and ChIPMunk [328]. The motif discovery phase usually ends up with one or more motifs, with one being the binding site of the target protein and others being that of its partners. The discovered

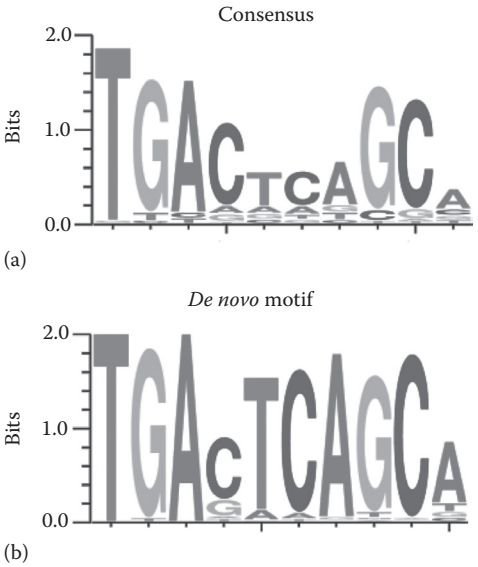

FIGURE 11.9
The consensus DNA binding motif of the transcription factor NRF2. (a) The currently known NRF2-binding motif. (b) The result of a *de novo* motif analysis using NRF2 ChIP-Seq data. (From BN Chorley, MR Campbell, X Wang, M Karaca, D Sambandan, F Bangura, P Xue, J Pi, SR Kleeberger, DA Bell, Identification of novel NRF2-regulated genes by ChIP-Seq: Influence on retinoid X receptor alpha, *Nucleic Acids Research* 2012, 40(15):7416–7429. With permission.)

motif(s) can be compared with currently known motifs to detect similarities, find relationships with other motifs, or locate other proteins that might bind at or near the peak region as part of a protein complex. Tools for motif comparison include STAMP [329] and Tomtom [330]. Motif enrichment analysis can also be carried out to find out if other known motifs are enriched in the peak regions using tools such as CentriMo [331]. Finally motif scanning and mapping by tools like FIMO [332] allows visualization of the discovered motif(s) in the ChIP-Seq peak areas. Some of these tools have been integrated into motif analysis pipelines, such as the MEME Suite (http://meme-suite.org), which includes MEME-Chip, Tomtom, CentriMo, and FIMO.

11.7 Integrated ChIP-Seq Data Analysis

Because genomic functions are to a large degree controlled by concerted binding of a wide array of DNA-interacting proteins, integrated analysis of ChIP-Seq data sets generated for a multitude of these proteins affords new opportunities to gain a comprehensive overview of the functional states of

a genome and the host cell. As a good example, such an integrated analysis has led to the discovery of a large number of chromatin states, each of which display distinct sequence motifs and functional characteristics [333]. The discovery of these chromatin states was achieved with the use of a multivariate hidden Markov model on a large collection of ChIP-Seq data, generated for 38 different histone methylation and acetylation marks, H2AZ (a variant of histone H2A), RNA polymerase II, and CTCF (a transcriptional repressor). Besides meta-analysis of multiple ChIP-Seq data sets, integrated analysis of ChIP-Seq with other genomics data, such as RNA-Seq data, offers further information on genome function and regulation. The majority of protein factors used in various ChIP-Seq studies are transcription factors and histones that carry a large array of modified marks, all of which are key regulators of genome transcription. Coupled analysis of matched ChIP-Seq and RNA-Seq data augments the utility of both data types, and provides new insights that cannot be obtained from analyzing either data type alone. To carry out integrative analysis of ChIP-Seq and RNA-Seq data, Bayesian mixed models [334] can be applied. In addition, tools such as CEAS [314] and ChIPpeakAnno [315] can also be used to help investigate the correlation between the DNA binding profile and regulation of nearby gene transcription.

12

Epigenomics and DNA Methylation Analysis by Next-Generation Sequencing (NGS)

The genomic information embedded in the primary nucleotide sequence of DNA is modulated by epigenomic code generated from chemical modifications of DNA bases and key DNA-interacting proteins such as the histones. The methylation of cytosines leading to the formation of 5-methylcytosines (5mCs), for example, provides a major means for the modification of the primary DNA code. As detailed in Chapter 2, DNA methylation plays important roles in many biological functions such as embryonic development, cell differentiation, and stem cell pluripotency, by regulating gene expression and chromatin remodeling. Abnormal patterns of DNA methylation, on the other hand, lead to diseases such as cancer. DNA methylome analysis, as a key component of epigenomics, has for many years been conducted with the use of microarrays (such as the Illumina Infinium 450K BeadChips). Although microarrays are low-cost and easy to use, their inherent constraints, such as limited genomic coverage from the use of preselected probes and being available for only a few model organisms, have limited their use. In comparison, next-generation sequencing (NGS) offers a more unbiased, comprehensive, and quantitative approach for the study of DNA methylation status in a wide array of species. This chapter is focused on DNA methylation sequencing data generation and analysis. For epigenomic studies that involve interrogation of histone modifications, ChIP-Seq (covered in Chapter 11) can be used.

12.1 DNA Methylation Sequencing Strategies

Because the DNA polymerases used in the regular NGS sequencing library construction process cannot distinguish methylated from unmethylated cytosines, the DNA methylation pattern is usually not retained in the process. In order to study DNA methylation status with NGS, two strategies are usually used, with one based on bisulfite conversion and the other on methylated DNA enrichment. The first strategy employs a chemical conversion process, which uses sodium bisulfite to deaminate unmethylated cytosines. After the conversion, unmethylated cytosines in a DNA molecule are converted to uracils, while 5mCs in the same molecule are retained since they are nonreactive. The subsequent sequencing of the converted DNA, therefore,

reads unmethylated cytosines as thymines, and methylated cytosines still as cytosines. The efficiency and specificity of this process can be monitored and optimized through the use of certain methylated and unmethylated DNAs as controls. Based on genomic coverage, bisulfite conversion-based DNA methylation sequencing, or simply bisulfite sequencing, can be further divided into different subcategories.

12.1.1 Whole-Genome Bisulfite Sequencing (WGBS)

As the name suggests, whole-genome bisulfite sequencing (WGBS) analyzes cytosine methylation in the entire genome, that is, the methylome. In preparing WGBS libraries from total genomic DNA, regular DNA library construction protocols need to be modified. For example, if adapters are added prior to the bisulfite conversion step, they must not contain unmethylated cytosines, that is, all cytosines in the adapter sequence must be methylated. In the polymerase chain reaction (PCR) step, a polymerase that can tolerate uracil residues needs to be used. As a result of the conversion and subsequent PCR amplification, the two DNA strands that were originally complementary are no longer complementary. Instead, four strands that are distinct from the original complementary strands are generated (Figure 12.1). Furthermore, the conversion leads to reduced sequence complexity due to

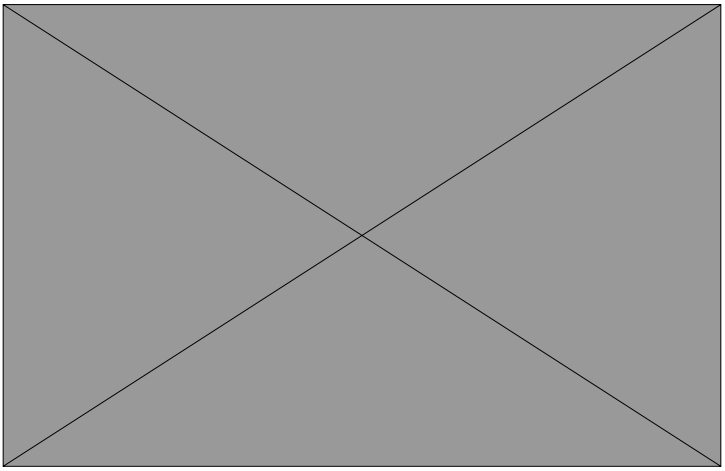

FIGURE 12.1
Major steps of bisulfite sequencing. Prior to bisulfite treatment, the two strands of DNA are first separated by denaturation. The bisulfite treatment then converts unmethylated, but not methylated, cytosines to uracils. The two strands from the treatment, BSW and BSC, are then subjected to PCR amplification. This leads to the generation of four strands (BSW, BSWR, BSC, and BSCR), all of which are distinct from the original Watson and Crick strands. (From Y Xi, W Li, BSMAP: Whole genome bisulfite sequence MAPping program, *BMC Bioinformatics* 2009, 10:232. Used under the terms of the Creative Commons Attribution License, http://creativecommons.org/licenses/by/2.0, ©2009 Xi and Li.)

underrepresentation of cytosines in the generated reads. Without the use of an external sequencing library to create a calibration table for the base caller, the reduced sequence complexity will lead to a high base call error rate. Therefore, use of such a calibration library, such as the phiX174 control library for Illumina sequencing, is needed for bisulfite sequencing data generation.

The power to detect DNA methylation levels, and differentially methylated sites or regions between different experimental conditions (e.g., disease versus normal), is dependent on the sequencing depth and the number of biological replicates. The National Institutes of Health (NIH) Roadmap Epigenomics Project recommends at least two replicates per condition with a combined depth of at least 30× [335]. While this can be used as a general guideline for many projects, key statistical issues, that is, within-condition biological variation and between-condition difference, determine the actual detection power. Consistent with the aforementioned recommendations, currently available data suggests a per-sample coverage of 5× to 15× [336]. Sequencing above this range may not be as cost-effective as adding more biological replicates in reaching higher detection power.

12.1.2 Reduced Representation Bisulfite Sequencing (RRBS)

Although WGBS enables detection of methylation in the entire genome, the cost associated with such analyses was high at the earlier days of NGS. To reduce the cost, strategies such as reduced representation bisulfite sequencing (RRBS) [337] were devised. To perform RRBS, genomic DNA is first digested with a methylation-insensitive restriction enzyme (such as MspI) that recognizes CpG-containing restriction sites. The digested DNA products are then separated and size selected to pick fragments in a certain size range for bisulfite conversion and then sequencing. While it only interrogates a few percent of the genome, RRBS provides a rough survey of DNA methylation in the genome. If particular region(s) of the genome are found to be of special interest, they can be captured for subsequent sequencing using approaches such as ligation capture [338,339], bisulfite padlock probes [340], or liquid hybridization capture [341].

12.1.3 Methylation Sequencing Based on Methylated DNA Enrichment

Different from the aforementioned bisulfite conversion-based methods, the methylated DNA enrichment strategy captures methylated DNA for targeted sequencing. The target DNA capturing is achieved with the use of 5mC antibodies or proteins that bind to methylated cytosines. One of the methods based on this strategy is MeDIP-Seq, or methylated DNA immunoprecipitation coupled with NGS. In this method, antibodies against 5mC are used to precipitate methylated single-stranded DNA fragments for sequencing. Another commonly used method is MBD-Seq, or methyl-CpG-binding domain capture (MBDCap) followed by NGS. MBD-Seq utilizes proteins such as MBD2 or MECP2 that contain the methyl-CpG binding domain

to enrich for methylated double-stranded DNA fragments. In one type of MBDCap method called MIRA (Methylated-CpG Island Recovery Assay), a protein complex of MBD2 and MBD3L1 (methyl-CpG-binding domain protein 3-like-1) is used to achieve enhanced affinity to methylated CpG regions. While MeDIP-Seq and MBD-Seq usually generate highly concordant results, there are some differences between these two approaches. MeDIP-Seq can detect both CpG and non-CpG methylation, while MBD-Seq is focused on methylated CpG sites because of the binding affinity of MBD. At methylated CpG sites, MeDIP tends to enrich at regions that have low CpG density, while MBD-Seq favors regions of relatively higher CpG content [342,343].

In principle, these enrichment-based methods are very similar to ChIP-Seq (Chapter 11), based on the same process of specific protein-based DNA capture, protein-DNA complex affinity binding, and target DNA elution. Likewise, their sequencing data generation and subsequent analysis are also similar to those in ChIP-Seq. Therefore, the data analysis methods covered in Chapter 11 equally apply to the analysis of sequencing data generated by MeDIP-Seq, MBD-Seq, or other methylated DNA enrichment-based NGS methods. This chapter is, therefore, mostly focused on the analysis of bisulfite sequencing data.

12.1.4 Differentiation of Cytosine Methylation from Demethylation Products in Bisulfite Sequencing

Among the three 5mC demethylation intermediate products—5hmC, 5fC, and 5caC (see Chapter 2)—5hmC is not reactive to the sodium bisulfite, whereas 5fC and 5caC are reactive and converted to uracils. During the subsequent sequencing, as a consequence, 5hmC cannot be differentiated from 5mC, while 5fC/5caC cannot be differentiated from unmethylated cytosines. However, since these demethylation products are usually at levels that are much lower than 5mC or unmethylated cytosines in cells, their interference is minimal. For samples prepared from the brain or embryonic stem cells where 5hmC is relatively high, strategies such as oxBS-Seq [344] are available to differentiate 5mC from 5hmC. Some third-generation single-molecule sequencing technologies, such as the Pacific Biosciences's SMRT sequencing and nanopore sequencing, have been shown to be capable of differentially detecting these modifications without relying on bisulfite conversion [345–347].

12.2 DNA Methylation Sequencing Data Analysis

12.2.1 Quality Control and Preprocessing

After raw data generation, the quality control (QC) step removes low-quality reads or base calls as they directly affect subsequent alignment to the reference

genome and DNA methylation site identification. The general data QC steps detailed in Chapter 5 should be performed for their removal. Other QC steps include adapter trimming, as some sequencing reactions may run through DNA inserts into adapters. In addition, for MspI-digested RRBS libraries, the DNA fragment end repair step during the library construction artificially introduces two bases (an unmethylated cytosine and a guanine) to both ends, both of which should be trimmed off as well. Tools such as Trim Galore (a wrapper tool using Cutadapt and FastQC) [348] can be used for these trimming steps, especially removing the two artificially introduced bases in RRBS reads derived from MspI digestion. Besides these general-purpose QC tools, some packages designed for bisulfite sequencing reads processing, including BSmooth [349] and WBSA [350], also contain QC modules.

12.2.2 Read Mapping

In order to identify methylated DNA sites, sequencing reads derived from bisulfite conversion or methylated DNA enrichment need to be first mapped to the reference genome. Mapping of reads generated from the enrichment-based methods is rather straightforward, and like mapping ChIP-Seq reads, is usually conducted with general aligners, such as Bowtie, BWA, or SOAP. Mapping of bisulfite sequencing reads, however, is less straightforward. This is because through the bisulfite conversion and the subsequent sequencing process, a converted unmethylated cytosine is read as a thymine (T), or an adenine (A) on the complementary strand, whereas a methylated cytosine remains as a cytosine (C), or a guanine (G) on the complementary strand (see Figure 12.1). The conversion therefore has several implications for the read mapping process:

- Fuzziness in mapping—A T in the reads could be mapped to a C or T in the reference sequence, thus complicating the searching process.
- Increase in search space—This is partly caused by the non one-to-one mapping, and more seriously, by the generation of the four bisulfite-converted strands that are distinct from the reference strands (also as illustrated in Figure 12.1), leading to a significant increase in search space.
- Reduction in sequence complexity—The amount of C's in the bisulfite reads is significantly reduced, and this reduction in sequence complexity leads to higher levels of mapping ambiguity. Consequently, aligning bisulfite sequencing reads to the reference genome is not as straightforward as that for ChIP-Seq or other DNA deep sequencing data.

There are two general strategies for mapping bisulfite sequencing reads: (1) replacing all C's in the reference genome with the wild-card letter Y to

match both C's and T's in the reads; and (2) converting all C's in the reference sequence and reads to T's, and then aligning with a seed-and-extend approach. Aligners that use the wild-card approach include BSMAP [351], Pash [352], and RRBSMAP (a version of BSMAP specifically tailored for RRBS reads) [353]. In the example of BSMAP, it uses SOAP for carrying out read alignment, and deploys genome hashing and bitwise masking for speed and accuracy. BSMAP indexes the reference genome using a hash table containing original reference seed sequences and all their possible bisulfite conversion variants through the replacement of C's with T's. After determining the potential genomic position of each read by looking up the hash table, for the T's in each bisulfite read that are mapped to reference genome position(s) where the original reference bases are C's, BSMAP masks as C's. The masked bisulfite reads are then mapped again to the reference genome.

Aligners such as BatMeth [354], Bismark [355], BRAT-BW [356], BS-Seeker/BS-Seeker2 [357,358], and MethylCoder [359] use the other three-letter approach. Among these aligners, Bismark is commonly used. To carry out its alignment (illustrated in Figure 12.2), Bismark first converts C's in the reads into T's, and G's into A's (equivalent of the C-to-T conversion on the

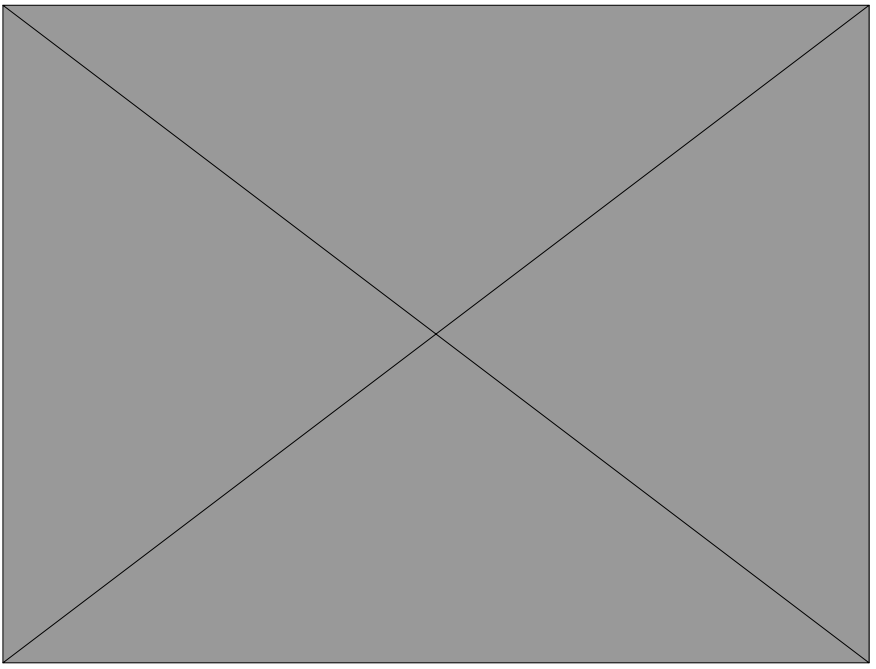

FIGURE 12.2
The "three-letter" bisulfite sequencing read alignment approach used by Bismark. (Adapted from F Krueger, SR Andrews, Bismark: A flexible aligner and methylation caller for Bisulfite-Seq applications, *Bioinformatics* 2011, 27:1571–1572. With permission.)

complementary strand). This conversion process is also performed on the reference genome. The converted reads are then aligned, using Bowtie or Bowtie2, to the converted reference genome in four parallel processes (also refer to Figure 12.1), out of which a unique best alignment is determined (alignment 1 in Figure 12.2). Some benchmark studies [360] found that in comparison with other aligners, Bismark offers a good combination of speed, accuracy, and genomic region coverage.

After mapping, distribution of the mapped reads in the genome should be examined. This provides an initial survey of the results, and at the same time it also serves as an additional QC step. For example, duplicate reads that map exactly to the same position are most likely PCR artifacts and should be removed from further analysis. Other abnormalities in distribution, such as significantly unbalanced numbers of reads mapped to the two DNA strands in a genomic region, should be checked with caution, and the reads may need to be filtered out. Some tools, such as BSeQC [361], also carry out postalignment QC processing using the SAM/BAM alignment files as input.

12.2.3 Quantification of DNA Methylation

After bisulfite read mapping, uniquely mapped reads need to be aggregated to quantify the methylation level (also called β-value) at individual cytosine sites in the reference genome, based on the frequency of C's (methylated cytosines) and T's (unmethylated cytosines) in reads mapped to each of these sites. This quantitative step can be performed by dividing the total number of C's by the total combined number of C's and T's that are mapped to each site. All of the bisulfite sequence mappers introduced in the previous section generate this information. Postmapping tools such as GBSA [362] and methylkit [363] can also be used for methylation quantification. For this quantification step, it should be noted that the involved calculations usually require a minimum depth (e.g., at least three reads) at the individual sites to avoid deriving unreliable methylation levels from too few reads.

Besides quantifying methylation levels at individual cytosine sites, DNA methylation quantification is also often calculated on a regional basis, usually performed to facilitate comparisons between multiple samples. Different approaches can be used for regional DNA methylation quantification. One of the approaches is to divide the genome into a number of bins, and the mean of methylation levels at individual cytosine sites within each bin is used to represent the binned area. Alternatively, each bin's methylation level can be calculated as the overall proportion of methylated cytosines among all cytosines within the bin. Other approaches also use sliding windows, instead of individual bins, for regional methylation quantification.

These calculations, however, do not take into consideration the possible existence of SNPs that involve the change from C to T. Some algorithms, such as Bis-SNP, remove this potential confounding factor by distinguishing

bisulfite conversion from genetic variants. The use of sequence reads from the complementary strand makes this possible, because a T produced from bisulfite conversion will have a G on the opposite strand, whereas a C → T SNP will have an A on the other strand.

Different from the bisulfite-conversion-based sequencing methods, the methylated DNA enrichment sequencing approaches such as MeDIP-Seq and MBD-Seq cannot quantify methylation at the single-nucleotide resolution. In addition, the absolute levels of DNA methylation cannot be obtained from the enrichment-based methods, as the sequence read counts from these methods are a function of both absolute DNA methylation levels and regional CpG content. Since these approaches are based on affinity immunoprecipitation and more similar to ChIP-Seq, analytical methods developed for ChIP-Seq data analysis, including background determination, normalization, and peak detection, can be applied for quantification of DNA methylation by these approaches. As an output, the degree of DNA methylation can be summarized as coverage over a predefined region, such as per gene, promoter, or certain-sized bin.

12.2.4 Visualization of DNA Methylation Data

Visualizing DNA methylation data serves at least two purposes. First, the distribution pattern of DNA methylation may be discerned through visualization. Second, visual examination of known DNA methylation regions and other randomly selected regions also offers data validation and a quick estimate of data quality. One method to visualize bisulfite sequencing data and associated information, such as depth of coverage, is through the use of bedGraph files. This standard format (Figure 12.3), compatible with most genome browsers and tools including the Washington University EpiGenome Browser [364], can be directly generated from many of the methylation quantification tools such as Bismark, GBSA, and methylkit. Figure 12.4 shows an example of displaying methylation level along with read depth in the genome.

Alternatively, DNA methylation quantification results can be saved in tab-delimited files and then converted to bigBed or bigWig formats [365]. Both formats are compatible with and enable visualization of the methylation

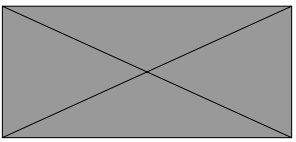

FIGURE 12.3
An example of the bedGraph file format. It includes a track definition line (the first line), followed by track data lines in a four-column format (i.e., chromosome, chromosome start position, chromosome end position, and data value).

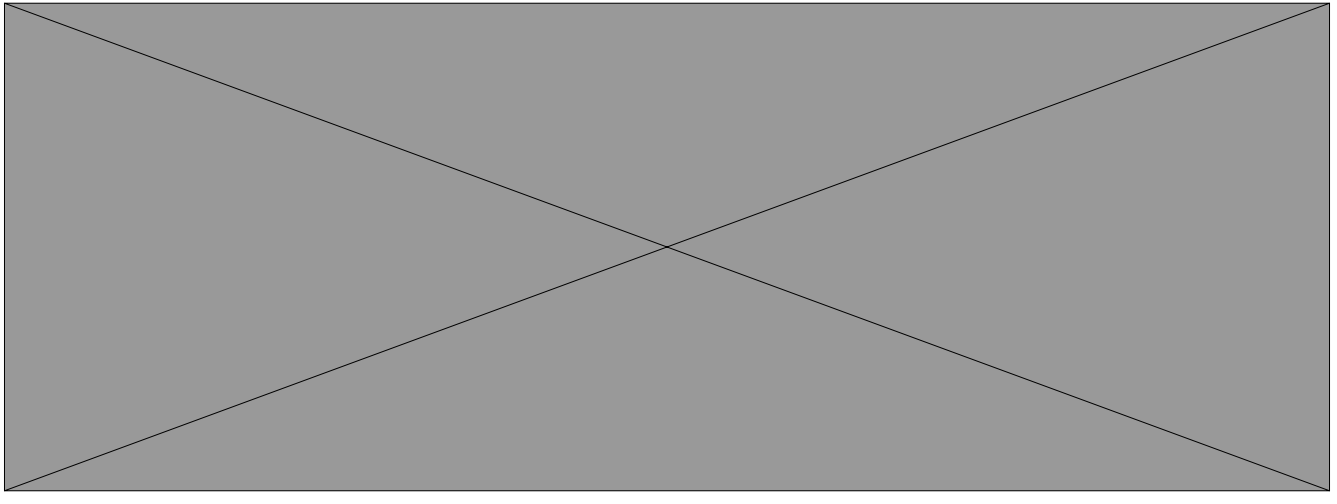

FIGURE 12.4
Visualization of DNA methylation data in a genome browser. Shown here is the methylC track in the Washington University EpiGenome Browser for a region of the human chromosome 7 where the *HOXA* gene cluster is located. The original WGBS data was collected from H1 human embryonic stem cells. Both DNA methylation levels (represented by vertical bars) and read depth (the smoothed curve) are displayed in a strand-specific fashion. A close-up view of the boxed region is shown on the top, with the left axis marking the DNA methylation level and the right marking read depth. (Modified from X Zhou, D Li, RF Lowdon, JF Costello, T Wang, methylC Track: Visual integration of single-base resolution DNA methylation data on the WashU EpiGenome Browser, *Bioinformatics* 2014, 30:2206–2207. With permission.)

results in web-based genome browsers such as the UCSC Genome Browser, or desktop-based ones such as IGV and IGB. An additional option is to export DNA methylation data to the VCF format using tools such as GobyWeb [366], and then visualize with genome browsers such as IGV and Savant.

12.3 Detection of Differentially Methylated Cytosines or Regions

One frequent goal of DNA methylation analysis is to compare and identify specific cytosines or genomic regions that show differential methylation between conditions. To identify differentially methylated cytosines or regions (DMCs or DMRs), different statistical approaches have been used. These include parametric tests such as *t*-test or ANOVA, and nonparametric tests such as Fisher's exact test, Wilcoxon test, chi-squared test, or Kruskal-Wallis test. The parametric tests assume normal distribution, which is likely to be violated for DNA methylation data as it tends to follow bimodal distribution. As a result, most currently available tools use nonparametric tests, for example, WBSA employs the Wilcoxon test. The package methylKit identifies DMCs/DMRs with the use of Fisher's exact test for comparison of groups without replicates, and logistic regression for comparison involving multiple samples per group. BSmooth uses a modified *t*-test with local data smoothing to increase detection power. Another package called Methy-Pipe [367] detects DMRs using the Mann-Whitney U test with a sliding window approach. More sophisticated approaches include the use of a beta-binomial hierarchical model in MOABS [368] and Shannon entropy in QDMR [369]. Besides these different statistical tests or models, another notable difference among these packages is in how biological replicates are handled. Earlier methods tend to pool replicate data for DMC/DMR detection, leading to the loss of information on sample-to-sample variation. Newer methods, such as BSmooth and MOABS, are more replicate-aware and provide estimation on biological variation, thereby increasing detection power. On multiple testing correction, FDR is mostly used, while other methods are also reported, such as a sliding linear model (SLIM) method used by methylKit.

Data obtained from approaches based on methylated DNA enrichment follows the negative binomial distribution, like the ChIP-Seq and RNA-Seq data. Therefore, they can be analyzed to identify DMRs using algorithms developed for RNA-Seq-based differential expression. For example, tools such as EdgeR and DESeq can be directly used. In some DNA methylation analysis tools, such as Repitools [370], EdgeR is directly called.

12.4 Data Verification, Validation, and Interpretation

The DMCs/DMRs identified in the previous step need to be verified and further validated. Verification is usually conducted on the same set of samples as those used for DNA methylation sequencing data generation. Further validation, on the other hand, is carried out on a new set of samples. For DNA methylation sequencing data verification and validation, the following techniques are often used: methylation-specific PCR (such as MethyLight), or methylation-independent PCR coupled with pyrosequencing, mass spectrometry, or combined bisulfite restriction analysis (COBRA).

Data interpretation is a key step in translating a list of DMCs/DMRs into a mechanistic understanding of the biological process under study. Most potential effects of the DMCs/DMRs can only be revealed through examining them in their genomic context. Tools such as EpiExplorer [371], GBSA, methylKit, or WBSA can be very helpful in this regard via placing them in the context of other genomic features such as CpG islands, transcription start sites, histone modification marks, or repetitive regions. DMCs/DMRs can also be mapped to nearby genes, which can then be subjected to gene set enrichment, biological pathway, and gene networking analyses. In this regard, the web-based Genomic Regions Enrichment of Annotations Tool (or GREAT) [372] can be used to map DMCs/DMRs to nearby genes, while controlling for gene size difference and distance, for functional annotation and interpretation.

13

Metagenome Analysis by Next-Generation Sequencing (NGS)

A small amount of environmental sample, such as a handful of soil, is rich in microbial life, but the number of microbial species in such a sample is unknown. The microbiome on or within our body contains tens of thousands, if not more, species of bacteria, fungi, and archaea. Besides their tremendous species diversity, the composition, as well as function, of such microbial communities is not static but constantly changing according to the status of their environment. Our current understanding of these diverse and dynamic microbial communities is still significantly lacking, as most of our knowledge comes from culturable species. For those that still cannot be cultured in the lab, which comprise the majority of microorganisms on earth, we know very little. Metagenomics offers an important approach to study microbial diversity in these environmental communities without relying on artificial culturing. Also referred to as environmental or community genomics, metagenomics examines all genomes existing in a microbial community as a whole without the need to capture or amplify individual genomes. Through simultaneous analysis of all DNA molecules present in a microbial community, metagenomics provides a profile of taxonomic composition and functional status of the community and its environment.

Before the advent of next-generation sequencing (NGS), metagenomics studies were usually conducted with DNA cloning combined with Sanger sequencing. In this approach, DNA extracted from a microbial community is first fragmented, and then the DNA fragments are cloned into plasmid vectors for amplification in order to produce enough materials for Sanger sequencing. With the continuous development and significant cost drop in NGS technologies, massively parallel metagenomic sequencing has quickly replaced this traditional low-throughput approach and become a major approach for studying various microbial communities. The high sensitivity offered by the NGS approach provides direct access to unculturable species that were previously "invisible" to analysis [373]. The Human Microbiome Project exemplifies the use of NGS in interrogating complex metagenomes, such as those at different sites of the human body including the gastrointestinal tract. The application of NGS in metagenomic analysis of a large variety of other microbial communities, such as those in soil, the phyllosphere, the ocean, and those associated with bioremediation and biofuel generation, has led to an exponential increase in the number of metagenomes studied.

Compared to the NGS data generated from a single species (most chapters in this book deal with individual species), the metagenomics data from microbial community sequencing is much more complicated. Each metagenome contains DNA sequences from a large but unknown number of species, including viruses, bacteria, archaea, fungi, and microscopic eukaryotes. To further complicate the situation, the relative abundance of these species varies widely. In comparison to sequencing reads collected from a single species, metagenomic sequencing reads contain much higher heterogeneity because of the tremendous genome diversity in each microbial community. Also because of the tremendous DNA sequence complexity contained in the metagenome, most metagenomic sequencing efforts can only sample part of the DNA pool. As a result of this limited sampling in a highly diverse DNA space, metagenomic NGS data is highly fragmented and has low redundancy. Due to the lack of redundant (i.e., partially overlapping, not duplicate) reads, metagenomic NGS data has an inherently higher error rate when compare to single-genome sequencing. All these differences between metagenomic and monogenomic NGS data require an entirely different set of tools for NGS-based metagenome data analysis for microbial community structural and functional profiling.

13.1 Experimental Design and Sample Preparation

Metagenomics studies aim to determine identities and relative abundance of different members, or taxa, in a microbial community, and how environmental factors affect the composition and function of these communities. To achieve this by sequencing, there are two general approaches: whole-genome shotgun (WGS) metagenomic sequencing and targeted metagenomic sequencing. The WGS approach provides random sampling of all genomes contained in an environmental or host-associated microbial sample. To carry out WGS, total DNA extracted from such a sample is first broken into small fragments, which are then sequenced.

In the targeted approach, genomic component(s) that are shared among different species are first amplified with polymerase chain reaction (PCR) and the amplicons are then sequenced. The most commonly used target in this approach is the 16S rRNA gene, while other genes that code for specific protein functions (such as resistance to specific antibiotics) or noncoding genes are also used. The 16S rRNA gene, being considered as the universal clock of life [374], is usually used as a surrogate marker for measuring the relative abundance of different operational taxonomic units (OTUs, a metagenomics term to describe a species or a group of species when only DNA sequence information is available). By focusing on the 16S rRNA gene or other specific genomic target(s), this approach greatly reduces complexity in the generated

data, thereby achieving deeper coverage and accommodating more samples. It should be noted that the 16S-rRNA-based approach only produces approximate estimation of relative taxonomic abundance, due to 16S rRNA copy number variation in some species and the fact that the standard 16S rRNA PCR primers may not bind to their supposed target sites in all cases because of random mutation. In comparison, the WGS approach, while relatively lacking on depth and affordability, takes an unbiased path to offer a comprehensive assessment of genome content in the community, and thereby provides in-depth information on community composition and function. This chapter is focused on WGS metagenome sequencing data analysis.

13.1.1 Metagenome Sample Collection

The success of a metagenomics project is to a degree dependent on factors that are not related to genomics. One such factor is how much is known about the habitat where study samples will be collected. The more physically, chemically, and ecologically characterized the habitat is, the more knowledge will be gathered from the metagenomic NGS data. In-depth characterization and detailed description of the sampling environment is one foundation of a successful metagenomics experiment. Keeping detailed metadata on the habitat and the sampling process (such as characteristics of the general environment, geographical location, and specific features of the sampling locales and the sampling method) is of great importance to downstream data interpretation.

As the composition and complexity of a metagenome sample are determined by the habitat and the sampling site, the unique characteristics of a sampling environment, along with the question to be answered or specific hypothesis to be tested, eventually determine how many reads are required. It should also be emphasized that since where the samples are collected directly shapes the outcome, the sampling sites must be representative of the habitat under study. In order to collect representative samples, information on spatial and temporal variation in the habitat must be known prior to sample collection. If this information is not available, a small-scale trial shotgun sequencing run might prove helpful with a small number of samples sequenced. Alternatively, a targeted 16S rRNA amplicon sequencing can also be used to survey the diversity of the microbial community.

13.1.2 Metagenome Sample Processing

DNA extraction is the first and also a key step in metagenome sequencing sample preparation. The DNA extracted from this step should represent all, or at least most, members of the sampling community and their relative abundance, be of high purity and free of contaminants that might interfere with the subsequent sequencing library construction. While this step might be routine in conventional genome sequencing for a single organism,

extracting high-quality DNA from microbial community samples collected from various habitats poses challenges. For example, humic acids, polysaccharides, tannins, and other compounds are major contaminants in environmental samples such as those from the soil, which if not removed can lead to inhibition of enzymes used in library construction. In host-associated habitats such as the human gut, host DNA is the major potential contaminant.

Besides purity, extracting DNA in equal efficiency from different community members is another challenge, as the optimal condition of cell lysis for DNA release from one group of microbes may not be ideal for another. For example, mechanical disruption is often used for breaking up cells in metagenomics studies, but by using this method DNA released from easily lysed cells may be sheared to fragments when tougher cells are eventually disrupted. While these challenging issues should be acknowledged and addressed, they are not insurmountable and robust extraction protocols are available for various habitats [375].

Advancements in sequencing library preparation protocols have reduced the amounts of DNA required considerably to lower nanogram levels (e.g., the Nextera XT protocol needs only 1 ng DNA to start). This should accommodate DNA extracted from most habitats. In situations where only a very limited amount of DNA is available, amplification of the DNA might be needed to generate enough material for creating sequencing libraries. To maintain the relative abundance level between community members, strategies such as multiple displacement amplification can be used. Such amplification can generate more than enough DNA for library construction from femtograms of starting DNA.

13.2 Sequencing Approaches

There are several key factors that need to be considered before the sequencing process starts. These include sequencing depth, read length, and sequencing platforms. The depth of sequencing is dependent on the goal to be pursued. Studies that attempt to locate rare members of microbial communities require deeper sequencing than those that are only focused on more abundant members. With regard to read length, longer reads are always better than shorter reads in metagenomics for sorting out the inherent sequence complexity. The read length from the commonly used Illumina HiSeq system can currently read 125 bp from one end using the high-output mode, and can generate ~450 bp reads if using partially overlapped paired-end sequencing during the rapid run mode (see Chapter 10). As overviewed in Chapter 4, other technologies, such as Pacific Biosciences's SMRT (single molecule real-time) and the 454 pyrosequencing, generate longer (but fewer) reads. A hybrid approach is often used to take advantage of the different

strength of these technologies, with the use of shorter reads to generate an in-depth survey of the community and longer reads to provide scaffolding for assembling contigs (see Section 13.5.1). Future advancements in sequencing technologies will undoubtedly lead to continuous increase in read length and drop in cost making the goals of metagenomics more achievable.

13.3 Overview of Whole-Genome Shotgun (WGS) Metagenome Sequencing Data Analysis

For microbial community profiling, whole-metagenome shotgun sequencing has the benefit of being able to detect microbial genes without having to assemble entire genomes contained in the community first. When the microbial community is complex and most species have low coverage, sequence reads can be directly searched against currently known gene sequences to identify gene tags and analyzed for taxonomic composition and functional status. For less complex communities, reads can be assembled into contigs before conducting further analysis, and an increase in sequence length generally produces better results in subsequent taxonomic and functional analyses including pathway reconstruction, although the assembly process is not without challenges.

Figure 13.1 shows an overview of WGS metagenome data analysis, including the use of short reads directly for gene mapping and the alternative metagenome-assembly-based approach. For both approaches, subsequent sequence homology and other feature searches against currently cataloged genes in various public databases are the key steps. Although the results from these key steps are limited to the currently known sequences, the rapid increase in the number of sequenced microbial genomes will gradually alleviate this limitation. Besides taxonomic identification and functional profiling in one condition or habitat, comparative metagenomics analysis between conditions or habitats is usually performed to achieve the final goal of studying the effects of environmental factors on a microbial community. The following sections cover these various aspects of metagenomics data analysis. Because of the great diversity in sampling habitats/conditions and the specific questions asked in each study, there is no fixed workflow for metagenomics data analysis. Many of the steps outlined in Figure 13.1 and covered next are not necessarily arranged in the most appropriate order for a particular project, and they can be used in different combinations or with some step(s) omitted. Compared to other NGS applications, there is still a relative lack of tools for metagenome analysis. Some of the currently available tools, such as those required for contig assembly and sequence search against multiple databases, require considerable computing resources and power.

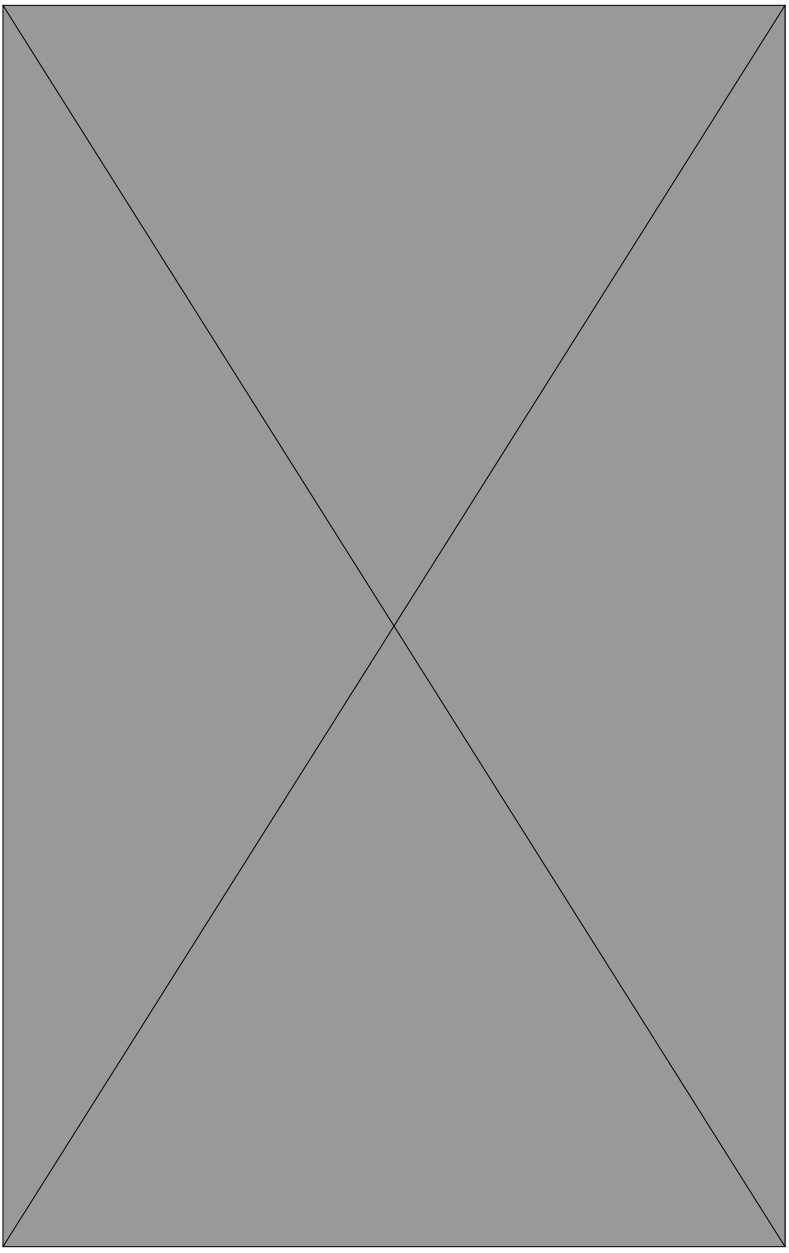

FIGURE 13.1
Major steps of metagenome analysis.

13.4 Sequencing Data Quality Control and Preprocessing

To ensure data quality and avoid erroneous results, metagenomic shotgun sequencing reads should be examined and preprocessed prior to conducting downstream analysis. Using the tools introduced in Chapter 5, reads of low quality should be filtered out, and low-quality bases and adapter sequences trimmed off. In addition, for samples from host-associated habitats, contaminating host sequences need to be marked and excluded from further analysis. Currently available tools for marking and removing DNA contamination sequences include BMTagger [376] and DeconSeq [377]. Additional data preprocessing also includes removal of duplicated reads. This can be conducted with tools such as the Picard module called EstimateLibraryComplexity, which identifies and removes duplicate reads without the need to align reads to a reference genome.

13.5 Taxonomic Characterization of a Microbial Community

13.5.1 Metagenome Assembly

Though the ultimate goal of metagenomics is to assemble each genome in a microbial community, this is currently still far from achievable for several reasons. The number of organisms in a metagenome is unknown, and there are wide variations in their relative abundance and therefore sequencing depth among the organisms. This is especially the case for samples collected from highly complex microbial communities. The large number of species in these samples and the concomitant low-sequencing depth for most species make metagenome assembly extremely challenging. Sequence similarity between closely related species poses further challenges to assemblers, often leading to chimeric assemblies that contain reads from different OTUs. Despite the challenges, metagenome reads assembly is an important step, especially for low-complexity samples. It enables discovery of novel genomes (e.g., the discovery of three novel viral genomes in Yellowstone lake [378]), discovery of novel genes (e.g., the first bacterial rhodopsin was discovered by metagenomics [379]), and characterization of long complex genomic elements (such as clustered regularly interspaced short palindromic repeats, or CRISPRs [380]).

For *de novo* metagenome assembly, the assemblers introduced in Chapter 10 for single-genome *de novo* assembly, such as SOAPdenovo and Velvet, were initially applied but with limited success. As a result, assemblers tailored for metagenome reads have been developed. For assembling longer reads such as those generated from Sanger or 454 technologies, assemblers like MAP [381],

Genovo [382], and Xgenovo [383] can be used. For the relatively short Illumina reads, more assemblers are currently available, including MetaVelvet/MetaVelvet-SL [384,385], meta-IDBA/IDBA-UD [386,387], GeneStitch [388], Ray Meta [389], and Omega [390]. Similar to single-genome assemblers, many of these short-read metagenome assemblers are based on the de Bruijn graph approach (see Chapter 10). The difference from the single-genome assemblers, though, is that they attempt to identify subgraphs within a mixed de Bruijn graph, each of which is expected to represent an individual genome. For example, MetaVelvet first builds a large mixed de Bruijn graph from metagenomic reads, which is then decomposed into individual subgraphs.

After the assembling process, a metagenome usually comprises mostly small contigs. To evaluate the assembly quality, traditional evaluation metrics, such as N50, are not as informative and representative as in evaluating single-genome assemblies. Instead, aggregate statistics such as the total number of contigs, and the maximum, median, and average length of the contigs are often used. Further inspection of the assembly quality includes looking for chimeric assemblies. While there are currently no tools available to detect chimeric assembly, the assemblies should nevertheless be checked by looking for signs of chimeric assembly, such as sudden changes in coverage, G/C content, and codon usage (different species have different codon usage patterns). The use of paired-end reads and a higher sequence match threshold helps reduce the rate of chimeric assemblies.

After contig assembly, if paired reads are available, metagenome scaffolds can be built from the contigs. Many of the metagenome assemblers have a module to carry out scaffolding. Besides these modules, dedicated metagenome scaffolding tools like Bambus 2 [277] may be used to determine if additional scaffolding is needed. Bambus 2 accepts contigs constructed with most assemblers using reads from all sequencing platforms. In the process of building scaffolds from contigs, ambiguous and inconsistent contigs may also be identified.

13.5.2 Sequence Binning

Metagenomic sequence binning refers to the process of grouping sequence fragments in the mixture and placing them into different "bins" corresponding to their taxonomic origins. This process can be conducted on both assembled and unassembled reads. With longer reads or contigs, high-resolution binning can be achieved at the levels of family or genus. Short sequences may be binned only to the level of phylum due to the limited information carried in the sequences. Since it reduces the complexity inherent in the metagenomics data, each set of binned sequences can also be subjected to independent analysis in other steps. For example, assembly can be performed postbinning on each binned sequence set to improve performance.

Three binning approaches are usually used: those based on sequence composition, homology, and fragment recruitment. Composition-based binning

assigns sequences into different taxonomic groups based on characteristics such as G/C content, oligonucleotide sequence frequency, and codon usage. This binning approach is based on the assumption that sequences from closely related species are more similar to each other in these characteristics than to distantly or nonrelated species. Binning methods that use this approach include PhyloPythia [391] and its successor PhyloPythiaS [392], TETRA [393], TACOA [394], Phymm [395], S-GSOM [396], and PCAHIER [397]. Unlike the other two approaches, these methods do not compare reads to a database of reference sequences, although some use reference sequences from different taxonomic groups to train their algorithms. For example, PhyloPythia uses the frequency of oligonucleotide sequence of variable length to assign metagenomic sequences to different clades, based on the support vector machine (SVM) model trained with taxonomically annotated reference sequences. Composition-based binning is more reliable for long and assembled reads, because short sequences carry less information due to their limited length. While the composition-based binning approach has the advantage of being fast, as it does not rely on aligning metagenomic sequences to references, variation in the distribution of composition characteristics can lead to inaccuracies. For those methods that use reference sequences to train their binning models, the selection of training sequences can also affect the results.

Homology-based binning is based on sequence similarity and the assumption that sequences from closely related species are more similar to each other than to unrelated species. This approach assigns metagenomic sequences to their taxonomic sources of origin by searching against a database of microbial sequences that are taxonomically annotated. Currently available methods based on this approach include MEGAN4/5 [398,399], CARMA and WebCARMA [400,401], SOrt-ITEMS [402], and MetaPhyler [403]. MEGAN, for example, conducts a BLAST search on metagenomic sequences using a database of NCBI (National Center for Biotechnology Information) sequences that have known taxonomic origins. Because of the tremendous amount of BLAST search involved, this process is computationally intensive and demanding on computing resources. As it is based on the current annotation of cataloged sequences, this approach is not suitable to find currently unknown species or taxa. Some methods, such as MetaCluster [404], PhymmBL [395], and SPHINX [405], take a hybrid strategy combining both the composition and the homology approaches.

The third approach, based on fragment recruitment, maps metagenomic reads to available microbial genomes in order to identify their sources of origin. This approach was first used by the Global Ocean Sampling Expedition to study the marine planktonic microbiota [406]. Although the mapping can be conducted with general-purpose mappers such as Bowtie or BWA, there are few algorithms, except Genometa [407] and FR-HIT [408], that are specifically designed for this approach. Since it is limited to assigning metagenomic sequences to species that have a reference genome, this approach

is not very well suited to study microbial communities that contain many unknown species.

13.5.3 Calling of Open Reading Frames (ORFs) and Other Genomic Elements from Metagenomic Sequences

To answer the questions of what taxonomic groups are in a microbial community and what they are doing, identification of genes and other genomic elements (such as noncoding RNA) from assembled contigs or unassembled reads is an essential step. For gene coding region identification, since metagenomic sequences containing open reading frames (ORFs) may not carry full-length ORFs, metagenome ORF calling algorithms do not penalize for their incompleteness. Many metagenome ORF callers employ machine-learning strategies such as hidden Markov models (HMMs) or artificial neural networks (ANNs). Examples of these callers include FragGeneScan (FGS) [409], MetaGeneMark and other programs in the GeneMark family such as GeneMark.hmm [410], MetaGeneAnnotator (MGA) [411], and Orphelia [412]. Identification of other genomic elements, such as ncRNAs and CRISPRs, may require long reads or contigs as well as more computational resources. A limited number of tools such as tRNA-SE [413] and CRISPRFinder [414] are currently available to identify these elements. Besides providing answers to the composition and function of a microbial community, calling of ORFs and other genomic elements also helps identify misassembled reads or locate adjoining contigs that are not yet placed into the same scaffold.

13.5.4 Phylogenetic Gene Marker Analysis

Phylogenetic gene markers are ubiquitous genes that are phylogenetically diverse and therefore can be used to determine the structure and composition of a microbial community. Examples of these marker genes are the rRNA genes (e.g., 16S), *recA* (DNA recombinase A), *rpoB* (RNA polymerase beta subunit), *fusA* (protein chain elongation factor), and *gyrB* (DNA gyrase subunit B). There are two general approaches to apply these gene markers to the determination of taxonomic groups in a community. The first is based on a sequence similarity search. Methods that use this approach include MetaPhlAn [415] and MetaPhyler. MetaPhlAn, for example, conducts a metagenomic sequence similarity search against an extensive list of clade-specific gene markers to determine taxonomic composition. The other approach uses the phylogenetic information embedded in gene marker sequences to infer phylogenetic trees from metagenomic reads via multiple sequence alignment. AMPHORA (also AMPHORA2 and AmphoraNet) [416–418], PhylOTU [419], and PhyloSift [420] are some examples of this approach. In the case of AMPHORA, HMMs are used to align metagenomic reads to multiple marker sequences. A phylogenetic tree is then inferred from the multiple-alignment results.

13.6 Functional Characterization of a Microbial Community

13.6.1 Gene Function Annotation

ORF calling from metagenomic sequences provides substrate for functional analysis of the underlying community, that is, answering the question of what they are doing. Functional annotation of called ORFs can reveal the full repertoire of genes (or their protein products) in a habitat, which usually perform a wide range of functions such as metabolism, signal transduction, stress tolerance, and virulence. Uncommon functions may suggest an unusual lifestyle and activity in a community. The relative abundance of different types of genes also reveals specificity about a community and how organisms in the community deal with environmental factors in the habitat.

To conduct functional annotation, predicted protein sequences from called ORFs are searched against a database of reference protein sequences, or HMMs that describe protein families. Protein sequence databases (such as COG/KOG, eggNOG, FIGfams, and UniRef) and HMM databases (such as Pfam and TIGRFAMs) are among the most commonly used databases. This task of database searching to identify all possible peptides coded by the metagenome is a computationally intensive process. If local computing resources permit, locally installed standalone tools such as RAAMCAP [421], SmashCommunity [422], and MetAMOS [423] can be used. Alternatively, the task can also be submitted to a web-based system such as the MG-RAST SEED system [424] or IMG/M.

This process of database searching at the ORF or gene level provides a functional snapshot of the community in terms of what functions are most possibly active. Although this functional annotation is based on metagenomic DNA instead of metatranscriptomic RNA analysis, comparison of metagenomic and metatranscriptomic data has found that the relative abundance levels of genes and their transcripts are usually well correlated in the same communities [425]. Therefore, the functional snapshot revealed by metagenomics data serves as an approximation of gene activity in the community.

13.6.2 Metabolic Pathway Reconstruction

To perform functional analysis at the metabolic pathway level, which offers a different layer of understanding of community activities, the same metagenomically predicted peptide sequences can be searched against KEGG Orthology and MetaCyc. Both databases allow mapping of the peptide sequences to different biological pathways. One of the currently available tools for metabolic pathway analysis is HUMAnN [426]. It employs MBLASTX to search metagenomic reads against the KEGG Orthology to determine the abundances of individual orthologous protein families. HUMAnN reconstructs pathways using MinPath [427], which is a maximum parsimony

approach to explain the observed families and their abundances with a minimal set of pathways. After further noise reduction and smoothing, the output from HUMAnN displays pathway coverage (i.e., whether each pathway is present or absent) and the relative abundance of each pathway in the metagenomic samples. MetaPath [428], another currently available tool, identifies differentially abundant metabolic subnetworks between metagenomic samples.

13.7 Comparative Metagenomic Analysis

Comparative metagenomic analysis between habitats or conditions can lead to insights about the underlying microbial communities and their dynamics. However, statistical comparison between metagenomes is not as straightforward as other NGS-based comparative analyses (such as RNA-Seq). This is mostly due to the tremendous amount of variability involved in comparative metagenomic analysis. One source of this variability is biological, as microbial composition can vary greatly between different samples. Another source is technical, due to insufficient sequencing depth and therefore undersampling of low-abundance species. These species generate fewer reads and are more affected by stochastic factors in the sequence sampling process, as in general the number of reads from a species is dependent on a number of factors, including relative abundance of the species, genome size, genome copy number, within-species heterogeneity, and DNA extraction efficiency. Due to these biological and technical factors, many species or OTUs detected in one sample or condition are often absent in another sample or condition. If rare species need to be studied in a metagenome study, it is more cost-effective to artificially increase their abundance using cell enrichment technologies such as flow cell sorting rather than increasing sequencing depth. In a typical metagenomics project that does not artificially increase the abundance of rare species, their undersampling can lead to significant biases in subsequent data normalization and detection of significant differences between samples. Compared to other steps in the metagenomic data analysis pipeline, there has been relatively less method development in comparative metagenomic analysis.

13.7.1 Metagenome Sequencing Data Normalization

Similar to RNA-Seq data, metagenomic abundance data needs be normalized prior to comparative analysis. Currently there is still no consensus as to how metagenomics data should be normalized. Among the normalization approaches that have been reported, total-sum scaling (TSS), equivalent to the total count approach in RNA-Seq (Chapter 7), is performed by

dividing the raw count of reads assigned to a certain species or OTU by the total number of reads in the same sample. Another approach is cumulative-sum scaling (CSS), which, similar to the upper quartile approach in RNA-Seq, is calculated by dividing the raw count of reads assigned to a species or OTU by the cumulative sum of counts up to a certain percentile. In a currently available study [429], CSS performed better than other normalization approaches including TSS.

13.7.2 Identification of Differentially Abundant Species or Operational Taxonomic Units (OTUs)

To identify species or OTUs that are differentially abundant between habitats or conditions, currently available tools include metagenomeSeq [429], LEfSe [430], METASTATS [431], STAMP [432], Xipe [433], MEGAN4/5, and MG-RAST. These tools use different methods and statistics to detect differential abundance between metagenomes. For example, metagenomeSeq implements the CSS normalization and a distribution mixture statistical model to deal with the biases caused by the undersampling issue that confounds comparative metagenomic analyses. LEfSe uses the Kruskal-Wallis rank-sum test to detect features that display significant differential abundance between conditions. Besides comparative abundance analysis, some of these tools, such as MEGAN4/5 and MG-RAST, can also be used to compare functional profiles between contrasting conditions in terms of Gene Ontology (GO) and KEGG pathways. Tools dedicated to the comparison of functional profiles between habitats or conditions are also available, such as ShotgunFunctionalizeR [425].

13.8 Integrated Metagenomics Data Analysis Pipelines

Besides the tools developed for each of the aforementioned individual steps, pipelines designed for integrated comprehensive analysis of metagenomics data are also available. These pipelines, including IMG/M, MEGAN4/5, MetAMOS, and MG-RAST, contain a large collection of tools that encompass the many aspects of metagenomics data mining including preprocessing, binning, feature identification, functional annotation, and cross-condition comparison. For example, MG-RAST directly takes sequencing and metadata files as input, conducts reads quality checks and preprocessing, gene calling, protein identification, annotation mapping, abundance profiling, comparative analysis, and metabolic reconstruction. Currently these pipelines require different input files. IMG/M prefers preassembled contigs, MEGAN4/5 requires reads BLAST search results against a database of reference sequences, and MetAMOS can take both sequence reads and preassembled contigs.

13.9 Metagenomics Data Repositories

In the United States, like for other NGS data, the NCBI SRA database provides the official repository for all metagenomic data collected by NGS technologies. In Europe, the EBI Metagenomics service offers archiving and analysis of metagenomics data. The data archived by the EBI Metagenomics service is also accessible through ENA-SRA. Besides these official metagenomics data repositories, MG-RAST and IMG/M are two *de facto* metagenomic data repositories that also enable data sharing in a collaborative environment and with the entire research community. The value of these repositories will become more apparent when more metagenomics data becomes available. For example, they can accelerate the discovery of new genes and species by providing opportunities to compare currently unknown sequences that exist in multiple metagenomes. In a typical WGS metagenomics study, many sequences are previously unknown and may represent novel genes or sequences from currently uncataloged species. To discover novel genes and new species, meta-analysis of data (including metadata) is needed, which is only enabled by these repositories.

Section IV

The Changing Landscape of Next-Generation Sequencing Technologies and Data Analysis

14

What Is Next for Next-Generation Sequencing (NGS)?

14.1 The Changing Landscape of Next-Generation Sequencing (NGS)

Massively parallel sequencing is a highly dynamic area of genomics. While the current technologies are still evolving to further improve performance, new technologies are constantly being developed. With more researchers adopting the next-generation sequencing (NGS) approach for transcriptomics, genotyping, *de novo* genome assembly, protein–DNA interaction analysis, epigenomics, and metagenomics, the drive for cheaper, faster, more accurate, and more sensitive sequencing technologies that generate longer reads will only become greater. With the power of NGS being proven in research labs, it has been gradually accepted in clinical settings to improve diagnosis, prognosis, and treatment of patients. On November 19, 2013, the U.S. Food and Drug Administration (FDA) for the first time approved the use of an NGS platform (the Illumina MiSeqDx system) for clinical use. The broadened use of NGS technologies in research and clinical settings has further accelerated the development of third- and future-generation sequencing technologies, including those based on the detection of electrical signals differentially induced by individual nucleotides.

The Oxford bio-nanopore technology, for instance, reads nucleotide sequences off a single-stranded DNA (or RNA) while it is threaded across a biological nanopore. The speed at which the DNA (or RNA) strand passes through the pore is critical for signal measurement and controlled by a processive enzyme located at the pore orifice. The raw sequencing signal from each pore is a trace of ionic current changes emitted from five-nucleotide DNA (or RNA) k-mers (not individual nucleotides) [434]. Deducing bases from the electrical signal trace is performed by the company's cloud-based software called Metrichor, which is based on hidden Markov models (HMMs). Another platform that is also based on the detection of electrical signals has been developed by the Japanese company Quantum Biosystems. Different from the Oxford technology, this platform conducts random DNA (or RNA)

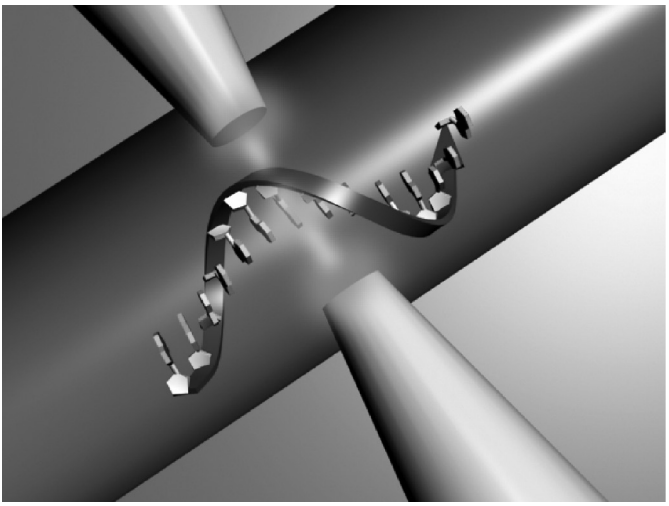

FIGURE 14.1
Third-generation single DNA/RNA molecule sequencing by measuring tunneling currents generated from the passing of a DNA/RNA molecule through a pair of nanoelectrodes with a subnanometer gap. (From T Ohshiro, K Matsubara, M Tsutsui, M Furuhashi, M Taniguchi, T Kawai, Single-molecule electrical random resequencing of DNA and RNA, *Scientific Reports* 2012, 2:501. With permission.)

single-base sequencing via measuring tunneling currents (Figure 14.1). These currents are produced when DNA (or RNA) molecules pass between pairs of nanoelectrodes that are separated by a gap of subnanometer scale [435]. Similar to the Oxford technology, the key to this technology generating a high-quality electrical signal for base calling is to control the speed of DNA (or RNA) molecule translocation through the gap, while at the same time confining the molecule's configuration during translocation. Specific statistical methods and algorithms are required for base calling from the generated tunneling currents.

While emerging sequencing technologies as exemplified by the Oxford and Quantum systems have to overcome technical including computational hurdles before becoming widely adopted, some characteristics of upcoming DNA (or RNA) sequencing technologies seem to be clear. Such characteristics include

- Single DNA (or RNA) molecule sequencing, that is, the ability to directly read individual target DNA molecules without relying on polymerase chain reaction (PCR) amplification or conversion to cDNA in the case of RNA
- Much improved read length

- Smaller equipment footprint and increased portability; for example, the Oxford MinION sequencing system is a miniaturized device that directly works through a computer's USB port
- Further drop in sequencing costs

The increased sensitivity that leads to the achievement of single DNA (or RNA) molecule sequencing makes it possible to directly sequence the genome or transcriptome of a single cell without any amplification. The reduced equipment size and increased affordability makes high-throughput sequencing more accessible to individual research and clinical labs, instead of being mostly limited to large genome centers or core facilities. The change in read length, and other aspects of sequence read output such as error model, also drives further evolution of bioinformatic tools.

14.2 Rapid Evolution and Growth of Bioinformatics Tools for High-Throughput Sequencing Data Analysis

The increased read length will undoubtedly improve the efficiency of bioinformatic tools for sequence mapping and assembly. With gradually improving chemistry since the introduction of NGS, we have already seen progressively increasing read lengths from the currently available platforms. Significantly longer reads associated with third- or future-generation sequencing technologies, as well as new techniques developed on the basis of currently available technologies (e.g., the Moleculo long-read technology acquired by Illumina), will not only improve *de novo* genome assembly but also all the other applications that depend on mapping to a reference genome. For example, increased read length in RNA-Seq can lead to recognition of different transcripts that are produced from the same gene, and therefore facilitate studies of alternative splicing. As higher read length increases sequence information content and uniqueness, which in turn leads to increased "assemblability" or "mappability," newer alignment algorithms or updated versions of existing ones are surely to be developed to harness the power afforded by this increase in read length. For example, long-read *de novo* genome assemblers, such as HGAP [436] and FALCON [437], have been developed more recently to assemble long reads generated from platforms such as the Pacific Biosciences system. BWA-MEM has also been added recently to the widely used BWA alignment package to accommodate longer reads, and it generates better performance on these reads than previous versions designed for shorter reads.

The adaptation of algorithms to the increase in read length is only one example of the impact of sequencing technology advancements on the evolution

of bioinformatic tools. Besides assembly and alignment, algorithms and tools for other steps or applications, including base calling, variant calling, transcriptomic analysis, ChIP-Seq peak calling, DNA methylation sequencing, and metagenome characterization, are also under constant development. While new ones are continuously being introduced, many existing algorithms and tools are also under constant revision. As base calling is highly platform-dependent, base callers are usually developed as part of the sequencing platform development process. Although there are also third-party base callers being developed in an attempt to further improve performance, efforts on algorithmic and software tool development are mostly focused on more downstream analyses.

As an illustration of the dynamic nature of these efforts, RNA-Seq analysis algorithmic development and utilization have seen continuous growth since 2010. Figure 14.2 shows the total number of reports related to RNA-Seq data analysis algorithms published each year between 2010 and 2014. While these numbers do not directly measure the total number of new or updated RNA-Seq algorithms, they do to a large degree reflect the amount of algorithmic development efforts as well as the demand in this direction. Algorithmic development and application in other steps/applications show the same trend. Because of the constant introduction and improvement of bioinformatic tools, researchers might find it necessary at times to rerun previously performed analyses using newer tools.

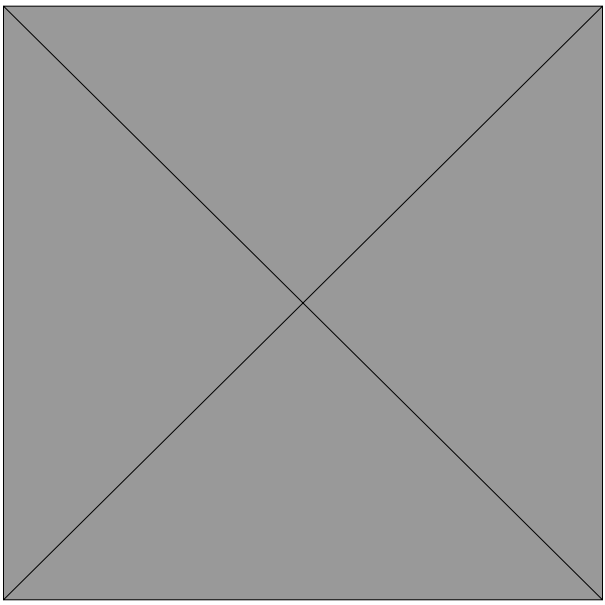

FIGURE 14.2
The increase in the number of publications from 2010 to 2014 related to the development and application of RNA-Seq algorithms. (From Google Scholar.)

14.3 Standardization and Streamlining of NGS Analytic Pipelines

While the active development of algorithms and the wide array of bioinformatic tools becoming available may seem to make it more difficult to test and choose the right tools, there are also more efforts focusing on standardizing and streamlining bioinformatic workflows for the different NGS applications. Commercial packages, such as CLC Genomics Workbench and GeneSpring NGS, tend to incorporate different modules into one suite to cover most commonly used NGS applications. Whereas packages developed out of academic settings tend to be more specialized and as a result a bit fragmented, there have also been efforts on the unification of different components into frameworks, such as the GATK for variant calling [438]. The clinical use of NGS on diagnosis, genetic risk assessment, and patient management further demands the standardization and streamlining of NGS data analytic workflow, which has led to the deployment of pipelines such as Mercury [439] and Rainbow [440]. To handle the vast volumes of NGS data effectively, many of these pipelines take advantage of high-performance parallel computing, and increasingly with the use of cloud technologies.

14.4 Parallel Computing

Parallelization, a computation term that describes splitting of a task into a number of independent subtasks, can significantly increase the processing speed of highly parallelizable tasks, which include many NGS data analysis steps. For example, although millions of reads are generated from a sequencing run, mapping of these reads to a reference genome is a process that is "embarrassingly parallel," as each read is mapped independently to the reference. As parallel computing can be efficiently carried out by graphics processing units (GPUs) since rendering of each pixel on a computer screen is also a highly parallel process, the integration of GPUs with CPUs in heterogeneous computing systems can increase throughput ten- to hundredfold, and turn individual computers into mini-supercomputers. While these systems can be applied to various aspects of NGS data analysis, many NGS analytical tools have yet to take full advantage of the power of parallel computing in such systems.

Parallelization is also an important factor in determining how an increase in the number of CPU (or GPU) cores might affect actual NGS data processing performance. If a step is highly parallelizable, and the algorithm designed for it employs parallelization, then an increase in core number will most likely lead to improved performance. On the contrary, if the step is not

readily parallelizable, or even if the task is parallelizable but the algorithm deployed does not use parallelization, simply having more cores may not lead to improvement in performance.

14.5 Cloud Computing

Because the rates at which NGS technologies advance and sequencing costs drop are faster than those of development in the computer hardware industry and the resultant increase in computing power (i.e., Moore's law), the gap between NGS data generation and their computational analysis will only widen. To narrow this gap and speed up NGS data processing, the NGS community has begun to embrace a trend that has been taking place in computing resource distribution from the long-existing model of local computing to cloud computing. Companies such as Amazon, Microsoft, and Google have been building megascale cloud computing clusters and data storage systems for end users to use over the Internet. Compared to local computing, cloud computing enables access to supercomputing and mass data storage capabilities without the need to build and maintain a local workstation, server, or high-performance computing cluster.

At the core of cloud computing is virtualization technology, which allows an end user to create a virtual computer system on demand with the flexibility of specifying the number of CPU cores, memory size, disk space, and operating system that are required for a job. With this technology, multiple virtual computer systems can be run simultaneously on the same physical cloud server. The adoption of cloud computing for NGS data processing has demonstrated the advantages of this "supercomputing-on-demand" model, which include flexibility, scalability, and oftentimes cost-savings. The flexibility and scalability offered by cloud computing allow a researcher to conduct NGS data analysis using supercomputing capabilities that previously only existed in large genome centers. Cost savings are achieved as the user only needs to pay for the time used by the user-configured computing instance.

Another advantage of using the cloud is with data sharing among researchers and projects. By providing single, centralized data storage, the cloud enables different groups located in different geographical locations to have access to the same data sets and share analytical results. Furthermore, with cloud computing, the task of bringing software tools to the "big" NGS data can be more readily realized. In contrast to the large sizes of NGS data files, the software and scripts designed to process them are much smaller. Therefore, it is much easier and more efficient to download and install them to wherever the data is stored, rather than moving or replicating the high volumes of NGS data to where the tools are installed. By directly storing production data in the cloud, the burden of data transfer is greatly reduced;

by coupling data and tools in the same place, optimal performance can be achieved.

While cloud computing enables users to offload the hassle and cost of running and maintaining a local computing system, it does have downsides that need to be considered. One of the practical barriers of moving to the cloud is the speed of data transfer into and out of the cloud. It may take a week to upload 100 GB of data to the cloud using low-speed Internet connections. The question of whether to run analysis in the cloud is heavily dependent on the amount of data to be transferred and the computational complexity of the analytical steps. As a general rule, it is only worthwhile to upload data to the cloud for processing when the analytical task requires more than 10^5 CPU cycles per byte of data [441]. So for projects that deal with large amounts of data but do not involve a lot of highly intensive computational steps, more time may be spent on data transfer to the cloud rather than data processing. Other potential factors include data security, cost ineffectiveness under some circumstances, availability of analytical tools in the cloud environment, and network downtime. Although users can access their data from anywhere on the Internet, the convenience also means the possibility of data security being breached or compromised. Some heavy users may find cloud computing not as cost-effective as running a local server. While more tools are becoming available in the cloud, users still need to use due diligence to make sure that the tools they need are available. For users at places that suffer frequent network outages, cloud computing can be problematic as all cloud-based operations are dependent on Internet traffic.

Despite the potential downsides, cloud computing has been proven to be a viable approach for NGS data analysis. Table 14.1 is a list of some of the current cloud-computing providers that can be used for NGS applications. To illustrate how cloud computing can be deployed for analyzing NGS data, following is an example on the conduct of reads alignment using the Amazon Elastic Compute Cloud (Amazon EC2). As the first step, input data files (FASTQ files and a reference genome file) are uploaded from a local computer to a "bucket" in the Amazon S3 cloud storage. This bucket, which is also used to hold program scripts and output files, can be created with the

TABLE 14.1

Providers of Cloud Computing That Can Be Used for NGS Data Analysis

Provider	URL
Amazon Elastic Compute Cloud	http://aws.amazon.com/ec2/
Rackspace	http://www.rackspace.com
Bionimbus	http://bionimbus.opensciencedatacloud.org
Open Cloud Consortium (not-for-profit)	http://opencloudconsortium.org
Microsoft Azure	http://azure.microsoft.com/
Google Cloud	https://cloud.google.com

AWS (Amazon Web Services) Management Console, a unified interface to access all Amazon cloud resources. To initiate alignment, a workflow must be defined first using the Console's "create workflow" function. To define the workflow, the input sequence read files, the aligner script, and the saving location for alignment output files are specified. In the meantime, the number of Amazon EC2 instances required for the job, which determines memory and processor allocation, is also configured. After the configuration, the job is submitted through the Management Console. When the instances are finished, alignment output files are deposited into the prespecified file location in the S3 cloud storage.

Appendix A: Common File Types Used in Next-Generation Sequencing (NGS) Data Analysis

BAM: A file format for storing reads alignment data. It is the binary version of the SAM format (see *SAM*). Compared to its equivalent SAM file, a BAM file is considerably smaller in size and much faster to load. Unlike SAM files, however, the BAM format is not human-readable. BAM files have a file extension of .bam. Some tools require BAM files to be indexed. Besides the .bam file, an indexed BAM file also has a companion index file of the same name but with a different file extension (.bai).

BCF: Binary VCF (see *VCF*). While it is equivalent to VCF, BCF is much smaller in file size due to compression, and therefore achieves high efficiency in file transfer and parsing.

BCL: Binary base call files generated from Illumina's proprietary base calling process.

BED: Browser Extensible Display format used to describe genes or other genomic features in a genome browser. It is a tab-delimited text format that defines how genes or genomic features are displayed as an annotation track in a genome browser such as the UCSC Genome Browser. Each entry line contains three mandatory fields (chrom, chromStart, and chromEnd, specifying for each genomic feature the particular chromosome it is located on and the start and end coordinates) and nine optional fields. Binary PED files (see *PED*) are also referred to as BED files, but this is a totally different file format.

bedGraph: Similar to the BED format, bedGraph provides descriptions of genomic features for their display in a genome browser. Distinctively the bedGraph format allows display of continuous values, such as probability scores or coverage depth, in a genome.

bigBed: A format similar to BED, but bigBed files are binary, compressed, and indexed. Display of bigBed files in a genome browser is significantly faster due to the compression and indexing, which allow transmittal of only the part of the file that is needed for the current view instead of the entire file.

bigWig: A format for visualization of dense, continuous data, such as GC content, in a genome browser. A newer format than the WIG format

(see *WIG*), bigWig is a compressed and indexed binary file format and loads significantly faster.

FASTA: A text-based format for storing sequences. A sequence stored in the FASTA format contains only two elements: a single-line description (or defline) and the sequence text. The defline starts with the ">" symbol, followed by a sequence identifier, and then a short description. The sequence text is usually divided into multiple lines with each less than 80 characters in length. This format has its origin in the FASTA program package developed in the late 1980s. Multiple sequences can be stored in a FASTA file. FASTA files often have file extensions of .fa, .fasta, or .fsa.

FASTQ: The current *de facto* standard for storing sequencing data generated from various NGS systems. It is a compact text-based format containing nucleotide base sequences and their call quality scores. Each read sequence in a FASTQ file is represented by four lines of information. The first line starts with the symbol "@," followed by the sequence ID and descriptor. The second line is the read sequence. Line 3 starts with the "+" symbol, which may be followed by the sequence ID and description (optional). Line 4 lists base-call quality scores for each base in the read sequence. This format was originally developed by the Sanger Institute. FASTQ files have file extensions of .fq or .fastq. Compressed FASTQ files also have the suffix .gz or .gzip from the compression utility used to create them.

GFF: General (or Generic) Feature Format. GFF is a tab-delimited text file format that describes how genes or other genomic features are displayed in a genome browser. There are different versions of this format, and GFF2 and GFF3 are currently the two major versions in use. The GFF format can be converted to the BED format (see *BED*).

GTF: Gene Transfer Format. A refined GFF format. Identical to GFF2.

PED: A file format used by PLINK (a toolset for genome-wide association analysis) that contains pedigree/phenotype data.

SAM: Standing for Sequence Alignment/Map, SAM is a standard NGS reads alignment file format that describes how reads are mapped to a reference genome. It is a tab-delimited text format and human-readable. SAM files can be converted into a compressed binary version (BAM) for faster parsing and file size reduction. SAM files have a file extension of .sam. An indexed SAM file also has an accompanying index file that has the file extension of .sai.

SFF (Standard Flowgram Format): A type of binary sequencing file generated by 454 sequencers. Can be converted to the FASTQ format using utilities such as sff2fastq.

VCF: Stands for Variant Call Format. A commonly used file format for storing variant calls. It is a tab-delimited, human-readable text format that contains metainformation lines, a header line, and data lines that describe each variant.

WIG: Wiggle Track Format. It is used for displaying continuous data tracks, such as GC content, in a genome viewer such as the UCSC Genome Browser. The WIG format is similar to the bedGraph format (see *bedGraph*), but a major difference between the two is that data exported from a WIG track is not as well preserved as that from a bedGraph track. The WIG format can be converted to bigWig (see *bigWig*) for improved performance.

Appendix B: Glossary

5-methylcytosine (5-mC): The most frequently observed form of epigenetic DNA modification. Produced by the addition of a methyl group to the fifth carbon of cytosine. Cytosine methylation reduces gene transcription and regulates chromatin remodeling.

algorithm: A well-defined procedure that comprises a set of instructions for solving a recurring problem.

alignment: Similarity-based arrangement of sequences. In next-generation sequencing data analysis, sequence reads are usually aligned against a reference genome to locate their genomic origins.

allele: One particular variant form of a gene that has a number of alternative sequence variants.

annotation: The process of providing biologically relevant information to a piece of DNA or RNA sequence. Also refers to the biological information itself that is attached to a sequence.

ASCII: Standing for American Standard Code for Information Interchange, ASCII provides a standard for encoding characters. Since a computer only deals with numbers, each human-readable character has to be encoded with a unique number in a computer. An ASCII code is the numerical representation of a character in a computer. For example, in the ASCII table, the character "A" is represented by the number 65.

assembly: A computational process to reconstruct a longer sequence from short sequences.

barcode: Unique short artificial sequence(s) attached to DNA molecules in a sequencing sample. The use of barcode sequences enables identification of different samples when they are sequenced together in a mixture (i.e., multiplex sequencing). Also see *multiplex sequencing* and *demultiplexing*.

base-call quality score: A score assigned to each base call in a sequence read to quantify the confidence level of making the call. In next-generation sequencing, it is defined in the same way as the Phred quality score originally developed for Sanger sequencing. Also see *Phred quality score*.

bisulfite conversion: A chemical process that leads to the differentiation of methylated cytosines from unmethylated cytosines. The treatment by bisulfite converts unmethylated cytosines in DNA to uracil, while methylated cytosines are not affected by this process. Bisulfite conversion coupled with next-generation sequencing is a major means to study genome-wide DNA methylation. Also see *whole-genome bisulfite sequencing*.

Burrows-Wheeler transform (BWT): A method of permuting the characters of one string into another string. In next-generation sequencing data analysis, BWT enables fast reference genome searching by providing efficient compression and indexing.

cDNA: Complementary DNA. Refers to DNA that is reversely transcribed from and therefore complementary to an mRNA species.

CDS (coding DNA sequence): The region of DNA that is translated into protein.

ChIP-Seq: Chromatin immunoprecipitation coupled with sequencing. A major application of next-generation sequencing for studying genome binding of DNA-interacting proteins such as transcription factors.

codon: A trinucleotide sequence of DNA or RNA that codes for a specific amino acid or the signal for protein synthesis termination. There are a total of 64 codons, with 61 specifying amino acids and 3 as termination signals.

contig: A contiguous segment of RNA or DNA sequence resulting from assembly of a set of overlapping sequence reads.

copy number variation (CNV): One type of genomic variation caused by changes in copy number of a DNA segment, usually as a result of deletion or duplication. CNV is a subcategory of structural variation and involves DNA segments that are usually larger than 1 Kb. Also see *structural variation*.

coverage: The average number of times that nucleotides in different genomic positions appear in a sequencing data set. Also known as sequencing depth or simply depth.

demultiplexing: The identification and separation of sequencing reads that are generated from different samples, based on the unique barcode sequence(s) they carry, after a multiplex sequencing run. Also see *barcode* and *multiplex sequencing*.

depth: See *coverage*.

DNA polymerase: A class of enzyme that catalyzes the synthesis of a new DNA strand from free nucleotides, using an existing DNA strand as template. Many molecular techniques, including polymerase chain reaction and sequencing-by-synthesis, are based on the use of DNA polymerases.

DNase: An enzyme that catalyzes the hydrolysis of DNA into oligonucleotides or nucleotides.

epigenome: Refers to chemical modifications to DNA and histones, which provides additional regulation to genomic activity.

exome: The complete set of exons in an organism's genome.

exon: A stretch of nucleotide sequence that is part of a gene providing coding information for protein synthesis. Exons are transcribed to and usually retained in mRNA.

false discovery rate (FDR): A measure of statistical significance after correcting for multiple testing. It estimates the proportion of false discoveries in the final list of findings. Among the various approaches for multiple testing correction, FDR estimation offers a balance between statistical stringency and rate of type II errors and therefore is widely used for high-throughput genomics data analysis. Also see *multiple testing correction*.

GC content: The percentage of guanines plus cytosines in a DNA/RNA sequence or genome.

gene expression: The process by which the information encoded in a gene's nucleotide sequence is used to direct the synthesis of a functional gene product. The level of gene expression in a cell or population of cells is represented by the abundance of its product. The composition of the large number of gene products and their expression levels in a cell or population of cells constitute the gene expression profile of the host cell(s).

Gene Ontology (GO): An initiative to provide consistent description of gene products using standardized vocabulary. Each gene product is described by three structured ontologies that encompass their associated biological processes, cellular components, and molecular functions.

genome: The complete set of DNA sequence in a cell or an organism. Contains the complement of information needed to form and maintain the cell or organism. Including both protein-coding and noncoding sequences.

genotype posterior probability: The probability of a genotype given an observed data set, calculated from next-generation sequencing reads and often with the use of prior genotype information.

hidden Markov model (HMM): Name after the Russian mathematician Andrei Markov (1856–1922), HMM is a commonly used machine learning and data mining approach for signal processing and pattern recognition. A Markov model is a statistical model that deals with observed sequences and state transitions. In bioinformatics, HMM is often used for base calling, sequence alignment, and gene prediction.

high-performance computing (HPC): A computer system that has the capability to perform over one teraflop (10^{12}) floating-point operations per second by the use of parallel processing.

indel: A generic term for either the insertion or deletion of nucleotide(s) in a DNA sequence. Such insertion/deletion events lead to DNA mutation and sequence length change.

indexing: The process of creating a data structure for fast search. Techniques of indexing for sequence alignment include hashing (storing information on where a particular subsequence can be found in a reference genome or a large collection of reads), suffix array (that consists

of lexicographically sorted genomic DNA sequence suffixes), and Burrows-Wheeler transform (permutation of a genome based on suffix array).

irreproducible discovery rate (IDR): A measure of experimental reproducibility. Developed to evaluate the reproducibility between replicates of a ChIP-Seq experiment, it calculates the rate of irreproducible discoveries, that is, peaks that are called in one replicate but not in another.

k-mer: In genome assembly or sequence alignment, k-mer refers to all the possible subsequences of length k in a sequence read.

library: Collection of many different DNA (or RNA) fragments that are systematically modified for target DNA screening or high-throughput analysis (including next-generation sequencing). Specifically, a sequencing library is a pool of DNA (or RNA) fragments with universal adapters attached to their ends. To construct a sequencing library, DNA (or RNA) molecules extracted from a population of cells are usually randomly fragmented, followed by addition of universal adapters to the two ends of the fragments. Sequences in the adapters enable subsequent enrichment and high-throughput sequencing of the fragments.

long noncoding RNA (lncRNA): Non-protein-coding RNA species that are over 200 nucleotides in length; compare to small RNAs.

machine learning: A branch of computer science that focuses on developing software algorithms that provide computers the capability to learn and make predictions on new data. Machine learning is built on computational model construction from existing input data, which is then applied to new data for generating predictions or decisions.

mapping: The process of searching the sequence of a read against a reference genome sequence to locate its origin in the genome. Also see *alignment*.

mapping quality: An estimation of the probability of misaligning a read to a reference genome. It is reported as a Phred-scale quality score. Also see *Phred quality score.*

mate-pair reads: Reads generated from two ends of a long DNA fragment. To achieve sequencing of the two ends, the long DNA fragment is first circularized and then fragmented. Paired-end sequencing of the fragment that contains the junction of the two ends generates mate-pair reads.

MeDIP: Methylated DNA immunoprecipitation with anti-5-methylcytosine antibody.

metagenome: The collection of all the genomes contained in a microbial community that consists of many individual organisms.

metagenomics: Studies of all the genomes existing in a microbial community as a whole without the need to capture or amplify individual genomes. Also referred to as environmental or community genomics.

microarray: A high-throughput genomics technology based on the use of predesigned detection probes that are printed or synthesized on a solid surface, such as glass or a silicon chip, in a high-density array format.

minor allele frequency (MAF): Frequency of the least abundant allelic variant in a population.

miRNA: MicroRNA. See *small RNA*.

mRNA: Messenger RNA that carries protein-coding information from DNA for protein translation. It acts as the intermediate between DNA and protein. An important component of a transcriptome.

multiple testing correction: Adjustment of statistical confidence based on the number of tests performed. Multiple testing without such an adjustment leads to high levels of false positives. For example, at a *p*-value of 0.05, performing 100 comparisons simultaneously will generate 5 positive outcomes simply by chance if a correction is not applied. Commonly applied multiple testing correction approaches include the Bonferroni adjustment (conservative) and false discovery rate estimation. Also see *false discovery rate*.

multiplex sequencing: Simultaneous sequencing of multiple samples together. The use of artificial barcode sequence(s) enables sample identification. Also see *barcode* and *demultiplexing*.

multireads: Reads that map to multiple genomic locations.

N50: The weighted mean contig size of a genome assembly. To calculate N50, all contigs are first ranked based on their lengths, which is then followed by adding the ranked lengths from the top downward. N50 refers to the length of the contig that makes the total added length equal to or greater than 50% of the assembly size. An often-used metric of *de novo* genome assembly quality.

NAS: Network attached storage. Specialized computer data storage server providing data access to a variety of clients through network.

noncoding RNA: RNA species that carry out functions other than coding for proteins. Examples include small RNAs and lncRNAs. Also see *small RNA* and *long noncoding RNA*.

normalization: A mathematical procedure to correct for unwanted effects of unintended factors and/or technical bias (such as differences in sequencing depth between samples in RNA-Seq). This procedure puts focus on the biological difference of interest and makes samples in different conditions comparable.

normalized strand correlation (NSC): A measure of signal-to-noise ratio in ChIP-Seq. It is calculated as the normalized ratio between the maximum strand cross-correlation (at the fragment-length peak) and the background cross-correlation. Also see *relative strand correlation (RSC)*.

open reading frame (ORF): A continuous segment of DNA containing nucleotide triplet codons that starts with the start codon (ATG) and ends with one of the stop codons (TAA, TAG, or TGA).

operational taxonomic unit (OTU): A common microbial diversity unit used in metagenomics that may represent a species or a group of species. OTUs are clustered together based on DNA sequence information alone.

paired-end reads: Reads obtained from the two ends of a DNA fragment. Since the length of the DNA fragment, that is, the distance between the reads, is known, use of paired-end reads provides additional positional information in mapping or assembly of the reads. Compare to *single-end reads*.

pathway: A succession of molecular events that leads to a cellular response or product. Each event is usually carried out by a gene product. Many biological pathways are involved in metabolism, signal transduction, and gene expression regulation.

PCR bottleneck coefficient (PBC): An index of sequencing library complexity. It is calculated after the read mapping step as the ratio between the number of genome locations to which only one unique sequence read maps and the total number of genome locations to which one or more unique reads maps. PBC measures the distribution of read counts toward one read per location.

Phred quality score (Q score): An integer value that is used to estimate the probability of making an error, that is, calling a base incorrectly. It is calculated as $Q = -10 \times \log(10)P(\text{Err})$. For example, a Q score of 20 (Q20) means a 1/100 chance of making a wrong call. Q30 represents a 1/1000 chance of making a wrong call, which is considered to be a high-confidence score. Q scores are often represented as ASCII characters for brevity.

Picard: A set of tools written in Java for handling next-generation sequencing data and file formats.

Pileup: A file format created with SAMtools showing how each genomic coordinate is covered by reference sequence-matching or -unmatching bases from all aligned reads.

piRNA: Piwi-interacting RNA. See *small RNA*.

polymerase chain reaction (PCR): A molecular biology technique that amplifies the amount of a DNA or RNA fragment, with the use of specific oligonucleotide primers that flank the two ends of the target fragment.

promoter: DNA sequence upstream of the open reading frame of a gene. The promoter region is recognized by RNA polymerase during initiation of transcription. Contains highly conserved sequence motifs.

proteome: The complete set of proteins in a cell, tissue, or organ at a certain point of time. Proteomics analyzes a proteome via identifying individual component proteins in the repertoire and their abundance.

quality score: See base-call quality score.

read: Sequence readout of a DNA (or RNA) fragment.

reduced representation of bisulfite sequencing (RRBS): An experimental approach based on next-generation sequencing that determines the DNA methylation pattern in a reduced genome (usually to save costs). The reduced representation of the genome is usually achieved by the use of restriction enzymes.

relative strand correlation (RSC): A metric of signal-to-noise ratio in ChIP-Seq. RSC is the ratio between background-adjusted cross-correlation coefficient at the fragment-length peak and that at the read-length peak. Also see *normalized strand correlation* (NSC).

RNA-Seq: Stands for RNA sequencing. Also referred to as whole transcriptome shotgun sequencing. RNA-Seq is a major technology for transcriptome analysis and a major application of next-generation sequencing.

RNAi: RNA interference, that is, inhibition of gene expression. RNAi is usually mediated by small RNAs, which lead to degradation of specific mRNA targets.

RNase: An enzyme that catalyzes the degradation of RNA molecules.

rRNA: Ribosomal RNA, that is, RNA species that are essential components of the ribosome. They play key roles in protein synthesis. By quantity, they are the most abundant RNA species in a cell.

SAN: Storage area network. A type of local area network (LAN) designed to handle large data transfers.

Sanger sequencing: The first widely adopted DNA sequencing technology. Devised by Dr. Fred Sanger, it is based on the principle of sequencing-by-synthesis with the use of dideoxynucleotides that irreversibly terminate new DNA strand synthesis once incorporated. With the advent of next-generation sequencing technologies, this sequencing method has become the synonym of first-generation sequencing.

scaffold: Ordered arrangement of *de novo* assembled contigs. The relative positional relationships between contigs are inferred by mate-pair or paired-end reads. In a scaffold, while the order of contigs is known, sequence gaps still exist between contigs.

sequencing depth: See *coverage*.

sequencing library: See *library*.

single-end read: Sequence read generated from one end of a DNA fragment. This is in comparison with paired reads generated from both ends of a DNA fragment. Also see *paired-end reads*.

single nucleotide polymorphism (SNP): DNA sequence polymorphism due to variation at a single nucleotide position. Different from the term single nucleotide variation (SNV), SNP only refers to SNV that is relatively common in a population with frequency reaching a certain threshold (usually 1%). Also see *single nucleotide variation*.

single nucleotide variation (SNV): DNA sequence variation that involves change at a single nucleotide position, for example, the sequence change from ATTGCA to ATCGCA.

siRNA: Small interfering RNA. See *small RNA*.

small RNA: Also called small noncoding RNA. The major categories of small RNA are miRNA, siRNA, and piRNA. In comparison to mRNA molecules, these RNA molecules are much smaller in size. Small RNA play important regulatory roles in cells through mediating RNAi. Also see *RNAi*.

splicing: The process of removing introns from primary RNA transcripts and joining of exons to form mature mRNAs. Splicing can be conducted in more than one way for many genes, and this alternative splicing can lead to the production of different mRNA species from the same gene through retaining different combinations of exons (or even introns sometimes).

SRA: Sequence Read Archive (also called Short Read Archive) maintained by the National Center for Biotechnology Information (NCBI). SRA is one of the major archives of next-generation sequencing data generated worldwide. Other publicly available next-generation sequencing data archives include the European Nucleotide Archive (ENA) maintained by the European Bioinformatics Institute (EBI).

strand cross-correlation: In ChIP-Seq, there is a shift in base position between reads generated from the forward and reverse strands of DNA. Strand cross-correlation is a measure of this shift, and calculated as the Pearson correlation coefficient between the forward and reverse read counts at each base position when the reads on the two strands are shifted toward and away from each other at different base shifts. Also see *normalized strand correlation* and *relative strand correlation*.

structural variation (SV): Large-scale genomic change that include large indel, inversion, translocation, or copy number variation. Different from SNPs or small indels, SVs involve DNA segments that are usually larger than 1 Kb. Also see *copy number variation*.

transcript: An RNA molecule transcribed from a segment of DNA.

transcription start site (TSS): The nucleotide site in a segment of DNA from which RNA transcription is initiated.

transcriptome: The complete set of RNA transcripts in a cell, tissue, or organ at a certain point of time.

transcriptomics: Studies of the composition of a transcriptome. Encompasses identification of the large number of RNA species in a transcriptome and determination of their abundance levels. Major transcriptomics technologies include microarray and RNA-Seq.

translation: The process of protein synthesis from mRNA. Carried out by ribosomes.

tRNA: Transfer RNA. The function of tRNAs is to transfer amino acids to ribosomes for protein synthesis according to the triplet genetic code.

UTR: Untranslated region of an mRNA molecule. Can be located on either the 5′ end or the 3′ end of the mRNA molecule.

variant calling: Identification of sequence difference at specific positions of an individual genome (or transcriptome) in comparison with a reference genome. Each called variant usually has a corresponding Phred-scale quality score.

whole-genome bisulfite sequencing (WGBS): An application of next-generation sequencing that determines DNA methylation pattern across the entire genome using bisulfite conversion. Also see *bisulfite conversion*.

References

1. Vale RD. The molecular motor toolbox for intracellular transport. *Cell* 2003, 112:467–480.
2. Cavalier-Smith T. The simultaneous symbiotic origin of mitochondria, chloroplasts, and microbodies. *Ann NY Acad Sci* 1987, 503:55–71.
3. Lopez de Heredia M, Jansen RP. mRNA localization and the cytoskeleton. *Curr Opin Cell Biol* 2004, 16:80–85.
4. Hirokawa N. mRNA transport in dendrites: RNA granules, motors, and tracks. *J Neurosci* 2006, 26:7139–7142.
5. Mayer F. Cytoskeletons in prokaryotes. *Cell Biol Int* 2003, 27:429–438.
6. Bender A, Krishnan KJ, Morris CM, Taylor GA, Reeve AK, Perry RH, Jaros E et al. High levels of mitochondrial DNA deletions in substantia nigra neurons in aging and Parkinson disease. *Nat Genet* 2006, 38:515–517.
7. Corral-Debrinski M, Shoffner JM, Lott MT, Wallace DC. Association of mitochondrial DNA damage with aging and coronary atherosclerotic heart disease. *Mutat Res* 1992, 275:169–180.
8. Santos RX, Correia SC, Zhu X, Smith MA, Moreira PI, Castellani RJ, Nunomura A, Perry G. Mitochondrial DNA oxidative damage and repair in aging and Alzheimer's disease. *Antioxid Redox Signal* 2013, 18:2444–2457.
9. Greaves LC, Reeve AK, Taylor RW, Turnbull DM. Mitochondrial DNA and disease. *J Pathol* 2012, 226:274–286.
10. Green BR. Chloroplast genomes of photosynthetic eukaryotes. *Plant J* 2011, 66:34–44.
11. Harris SA, Ingram R. Chloroplast DNA and biosystematics: The effects of intraspecific diversity and plastid transmission. *Taxon* 1991:393–412.
12. Roy U, Grewal RK, Roy S. Complex networks and systems biology. In *Systems and Synthetic Biology*. Springer; 2015:129–150.
13. Fraser CM, Gocayne JD, White O, Adams MD, Clayton RA, Fleischmann RD, Bult CJ et al. The minimal gene complement of Mycoplasma genitalium. *Science* 1995, 270:397–403.
14. Glass JI, Assad-Garcia N, Alperovich N, Yooseph S, Lewis MR, Maruf M, Hutchison CA, 3rd, Smith HO, Venter JC. Essential genes of a minimal bacterium. *Proc Natl Acad Sci USA* 2006, 103:425–430.
15. Nakabachi A, Yamashita A, Toh H, Ishikawa H, Dunbar HE, Moran NA, Hattori M. The 160-kilobase genome of the bacterial endosymbiont Carsonella. *Science* 2006, 314:267.
16. Colbourne JK, Pfrender ME, Gilbert D, Thomas WK, Tucker A, Oakley TH, Tokishita S et al. The ecoresponsive genome of Daphnia pulex. *Science* 2011, 331:555–561.
17. Shapiro JA, von Sternberg R. Why repetitive DNA is essential to genome function. *Biol Rev Camb Philos Soc* 2005, 80:227–250.
18. Roach JC, Glusman G, Smit AF, Huff CD, Hubley R, Shannon PT, Rowen L et al. Analysis of genetic inheritance in a family quartet by whole-genome sequencing. *Science* 2010, 328:636–639.

19. Alkan C, Coe BP, Eichler EE. Genome structural variation discovery and genotyping. *Nat Rev Genet* 2011, 12:363–376.
20. Malnic B, Godfrey PA, Buck LB. The human olfactory receptor gene family. *Proc Natl Acad Sci USA* 2004, 101:2584–2589.
21. Inai Y, Ohta Y, Nishikimi M. The whole structure of the human nonfunctional L-gulono-gamma-lactone oxidase gene—the gene responsible for scurvy—and the evolution of repetitive sequences thereon. *J Nutr Sci Vitaminol (Tokyo)* 2003, 49:315–319.
22. Law JA, Jacobsen SE. Establishing, maintaining and modifying DNA methylation patterns in plants and animals. *Nat Rev Genet* 2010, 11:204–220.
23. Cedar H, Bergman Y. Linking DNA methylation and histone modification: Patterns and paradigms. *Nat Rev Genet* 2009, 10:295–304.
24. Guo W, Chung WY, Qian M, Pellegrini M, Zhang MQ. Characterizing the strand-specific distribution of non-CpG methylation in human pluripotent cells. *Nucleic Acids Res* 2014, 42:3009–3016.
25. Wu H, Zhang Y. Reversing DNA methylation: Mechanisms, genomics, and biological functions. *Cell* 2014, 156:45–68.
26. Bertram L, McQueen MB, Mullin K, Blacker D, Tanzi RE. Systematic meta-analyses of Alzheimer disease genetic association studies: The AlzGene database. *Nat Genet* 2007, 39:17–23.
27. Baylin SB, Jones PA. A decade of exploring the cancer epigenome—Biological and translational implications. *Nat Rev Cancer* 2011, 11:726–734.
28. Ehrlich M. DNA hypomethylation in cancer cells. *Epigenomics* 2009, 1:239–259.
29. Serganov A, Nudler E. A decade of riboswitches. *Cell* 2013, 152:17–24.
30. Ray PS, Jia J, Yao P, Majumder M, Hatzoglou M, Fox PL. A stress-responsive RNA switch regulates VEGFA expression. *Nature* 2009, 457:915–919.
31. Wan Y, Kertesz M, Spitale RC, Segal E, Chang HY. Understanding the transcriptome through RNA structure. *Nat Rev Genet* 2011, 12:641–655.
32. Imashimizu M, Oshima T, Lubkowska L, Kashlev M. Direct assessment of transcription fidelity by high-resolution RNA sequencing. *Nucleic Acids Res* 2013, 41:9090–9104.
33. Wang ET, Sandberg R, Luo S, Khrebtukova I, Zhang L, Mayr C, Kingsmore SF, Schroth GP, Burge CB. Alternative isoform regulation in human tissue transcriptomes. *Nature* 2008, 456:470–476.
34. Pan Q, Shai O, Lee LJ, Frey BJ, Blencowe BJ. Deep surveying of alternative splicing complexity in the human transcriptome by high-throughput sequencing. *Nat Genet* 2008, 40:1413–1415.
35. Keegan LP, Gallo A, O'Connell MA. The many roles of an RNA editor. *Nat Rev Genet* 2001, 2:869–878.
36. Bratt E, Ohman M. Coordination of editing and splicing of glutamate receptor pre-mRNA. *RNA* 2003, 9:309–318.
37. Pfeiffer BE, Huber KM. Current advances in local protein synthesis and synaptic plasticity. *J Neurosci* 2006, 26:7147–7150.
38. Rustad TR, Minch KJ, Brabant W, Winkler JK, Reiss DJ, Baliga NS, Sherman DR. Global analysis of mRNA stability in Mycobacterium tuberculosis. *Nucleic Acids Res* 2013, 41:509–517.
39. Sharova LV, Sharov AA, Nedorezov T, Piao Y, Shaik N, Ko MS. Database for mRNA half-life of 19 977 genes obtained by DNA microarray analysis of pluripotent and differentiating mouse embryonic stem cells. *DNA Res* 2009, 16:45–58.

40. Yang E, van Nimwegen E, Zavolan M, Rajewsky N, Schroeder M, Magnasco M, Darnell JE, Jr. Decay rates of human mRNAs: Correlation with functional characteristics and sequence attributes. *Genome Res* 2003, 13:1863–1872.
41. Figueroa A, Cuadrado A, Fan J, Atasoy U, Muscat GE, Munoz-Canoves P, Gorospe M, Munoz A. Role of HuR in skeletal myogenesis through coordinate regulation of muscle differentiation genes. *Mol Cell Biol* 2003, 23:4991–5004.
42. Kulkarni M, Ozgur S, Stoecklin G. On track with P-bodies. *Biochem Soc Trans* 2010, 38:242–251.
43. Garneau NL, Wilusz J, Wilusz CJ. The highways and byways of mRNA decay. *Nat Rev Mol Cell Biol* 2007, 8:113–126.
44. Willis DE, Twiss JL. Regulation of protein levels in subcellular domains through mRNA transport and localized translation. *Mol Cell Proteomics* 2010, 9:952–962.
45. Jeffares DC, Poole AM, Penny D. Relics from the RNA world. *J Mol Evol* 1998, 46:18–36.
46. Cech TR. Structural biology. The ribosome is a ribozyme. *Science* 2000, 289:878–879.
47. Wilson RC, Doudna JA. Molecular mechanisms of RNA interference. *Annu Rev Biophys* 2013, 42:217–239.
48. Friedman RC, Farh KK, Burge CB, Bartel DP. Most mammalian mRNAs are conserved targets of microRNAs. *Genome Res* 2009, 19:92–105.
49. Kawamata T, Tomari Y. Making RISC. *Trends Biochem Sci* 2010, 35:368–376.
50. Carthew RW, Sontheimer EJ. Origins and Mechanisms of miRNAs and siRNAs. *Cell* 2009, 136:642–655.
51. Guttman M, Amit I, Garber M, French C, Lin MF, Feldser D, Huarte M et al. Chromatin signature reveals over a thousand highly conserved large non-coding RNAs in mammals. *Nature* 2009, 458:223–227.
52. Ponjavic J, Ponting CP, Lunter G. Functionality or transcriptional noise? Evidence for selection within long noncoding RNAs. *Genome Res* 2007, 17:556–565.
53. Liu X, Hao L, Li D, Zhu L, Hu S. Long non-coding RNAs and their biological roles in plants. *Genomics Proteomics Bioinformatics* 2015, 13:137–147.
54. Derrien T, Johnson R, Bussotti G, Tanzer A, Djebali S, Tilgner H, Guernec G et al. The GENCODE v7 catalog of human long noncoding RNAs: Analysis of their gene structure, evolution, and expression. *Genome Res* 2012, 22:1775–1789.
55. Gupta RA, Shah N, Wang KC, Kim J, Horlings HM, Wong DJ, Tsai MC et al. Long non-coding RNA HOTAIR reprograms chromatin state to promote cancer metastasis. *Nature* 2010, 464:1071–1076.
56. Zhao J, Sun BK, Erwin JA, Song JJ, Lee JT. Polycomb proteins targeted by a short repeat RNA to the mouse X chromosome. *Science* 2008, 322:750–756.
57. Li W, Notani D, Ma Q, Tanasa B, Nunez E, Chen AY, Merkurjev D et al. Functional roles of enhancer RNAs for oestrogen-dependent transcriptional activation. *Nature* 2013, 498:516–520.
58. Yoon JH, Abdelmohsen K, Srikantan S, Yang X, Martindale JL, De S, Huarte M, Zhan M, Becker KG, Gorospe M. LincRNA-p21 suppresses target mRNA translation. *Mol Cell* 2012, 47:648–655.
59. Gong C, Maquat LE. lncRNAs transactivate STAU1-mediated mRNA decay by duplexing with 3′ UTRs via Alu elements. *Nature* 2011, 470:284–288.
60. Yarmishyn AA, Kurochkin IV. Long noncoding RNAs: A potential novel class of cancer biomarkers. *Front Genet* 2015, 6:145.

61. Memczak S, Jens M, Elefsinioti A, Torti F, Krueger J, Rybak A, Maier L et al. Circular RNAs are a large class of animal RNAs with regulatory potency. *Nature* 2013, 495:333–338.
62. Hansen TB, Jensen TI, Clausen BH, Bramsen JB, Finsen B, Damgaard CK, Kjems J. Natural RNA circles function as efficient microRNA sponges. *Nature* 2013, 495:384–388.
63. Cech TR, Steitz JA. The noncoding RNA revolution-trashing old rules to forge new ones. *Cell* 2014, 157:77–94.
64. Carninci P, Kasukawa T, Katayama S, Gough J, Frith MC, Maeda N, Oyama R et al. The transcriptional landscape of the mammalian genome. *Science* 2005, 309:1559–1563.
65. Djebali S, Davis CA, Merkel A, Dobin A, Lassmann T, Mortazavi A, Tanzer A et al. Landscape of transcription in human cells. *Nature* 2012, 489:101–108.
66. Margulies M, Egholm M, Altman WE, Attiya S, Bader JS, Bemben LA, Berka J et al. Genome sequencing in microfabricated high-density picolitre reactors. *Nature* 2005, 437:376–380.
67. Bentley DR, Balasubramanian S, Swerdlow HP, Smith GP, Milton J, Brown CG, Hall KP et al. Accurate whole human genome sequencing using reversible terminator chemistry. *Nature* 2008, 456:53–59.
68. Eid J, Fehr A, Gray J, Luong K, Lyle J, Otto G, Peluso P et al. Real-time DNA sequencing from single polymerase molecules. *Science* 2009, 323:133–138.
69. Poptsova MS, Il'icheva IA, Nechipurenko DY, Panchenko LA, Khodikov MV, Oparina NY, Polozov RV, Nechipurenko YD, Grokhovsky SL. Non-random DNA fragmentation in next-generation sequencing. *Sci Rep* 2014, 4:4532.
70. Seguin-Orlando A, Schubert M, Clary J, Stagegaard J, Alberdi MT, Prado JL, Prieto A, Willerslev E, Orlando L. Ligation bias in illumina next-generation DNA libraries: Implications for sequencing ancient genomes. *PLoS One* 2013, 8:e78575.
71. Hafner M, Renwick N, Brown M, Mihailovic A, Holoch D, Lin C, Pena JT et al. RNA-ligase-dependent biases in miRNA representation in deep-sequenced small RNA cDNA libraries. *RNA* 2011, 17:1697–1712.
72. Aird D, Ross MG, Chen WS, Danielsson M, Fennell T, Russ C, Jaffe DB, Nusbaum C, Gnirke A. Analyzing and minimizing PCR amplification bias in Illumina sequencing libraries. *Genome Biol* 2011, 12:R18.
73. FastQC, http://www.bioinformatics.babraham.ac.uk/projects/fastqc/.
74. FASTX-Toolkit, http://hannonlab.cshl.edu/fastx_toolkit/.
75. Patel RK, Jain M. NGS QC Toolkit: A toolkit for quality control of next generation sequencing data. *PLoS One* 2012, 7:e30619.
76. Chen C, Khaleel SS, Huang H, Wu CH. Software for pre-processing Illumina next-generation sequencing short read sequences. *Source Code Biol Med* 2014, 9:8.
77. Sickle: A sliding-window, adaptive, quality-based trimming tool for FastQ files (version 1.33) [software], https://github.com/najoshi/sickle.
78. Bolger AM, Lohse M, Usadel B. Trimmomatic: A flexible trimmer for Illumina sequence data. *Bioinformatics* 2014, 30:2114–2120.
79. Li R, Li Y, Kristiansen K, Wang J. SOAP: Short oligonucleotide alignment program. *Bioinformatics* 2008, 24:713–714.
80. Li H, Ruan J, Durbin R. Mapping short DNA sequencing reads and calling variants using mapping quality scores. *Genome Res* 2008, 18:1851–1858.
81. Burrows M, Wheeler DJ. *A block-sorting lossless data compression algorithm*. Digital Systems Research Center: Palo Alto, CA, Research Report 124, 1994.

82. Li H, Durbin R. Fast and accurate long-read alignment with Burrows-Wheeler transform. *Bioinformatics* 2010, 26:589–595.
83. Langmead B, Salzberg SL. Fast gapped-read alignment with Bowtie 2. *Nat Methods* 2012, 9:357–359.
84. Li R, Yu C, Li Y, Lam TW, Yiu SM, Kristiansen K, Wang J. SOAP2: An improved ultrafast tool for short read alignment. *Bioinformatics* 2009, 25:1966–1967.
85. Lunter G, Goodson M. Stampy: A statistical algorithm for sensitive and fast mapping of Illumina sequence reads. *Genome Res* 2011, 21:936–939.
86. David M, Dzamba M, Lister D, Ilie L, Brudno M. SHRiMP2: Sensitive yet practical Short Read Mapping. *Bioinformatics* 2011, 27:1011–1012.
87. Li H. Aligning sequence reads, clone sequences and assembly contigs with BWA-MEM. arXiv preprint arXiv:13033997 2013.
88. Chaisson MJ, Tesler G. Mapping single molecule sequencing reads using basic local alignment with successive refinement (BLASR): Application and theory. *BMC Bioinformatics* 2012, 13:238.
89. Kielbasa SM, Wan R, Sato K, Horton P, Frith MC. Adaptive seeds tame genomic sequence comparison. *Genome Res* 2011, 21:487–493.
90. Harris RS. Improved pairwise alignment of genomic DNA. PhD dissertation, Pennslyvania State University, 2007. ProQuest.
91. Schneeberger K, Hagmann J, Ossowski S, Warthmann N, Gesing S, Kohlbacher O, Weigel D. Simultaneous alignment of short reads against multiple genomes. *Genome Biol* 2009, 10:R98.
92. Yuan Y, Norris C, Xu Y, Tsui KW, Ji Y, Liang H. BM-Map: An efficient software package for accurately allocating multireads of RNA-sequencing data. *BMC Genomics* 2012, 13 Suppl 8:S9.
93. Thorvaldsdottir H, Robinson JT, Mesirov JP. Integrative Genomics Viewer (IGV): High-performance genomics data visualization and exploration. *Brief Bioinform* 2013, 14:178–192.
94. Huang W, Marth G. EagleView: A genome assembly viewer for next-generation sequencing technologies. *Genome Res* 2008, 18:1538–1543.
95. Milne I, Bayer M, Cardle L, Shaw P, Stephen G, Wright F, Marshall D. Tablet—Next generation sequence assembly visualization. *Bioinformatics* 2010, 26: 401–402.
96. Li R, Zhu H, Ruan J, Qian W, Fang X, Shi Z, Li Y, Li S, Shan G, Kristiansen K et al. De novo assembly of human genomes with massively parallel short read sequencing. *Genome Res* 2010, 20:265–272.
97. Lampa S, Dahlo M, Olason PI, Hagberg J, Spjuth O. Lessons learned from implementing a national infrastructure in Sweden for storage and analysis of next-generation sequencing data. *Gigascience* 2013, 2:9.
98. Rasche A, Lienhard M, Yaspo ML, Lehrach H, Herwig R. ARH-seq: Identification of differential splicing in RNA-seq data. *Nucleic Acids Res* 2014. doi: 10.1093/nar/gku495.
99. Farazi TA, Brown M, Morozov P, Ten Hoeve JJ, Ben-Dov IZ, Hovestadt V, Hafner M et al. Bioinformatic analysis of barcoded cDNA libraries for small RNA profiling by next-generation sequencing. *Methods* 2012, 58:171–187.
100. Galaxy, https://usegalaxy.org.
101. Afgan E, Baker D, Coraor N, Chapman B, Nekrutenko A, Taylor J. Galaxy CloudMan: Delivering cloud compute clusters. *BMC Bioinformatics* 2010, 11 Suppl 12:S4.

102. Zhulidov PA, Bogdanova EA, Shcheglov AS, Vagner LL, Khaspekov GL, Kozhemyako VB, Matz MV et al. Simple cDNA normalization using kamchatka crab duplex-specific nuclease. *Nucleic Acids Res* 2004, 32:e37.
103. Gallego Romero I, Pai AA, Tung J, Gilad Y. RNA-seq: Impact of RNA degradation on transcript quantification. *BMC Biol* 2014, 12:42.
104. Yang L, Duff MO, Graveley BR, Carmichael GG, Chen LL. Genomewide characterization of non-polyadenylated RNAs. *Genome Biol* 2011, 12:R16.
105. Auer PL, Doerge RW. Statistical design and analysis of RNA sequencing data. *Genetics* 2010, 185:405–416.
106. Busby MA, Stewart C, Miller CA, Grzeda KR, Marth GT. Scotty: A web tool for designing RNA-Seq experiments to measure differential gene expression. *Bioinformatics* 2013, 29:656–657.
107. Wang Y, Ghaffari N, Johnson CD, Braga-Neto UM, Wang H, Chen R, Zhou H. Evaluation of the coverage and depth of transcriptome by RNA-Seq in chickens. *BMC Bioinformatics* 2011, 12 Suppl 10:S5.
108. Vijay N, Poelstra JW, Kunstner A, Wolf JB. Challenges and strategies in transcriptome assembly and differential gene expression quantification. A comprehensive in silico assessment of RNA-seq experiments. *Mol Ecol* 2013, 22:620–634.
109. Liu Y, Ferguson JF, Xue C, Silverman IM, Gregory B, Reilly MP, Li M. Evaluating the impact of sequencing depth on transcriptome profiling in human adipose. *PLoS One* 2013, 8:e66883.
110. Ching T, Huang S, Garmire LX. Power analysis and sample size estimation for RNA-Seq differential expression. *RNA* 2014, 20:1684–1696.
111. DeLuca DS, Levin JZ, Sivachenko A, Fennell T, Nazaire MD, Williams C, Reich M, Winckler W, Getz G. RNA-SeQC: RNA-seq metrics for quality control and process optimization. *Bioinformatics* 2012, 28:1530–1532.
112. Wang L, Wang S, Li W. RSeQC: Quality control of RNA-seq experiments. *Bioinformatics* 2012, 28:2184–2185.
113. Tang S, Riva A. PASTA: Splice junction identification from RNA-sequencing data. *BMC Bioinformatics* 2013, 14:116.
114. Chen LY, Wei KC, Huang AC, Wang K, Huang CY, Yi D, Tang CY, Galas DJ, Hood LE. RNASEQR—A streamlined and accurate RNA-seq sequence analysis program. *Nucleic Acids Res* 2012, 40:e42.
115. Grant GR, Farkas MH, Pizarro AD, Lahens NF, Schug J, Brunk BP, Stoeckert CJ, Hogenesch JB, Pierce EA. Comparative analysis of RNA-Seq alignment algorithms and the RNA-Seq unified mapper (RUM). *Bioinformatics* 2011, 27:2518–2528.
116. Xu G, Deng N, Zhao Z, Judeh T, Flemington E, Zhu D. SAMMate: A GUI tool for processing short read alignments in SAM/BAM format. *Source Code Biol Med* 2011, 6:2.
117. Ryan MC, Cleland J, Kim R, Wong WC, Weinstein JN. SpliceSeq: A resource for analysis and visualization of RNA-Seq data on alternative splicing and its functional impacts. *Bioinformatics* 2012, 28:2385–2387.
118. Trapnell C, Pachter L, Salzberg SL. TopHat: Discovering splice junctions with RNA-Seq. *Bioinformatics* 2009, 25:1105–1111.
119. Kim D, Pertea G, Trapnell C, Pimentel H, Kelley R, Salzberg SL. TopHat2: Accurate alignment of transcriptomes in the presence of insertions, deletions and gene fusions. *Genome Biol* 2013, 14:R36.

120. Wang K, Singh D, Zeng Z, Coleman SJ, Huang Y, Savich GL, He X et al. MapSplice: Accurate mapping of RNA-seq reads for splice junction discovery. *Nucleic Acids Res* 2010, 38:e178.
121. Au KF, Jiang H, Lin L, Xing Y, Wong WH. Detection of splice junctions from paired-end RNA-seq data by SpliceMap. *Nucleic Acids Res* 2010, 38:4570–4578.
122. Dimon MT, Sorber K, DeRisi JL. HMMSplicer: A tool for efficient and sensitive discovery of known and novel splice junctions in RNA-Seq data. *PLoS One* 2010, 5:e13875.
123. Marco-Sola S, Sammeth M, Guigo R, Ribeca P. The GEM mapper: Fast, accurate and versatile alignment by filtration. *Nat Methods* 2012, 9:1185–1188.
124. Wu TD, Nacu S. Fast and SNP-tolerant detection of complex variants and splicing in short reads. *Bioinformatics* 2010, 26:873–881.
125. Bao H, Xiong Y, Guo H, Zhou R, Lu X, Yang Z, Zhong Y, Shi S. MapNext: A software tool for spliced and unspliced alignments and SNP detection of short sequence reads. *BMC Genomics* 2009, 10 Suppl 3:S13.
126. Ameur A, Wetterbom A, Feuk L, Gyllensten U. Global and unbiased detection of splice junctions from RNA-seq data. *Genome Biol* 2010, 11:R34.
127. Dobin A, Davis CA, Schlesinger F, Drenkow J, Zaleski C, Jha S, Batut P, Chaisson M, Gingeras TR. *STAR*: Ultrafast universal RNA-seq aligner. *Bioinformatics* 2013, 29:15–21.
128. Schulz MH, Zerbino DR, Vingron M, Birney E. Oases: Robust de novo RNA-seq assembly across the dynamic range of expression levels. *Bioinformatics* 2012, 28:1086–1092.
129. Xie Y, Wu G, Tang J, Luo R, Patterson J, Liu S, Huang W et al. SOAPdenovo-Trans: De novo transcriptome assembly with short RNA-Seq reads. *Bioinformatics* 2014, 30:1660–1666.
130. Robertson G, Schein J, Chiu R, Corbett R, Field M, Jackman SD, Mungall K et al. De novo assembly and analysis of RNA-seq data. *Nat Methods* 2010, 7:909–912.
131. Grabherr MG, Haas BJ, Yassour M, Levin JZ, Thompson DA, Amit I, Adiconis X et al. Full-length transcriptome assembly from RNA-Seq data without a reference genome. *Nat Biotechnol* 2011, 29:644–652.
132. Bullard JH, Purdom E, Hansen KD, Dudoit S. Evaluation of statistical methods for normalization and differential expression in mRNA-Seq experiments. *BMC Bioinformatics* 2010, 11:94.
133. Law CW, Chen Y, Shi W, Smyth GK. Voom: Precision weights unlock linear model analysis tools for RNA-seq read counts. *Genome Biol* 2014, 15:R29.
134. Risso D, Schwartz K, Sherlock G, Dudoit S. GC-content normalization for RNA-Seq data. *BMC Bioinformatics* 2011, 12:480.
135. Marioni JC, Mason CE, Mane SM, Stephens M, Gilad Y. RNA-seq: An assessment of technical reproducibility and comparison with gene expression arrays. *Genome Res* 2008, 18:1509–1517.
136. Anders S, Huber W. Differential expression analysis for sequence count data. *Genome Biol* 2010, 11:R106.
137. Li J, Witten DM, Johnstone IM, Tibshirani R. Normalization, testing, and false discovery rate estimation for RNA-sequencing data. *Biostatistics* 2012, 13:523–538.
138. Hardcastle TJ, Kelly KA. baySeq: Empirical Bayesian methods for identifying differential expression in sequence count data. *BMC Bioinformatics* 2010, 11:422.

139. Trapnell C, Williams BA, Pertea G, Mortazavi A, Kwan G, van Baren MJ, Salzberg SL, Wold BJ, Pachter L. Transcript assembly and quantification by RNA-Seq reveals unannotated transcripts and isoform switching during cell differentiation. *Nat Biotechnol* 2010, 28:511–515.
140. Trapnell C, Hendrickson DG, Sauvageau M, Goff L, Rinn JL, Pachter L. Differential analysis of gene regulation at transcript resolution with RNA-seq. *Nat Biotechnol* 2013, 31:46–53.
141. Wang L, Feng Z, Wang X, Wang X, Zhang X. DEGseq: An R package for identifying differentially expressed genes from RNA-seq data. *Bioinformatics* 2010, 26:136–138.
142. Love MI, Huber W, Anders S. Moderated estimation of fold change and dispersion for RNA-Seq data with DESeq2. bioRxiv 2014.
143. Robinson MD, McCarthy DJ, Smyth GK. edgeR: A Bioconductor package for differential expression analysis of digital gene expression data. *Bioinformatics* 2010, 26:139–140.
144. Li J, Tibshirani R. Finding consistent patterns: A nonparametric approach for identifying differential expression in RNA-Seq data. *Stat Methods Med Res* 2013, 22:519–536.
145. Audic S, Claverie JM. The significance of digital gene expression profiles. *Genome Res* 1997, 7:986–995.
146. Katz Y, Wang ET, Airoldi EM, Burge CB. Analysis and design of RNA sequencing experiments for identifying isoform regulation. *Nat Methods* 2010, 7:1009–1015.
147. Wu J, Akerman M, Sun S, McCombie WR, Krainer AR, Zhang MQ. SpliceTrap: A method to quantify alternative splicing under single cellular conditions. *Bioinformatics* 2011, 27:3010–3016.
148. Roberts A, Pimentel H, Trapnell C, Pachter L. Identification of novel transcripts in annotated genomes using RNA-Seq. *Bioinformatics* 2011, 27:2325–2329.
149. Li JJ, Jiang CR, Brown JB, Huang H, Bickel PJ. Sparse linear modeling of next-generation mRNA sequencing (RNA-Seq) data for isoform discovery and abundance estimation. *Proc Natl Acad Sci USA* 2011, 108:19867–19872.
150. Mangul S, Caciula A, Glebova O, Mandoiu I, Zelikovsky A. Improved transcriptome quantification and reconstruction from RNA-Seq reads using partial annotations. *In Silico Biol* 2011, 11:251–261.
151. Griffith M, Griffith OL, Mwenifumbo J, Goya R, Morrissy AS, Morin RD, Corbett R et al.: Alternative expression analysis by RNA sequencing. *Nat Methods* 2010, 7:843–847.
152. Singh D, Orellana CF, Hu Y, Jones CD, Liu Y, Chiang DY, Liu J, Prins JF. FDM: A graph-based statistical method to detect differential transcription using RNA-seq data. *Bioinformatics* 2011, 27:2633–2640.
153. Drewe P, Stegle O, Hartmann L, Kahles A, Bohnert R, Wachter A, Borgwardt K, Ratsch G. Accurate detection of differential RNA processing. *Nucleic Acids Res* 2013, 41:5189–5198.
154. Shi Y, Jiang H. rSeqDiff: Detecting differential isoform expression from RNA-Seq data using hierarchical likelihood ratio test. *PLoS One* 2013, 8:e79448.
155. Zheng S, Chen L. A hierarchical Bayesian model for comparing transcriptomes at the individual transcript isoform level. *Nucleic Acids Res* 2009, 37:e75.
156. Glaus P, Honkela A, Rattray M. Identifying differentially expressed transcripts from RNA-seq data with biological variation. *Bioinformatics* 2012, 28: 1721–1728.

157. Leng N, Dawson JA, Thomson JA, Ruotti V, Rissman AI, Smits BM, Haag JD, Gould MN, Stewart RM, Kendziorski C. EBSeq: An empirical Bayes hierarchical model for inference in RNA-seq experiments. *Bioinformatics* 2013, 29:1035–1043.
158. Li B, Dewey CN. RSEM: Accurate transcript quantification from RNA-Seq data with or without a reference genome. *BMC Bioinformatics* 2011, 12:323.
159. Nicolae M, Mangul S, Mandoiu, II, Zelikovsky A. Estimation of alternative splicing isoform frequencies from RNA-Seq data. *Algorithms Mol Biol* 2011, 6:9.
160. Martin J, Bruno VM, Fang Z, Meng X, Blow M, Zhang T, Sherlock G, Snyder M, Wang Z. Rnnotator: An automated de novo transcriptome assembly pipeline from stranded RNA-Seq reads. *BMC Genomics* 2010, 11:663.
161. Sacomoto GA, Kielbassa J, Chikhi R, Uricaru R, Antoniou P, Sagot MF, Peterlongo P, Lacroix V. KISSPLICE: De-novo calling alternative splicing events from RNA-seq data. *BMC Bioinformatics* 2012, 13 Suppl 6:S5.
162. Huang da W, Sherman BT, Lempicki RA. Systematic and integrative analysis of large gene lists using DAVID bioinformatics resources. *Nat Protoc* 2009, 4:44–57.
163. Subramanian A, Tamayo P, Mootha VK, Mukherjee S, Ebert BL, Gillette MA, Paulovich A et al. Gene set enrichment analysis: A knowledge-based approach for interpreting genome-wide expression profiles. *Proc Natl Acad Sci USA* 2005, 102:15545–15550.
164. Salzman J, Gawad C, Wang PL, Lacayo N, Brown PO. Circular RNAs are the predominant transcript isoform from hundreds of human genes in diverse cell types. *PLoS One* 2012, 7:e30733.
165. Davare MA, Tognon CE. Detecting and targeting oncogenic fusion proteins in the genomic era. *Biol Cell* 2015, 107(5):111–129.
166. Huang V, Qin Y, Wang J, Wang X, Place RF, Lin G, Lue TF, Li LC. RNAa is conserved in mammalian cells. *PLoS One* 2010, 5:e8848.
167. Alon S, Vigneault F, Eminaga S, Christodoulou DC, Seidman JG, Church GM, Eisenberg E. Barcoding bias in high-throughput multiplex sequencing of miRNA. *Genome Res* 2011, 21:1506–1511.
168. Hafner M, Renwick N, Brown M, Mihailovic A, Holoch D, Lin C, Pena JT et al. RNA-ligase-dependent biases in miRNA representation in deep-sequenced small RNA cDNA libraries. *RNA* 2011, 17:1697–1712.
169. Linsen SE, de Wit E, Janssens G, Heater S, Chapman L, Parkin RK, Fritz B et al. Limitations and possibilities of small RNA digital gene expression profiling. *Nat Methods* 2009, 6:474–476.
170. Tian G, Yin X, Luo H, Xu X, Bolund L, Zhang X, Gan SQ, Li N. Sequencing bias: Comparison of different protocols of microRNA library construction. *BMC Biotechnol* 2010, 10:64.
171. Metpally RP, Nasser S, Malenica I, Courtright A, Carlson E, Ghaffari L, Villa S, Tembe W, Van Keuren-Jensen K. Comparison of analysis tools for miRNA high throughput sequencing using nerve crush as a model. *Front Genet* 2013, 4:20.
172. Huang PJ, Liu YC, Lee CC, Lin WC, Gan RR, Lyu PC, Tang P. DSAP: Deep-sequencing small RNA analysis pipeline. *Nucleic Acids Res* 2010, 38:W385–391.
173. Hackenberg M, Sturm M, Langenberger D, Falcon-Perez JM, Aransay AM. miRanalyzer: A microRNA detection and analysis tool for next-generation sequencing experiments. *Nucleic Acids Res* 2009, 37:W68–76.
174. Friedlander MR, Mackowiak SD, Li N, Chen W, Rajewsky N. miRDeep2 accurately identifies known and hundreds of novel microRNA genes in seven animal clades. *Nucleic Acids Res* 2012, 40:37–52.

175. Wang WC, Lin FM, Chang WC, Lin KY, Huang HD, Lin NS. miRExpress: Analyzing high-throughput sequencing data for profiling microRNA expression. *BMC Bioinformatics* 2009, 10:328.
176. Ronen R, Gan I, Modai S, Sukacheov A, Dror G, Halperin E, Shomron N. miRNAkey: A software for microRNA deep sequencing analysis. *Bioinformatics* 2010, 26:2615–2616.
177. Zhu E, Zhao F, Xu G, Hou H, Zhou L, Li X, Sun Z, Wu J. mirTools: microRNA profiling and discovery based on high-throughput sequencing. *Nucleic Acids Res* 2010, 38:W392–397.
178. Burge SW, Daub J, Eberhardt R, Tate J, Barquist L, Nawrocki EP, Eddy SR, Gardner PP, Bateman A. Rfam 11.0: 10 years of RNA families. *Nucleic Acids Res* 2013, 41:D226–232.
179. Axtell MJ. Butter: High-precision genomic alignment of small RNA-seq data. bioRxiv 2014:007427.
180. Morin RD, O'Connor MD, Griffith M, Kuchenbauer F, Delaney A, Prabhu AL, Zhao Y et al. Application of massively parallel sequencing to microRNA profiling and discovery in human embryonic stem cells. *Genome Res* 2008, 18:610–621.
181. Fernandez-Valverde SL, Taft RJ, Mattick JS. Dynamic isomiR regulation in Drosophila development. *RNA* 2010, 16:1881–1888.
182. Li SC, Tsai KW, Pan HW, Jeng YM, Ho MR, Li WH. MicroRNA 3′ end nucleotide modification patterns and arm selection preference in liver tissues. *BMC Syst Biol* 2012, 6 Suppl 2:S14.
183. Pan CT, Tsai KW, Hung TM, Lin WC, Pan CY, Yu HR, Li SC. miRSeq: A user-friendly standalone toolkit for sequencing quality evaluation and miRNA profiling. *Biomed Res Int* 2014, 2014:462135.
184. Pantano L, Estivill X, Marti E. SeqBuster, a bioinformatic tool for the processing and analysis of small RNAs datasets, reveals ubiquitous miRNA modifications in human embryonic cells. *Nucleic Acids Res* 2010, 38:e34.
185. Garmire LX, Subramaniam S. Evaluation of normalization methods in mammalian microRNA-Seq data. *RNA* 2012, 18:1279–1288.
186. Dillies MA, Rau A, Aubert J, Hennequet-Antier C, Jeanmougin M, Servant N, Keime C et al. A comprehensive evaluation of normalization methods for Illumina high-throughput RNA sequencing data analysis. *Brief Bioinform* 2013, 14:671–683.
187. Enright AJ, John B, Gaul U, Tuschl T, Sander C, Marks DS. MicroRNA targets in Drosophila. *Genome Biol* 2003, 5:R1.
188. Betel D, Koppal A, Agius P, Sander C, Leslie C. Comprehensive modeling of microRNA targets predicts functional non-conserved and non-canonical sites. *Genome Biol* 2010, 11:R90.
189. Krek A, Grun D, Poy MN, Wolf R, Rosenberg L, Epstein EJ, MacMenamin P et al. Combinatorial microRNA target predictions. *Nat Genet* 2005, 37:495–500.
190. Kertesz M, Iovino N, Unnerstall U, Gaul U, Segal E. The role of site accessibility in microRNA target recognition. *Nat Genet* 2007, 39:1278–1284.
191. Miranda KC, Huynh T, Tay Y, Ang YS, Tam WL, Thomson AM, Lim B, Rigoutsos I. A pattern-based method for the identification of MicroRNA binding sites and their corresponding heteroduplexes. *Cell* 2006, 126:1203–1217.
192. Rehmsmeier M, Steffen P, Hochsmann M, Giegerich R. Fast and effective prediction of microRNA/target duplexes. *RNA* 2004, 10:1507–1517.

193. Lewis BP, Burge CB, Bartel DP. Conserved seed pairing, often flanked by adenosines, indicates that thousands of human genes are microRNA targets. *Cell* 2005, 120:15–20.
194. Reczko M, Maragkakis M, Alexiou P, Grosse I, Hatzigeorgiou AG. Functional microRNA targets in protein coding sequences. *Bioinformatics* 2012, 28:771–776.
195. Maragkakis M, Alexiou P, Papadopoulos GL, Reczko M, Dalamagas T, Giannopoulos G, Goumas G et al. Accurate microRNA target prediction correlates with protein repression levels. *BMC Bioinformatics* 2009, 10:295.
196. Ritchie W, Flamant S, Rasko JE. Predicting microRNA targets and functions: Traps for the unwary. *Nat Methods* 2009, 6:397–398.
197. Vlachos IS, Kostoulas N, Vergoulis T, Georgakilas G, Reczko M, Maragkakis M, Paraskevopoulou MD, Prionidis K, Dalamagas T, Hatzigeorgiou AG. DIANA miRPath v.2.0: Investigating the combinatorial effect of microRNAs in pathways. *Nucleic Acids Res* 2012, 40:W498–504.
198. Veltman JA, Brunner HG. De novo mutations in human genetic disease. *Nat Rev Genet* 2012, 13(8):565–575.
199. Meyerson M, Gabriel S, Getz G. Advances in understanding cancer genomes through second-generation sequencing. *Nat Rev Genet* 2010, 11(10):685–696.
200. McKenna A, Hanna M, Banks E, Sivachenko A, Cibulskis K, Kernytsky A, Garimella K et al. The Genome Analysis Toolkit: A MapReduce framework for analyzing next-generation DNA sequencing data. *Genome Res* 2010, 20(9):1297–1303.
201. Li H, Handsaker B, Wysoker A, Fennell T, Ruan J, Homer N, Marth G, Abecasis G, Durbin R, Genome Project Data Processing Subgroup. The Sequence Alignment/Map format and SAMtools. *Bioinformatics* 2009, 25(16):2078–2079.
202. Li R, Li Y, Fang X, Yang H, Wang J, Kristiansen K, Wang J. SNP detection for massively parallel whole-genome resequencing. *Genome Res* 2009, 19(6):1124–1132.
203. Challis D, Yu J, Evani US, Jackson AR, Paithankar S, Coarfa C, Milosavljevic A, Gibbs RA, Yu F. An integrative variant analysis suite for whole exome next-generation sequencing data. *BMC Bioinformatics* 2012, 13:8.
204. Koboldt DC, Zhang Q, Larson DE, Shen D, McLellan MD, Lin L, Miller CA, Mardis ER, Ding L, Wilson RK. VarScan 2: Somatic mutation and copy number alteration discovery in cancer by exome sequencing. *Genome Res* 2012, 22(3):568–576.
205. Cibulskis K, Lawrence MS, Carter SL, Sivachenko A, Jaffe D, Sougnez C, Gabriel S, Meyerson M, Lander ES, Getz G. Sensitive detection of somatic point mutations in impure and heterogeneous cancer samples. *Nat Biotechnol* 2013, 31(3):213–219.
206. Larson DE, Harris CC, Chen K, Koboldt DC, Abbott TE, Dooling DJ, Ley TJ, Mardis ER, Wilson RK, Ding L. SomaticSniper: Identification of somatic point mutations in whole genome sequencing data. *Bioinformatics* 2012, 28(3):311–317.
207. Saunders CT, Wong WS, Swamy S, Becq J, Murray LJ, Cheetham RK. Strelka: Accurate somatic small-variant calling from sequenced tumor-normal sample pairs. *Bioinformatics* 2012, 28(14):1811–1817.
208. Roth A, Ding J, Morin R, Crisan A, Ha G, Giuliany R, Bashashati A et al. JointSNVMix: A probabilistic model for accurate detection of somatic mutations in normal/tumour paired next-generation sequencing data. *Bioinformatics* 2012, 28(7):907–913.

209. Albers CA, Lunter G, MacArthur DG, McVean G, Ouwehand WH, Durbin R. Dindel: Accurate indel calls from short-read data. *Genome Res* 2011, 21(6):961–973.
210. Li S, Li R, Li H, Lu J, Li Y, Bolund L, Schierup MH, Wang J. SOAPindel: Efficient identification of indels from short paired reads. *Genome Res* 2013, 23(1):195–200.
211. Tang X, Baheti S, Shameer K, Thompson KJ, Wills Q, Niu N, Holcomb IN et al. The eSNV-detect: A computational system to identify expressed single nucleotide variants from transcriptome sequencing data. *Nucleic Acids Res* 2014, 42(22):e172.
212. Piskol R, Ramaswami G, Li JB. Reliable identification of genomic variants from RNA-seq data. *Am J Hum Genet* 2013, 93(4):641–651.
213. Goya R, Sun MG, Morin RD, Leung G, Ha G, Wiegand KC, Senz J et al. SNVMix: Predicting single nucleotide variants from next-generation sequencing of tumors. *Bioinformatics* 2010, 26(6):730–736.
214. Danecek P, Auton A, Abecasis G, Albers CA, Banks E, DePristo MA, Handsaker RE et al. The variant call format and VCFtools. *Bioinformatics* 2011, 27(15):2156–2158.
215. vcflib, https://github.com/ekg/vcflib.
216. Freudenberg-Hua Y, Freudenberg J, Kluck N, Cichon S, Propping P, Nothen MM. Single nucleotide variation analysis in 65 candidate genes for CNS disorders in a representative sample of the European population. *Genome Res* 2003, 13(10):2271–2276.
217. Chen K, Wallis JW, McLellan MD, Larson DE, Kalicki JM, Pohl CS, McGrath SD et al. BreakDancer: An algorithm for high-resolution mapping of genomic structural variation. *Nat Methods* 2009, 6(9):677–681.
218. Sindi S, Helman E, Bashir A, Raphael BJ. A geometric approach for classification and comparison of structural variants. *Bioinformatics* 2009, 25(12):i222–i230.
219. Quinlan AR, Clark RA, Sokolova S, Leibowitz ML, Zhang Y, Hurles ME, Mell JC, Hall IM. Genome-wide mapping and assembly of structural variant breakpoints in the mouse genome. *Genome Res* 2010, 20(5):623–635.
220. Korbel JO, Abyzov A, Mu XJ, Carriero N, Cayting P, Zhang Z, Snyder M, Gerstein MB. PEMer: A computational framework with simulation-based error models for inferring genomic structural variants from massive paired-end sequencing data. *Genome Biol* 2009, 10(2):R23.
221. Zeitouni B, Boeva V, Janoueix-Lerosey I, Loeillet S, Legoix-ne P, Nicolas A, Delattre O, Barillot E. SVDetect: A tool to identify genomic structural variations from paired-end and mate-pair sequencing data. *Bioinformatics* 2010, 26(15):1895–1896.
222. Wang J, Mullighan CG, Easton J, Roberts S, Heatley SL, Ma J, Rusch MC et al. CREST maps somatic structural variation in cancer genomes with base-pair resolution. *Nat Methods* 2011, 8(8):652–654.
223. Ye K, Schulz MH, Long Q, Apweiler R, Ning Z. Pindel: A pattern growth approach to detect break points of large deletions and medium sized insertions from paired-end short reads. *Bioinformatics* 2009, 25(21):2865–2871.
224. Emde AK, Schulz MH, Weese D, Sun R, Vingron M, Kalscheuer VM, Haas SA, Reinert K. Detecting genomic indel variants with exact breakpoints in single- and paired-end sequencing data using SplazerS. *Bioinformatics* 2012, 28(5):619–627.

225. Zhang ZD, Du J, Lam H, Abyzov A, Urban AE, Snyder M, Gerstein M. Identification of genomic indels and structural variations using split reads. *BMC Genomics* 2011, 12:375.
226. Iqbal Z, Caccamo M, Turner I, Flicek P, McVean G. De novo assembly and genotyping of variants using colored de Bruijn graphs. *Nat Genet* 2012, 44(2):226–232.
227. Ivakhno S, Royce T, Cox AJ, Evers DJ, Cheetham RK, Tavare S. CNAseg—A novel framework for identification of copy number changes in cancer from second-generation sequencing data. *Bioinformatics* 2010, 26(24):3051–3058.
228. Xie C, Tammi MT. CNV-seq, a new method to detect copy number variation using high-throughput sequencing. *BMC bioinformatics* 2009, 10:80.
229. Abyzov A, Urban AE, Snyder M, Gerstein M. CNVnator: An approach to discover, genotype, and characterize typical and atypical CNVs from family and population genome sequencing. *Genome Res* 2011, 21(6):974–984.
230. Yoon S, Xuan Z, Makarov V, Ye K, Sebat J. Sensitive and accurate detection of copy number variants using read depth of coverage. *Genome Res* 2009, 19(9):1586–1592.
231. Magi A, Benelli M, Yoon S, Roviello F, Torricelli F. Detecting common copy number variants in high-throughput sequencing data by using JointSLM algorithm. *Nucleic Acids Res* 2011, 39(10):e65.
232. Chiang DY, Getz G, Jaffe DB, O'Kelly MJ, Zhao X, Carter SL, Russ C, Nusbaum C, Meyerson M, Lander ES. High-resolution mapping of copy-number alterations with massively parallel sequencing. *Nat Methods* 2009, 6(1):99–103.
233. Wong K, Keane TM, Stalker J, Adams DJ. Enhanced structural variant and breakpoint detection using SVMerge by integration of multiple detection methods and local assembly. *Genome Biol* 2010, 11(12):R128.
234. Sindi SS, Onal S, Peng LC, Wu HT, Raphael BJ. An integrative probabilistic model for identification of structural variation in sequencing data. *Genome Biol* 2012, 13(3):R22.
235. Zhang J, Wu Y. SVseq: An approach for detecting exact breakpoints of deletions with low-coverage sequence data. *Bioinformatics* 2011, 27(23):3228–3234.
236. Medvedev P, Fiume M, Dzamba M, Smith T, Brudno M. Detecting copy number variation with mated short reads. *Genome Res* 2010, 20(11):1613–1622.
237. Wang K, Li M, Hakonarson H. ANNOVAR: Functional annotation of genetic variants from high-throughput sequencing data. *Nucleic Acids Res* 2010, 38(16):e164.
238. SeattleSeq, http://snp.gs.washington.edu/SeattleSeqAnnotation138/.
239. Cingolani P, Platts A, Wang le L, Coon M, Nguyen T, Wang L, Land SJ, Lu X, Ruden DM. A program for annotating and predicting the effects of single nucleotide polymorphisms, SnpEff: SNPs in the genome of Drosophila melanogaster strain w1118; iso-2; iso-3. *Fly* 2012, 6(2):80–92.
240. McLaren W, Pritchard B, Rios D, Chen Y, Flicek P, Cunningham F. Deriving the consequences of genomic variants with the Ensembl API and SNP Effect Predictor. *Bioinformatics* 2010, 26(16):2069–2070.
241. PSEQ, http://atgu.mgh.harvard.edu/plinkseq/pseq.shtml.
242. Morgenthaler S, Thilly WG. A strategy to discover genes that carry multi-allelic or mono-allelic risk for common diseases: A cohort allelic sums test (CAST). *Mutation Res* 2007, 615(1–2):28–56.

243. Li B, Leal SM. Methods for detecting associations with rare variants for common diseases: Application to analysis of sequence data. *Am J Hum Genet* 2008, 83(3):311–321.
244. Schatz MC, Delcher AL, Salzberg SL. Assembly of large genomes using second-generation sequencing. *Genome Res* 2010, 20:1165–1173.
245. Zerbino DR, Birney E. Velvet: Algorithms for de novo short read assembly using de Bruijn graphs. *Genome Res* 2008, 18:821–829.
246. Simpson JT, Wong K, Jackman SD, Schein JE, Jones SJ, Birol I. ABySS: A parallel assembler for short read sequence data. *Genome Res* 2009, 19:1117–1123.
247. van Heesch S, Kloosterman WP, Lansu N, Ruzius FP, Levandowsky E, Lee CC, Zhou S et al. Improving mammalian genome scaffolding using large insert mate-pair next-generation sequencing. *BMC Genomics* 2013, 14:257.
248. Li R, Fan W, Tian G, Zhu H, He L, Cai J, Huang Q et al. The sequence and de novo assembly of the giant panda genome. *Nature* 2010, 463:311–317.
249. Nagarajan N, Pop M. Sequence assembly demystified. *Nat Rev Genet* 2013, 14:157–167.
250. Desai A, Marwah VS, Yadav A, Jha V, Dhaygude K, Bangar U, Kulkarni V, Jere A. Identification of optimum sequencing depth especially for de novo genome assembly of small genomes using next generation sequencing data. *PLoS One* 2013, 8:e60204.
251. Magoc T, Salzberg SL. FLASH: Fast length adjustment of short reads to improve genome assemblies. *Bioinformatics* 2011, 27:2957–2963.
252. Masella AP, Bartram AK, Truszkowski JM, Brown DG, Neufeld JD. PANDAseq: Paired-end assembler for illumina sequences. *BMC Bioinformatics* 2012, 13:31.
253. Kelley DR, Schatz MC, Salzberg SL. Quake: Quality-aware detection and correction of sequencing errors. *Genome Biol* 2010, 11:R116.
254. Gnerre S, Maccallum I, Przybylski D, Ribeiro FJ, Burton JN, Walker BJ, Sharpe T et al. High-quality draft assemblies of mammalian genomes from massively parallel sequence data. *Proc Natl Acad Sci USA* 2011, 108:1513–1518.
255. Pevzner PA, Tang H, Waterman MS. An Eulerian path approach to DNA fragment assembly. *Proc Natl Acad Sci USA* 2001, 98:9748–9753.
256. Marcais G, Kingsford C. A fast, lock-free approach for efficient parallel counting of occurrences of k-mers. *Bioinformatics* 2011, 27:764–770.
257. Chikhi R, Medvedev P. Informed and automated k-mer size selection for genome assembly. *Bioinformatics* 2014, 30:31–37.
258. Simpson JT. Exploring genome characteristics and sequence quality without a reference. *Bioinformatics* 2014, 30:1228–1235.
259. VelvetOptimiser, https://github.com/Victorian-Bioinformatics-Consortium/VelvetOptimiser.
260. Lander ES, Waterman MS. Genomic mapping by fingerprinting random clones: A mathematical analysis. *Genomics* 1988, 2:231–239.
261. Warren RL, Sutton GG, Jones SJ, Holt RA. Assembling millions of short DNA sequences using SSAKE. *Bioinformatics* 2007, 23:500–501.
262. Dohm JC, Lottaz C, Borodina T, Himmelbauer H. SHARCGS, a fast and highly accurate short-read assembly algorithm for de novo genomic sequencing. *Genome Res* 2007, 17:1697–1706.
263. Jeck WR, Reinhardt JA, Baltrus DA, Hickenbotham MT, Magrini V, Mardis ER, Dangl JL, Jones CD. Extending assembly of short DNA sequences to handle error. *Bioinformatics* 2007, 23:2942–2944.

264. Miller JR, Delcher AL, Koren S, Venter E, Walenz BP, Brownley A, Johnson J, Li K, Mobarry C, Sutton G. Aggressive assembly of pyrosequencing reads with mates. *Bioinformatics* 2008, 24:2818–2824.
265. Hernandez D, Francois P, Farinelli L, Osteras M, Schrenzel J. De novo bacterial genome sequencing: Millions of very short reads assembled on a desktop computer. *Genome Res* 2008, 18:802–809.
266. Li H. Exploring single-sample SNP and INDEL calling with whole-genome de novo assembly. *Bioinformatics* 2012, 28:1838–1844.
267. Diguistini S, Liao NY, Platt D, Robertson G, Seidel M, Chan SK, Docking TR et al. De novo genome sequence assembly of a filamentous fungus using Sanger, 454 and Illumina sequence data. *Genome Biol* 2009, 10:R94.
268. Myers EW. Toward simplifying and accurately formulating fragment assembly. *J Comput Biol* 1995, 2:275–290.
269. Gonnella G, Kurtz S. Readjoiner: A fast and memory efficient string graph-based sequence assembler. *BMC Bioinformatics* 2012, 13:82.
270. Butler J, MacCallum I, Kleber M, Shlyakhter IA, Belmonte MK, Lander ES, Nusbaum C, Jaffe DB. ALLPATHS: De novo assembly of whole-genome shotgun microreads. *Genome Res* 2008, 18:810–820.
271. Maccallum I, Przybylski D, Gnerre S, Burton J, Shlyakhter I, Gnirke A, Malek J et al. ALLPATHS 2: Small genomes assembled accurately and with high continuity from short paired reads. *Genome Biol* 2009, 10:R103.
272. Peng Y, Leung HC, Yiu S-M, Chin FY. IDBA—A practical iterative de Bruijn graph de novo assembler. In *Research in Computational Molecular Biology*, Springer; 2010: 426–440.
273. Luo R, Liu B, Xie Y, Li Z, Huang W, Yuan J, He G et al. SOAPdenovo2: An empirically improved memory-efficient short-read de novo assembler. *Gigascience* 2012, 1:18.
274. Ye C, Ma ZS, Cannon CH, Pop M, Yu DW. Exploiting sparseness in de novo genome assembly. *BMC Bioinformatics* 2012, 13 Suppl 6:S1.
275. Zimin AV, Marcais G, Puiu D, Roberts M, Salzberg SL, Yorke JA. The MaSuRCA genome assembler. *Bioinformatics* 2013, 29:2669–2677.
276. Pop M, Kosack DS, Salzberg SL. Hierarchical scaffolding with Bambus. *Genome Res* 2004, 14:149–159.
277. Koren S, Treangen TJ, Pop M. Bambus 2: Scaffolding metagenomes. *Bioinformatics* 2011, 27:2964–2971.
278. Gao S, Sung WK, Nagarajan N. Opera: Reconstructing optimal genomic scaffolds with high-throughput paired-end sequences. *J Comput Biol* 2011, 18:1681–1691.
279. Dayarian A, Michael TP, Sengupta AM. SOPRA: Scaffolding algorithm for paired reads via statistical optimization. *BMC Bioinformatics* 2010, 11:345.
280. Boetzer M, Henkel CV, Jansen HJ, Butler D, Pirovano W. Scaffolding preassembled contigs using SSPACE. *Bioinformatics* 2011, 27:578–579.
281. Hunt M, Newbold C, Berriman M, Otto TD. A comprehensive evaluation of assembly scaffolding tools. *Genome Biol* 2014, 15:R42.
282. Gurevich A, Saveliev V, Vyahhi N, Tesler G. QUAST: Quality assessment tool for genome assemblies. *Bioinformatics* 2013, 29:1072–1075.
283. Tsai IJ, Otto TD, Berriman M. Improving draft assemblies by iterative mapping and assembly of short reads to eliminate gaps. *Genome Biol* 2010, 11:R41.

284. Boetzer M, Pirovano W. Toward almost closed genomes with GapFiller. *Genome Biol* 2012, 13:R56.
285. Bao E, Jiang T, Girke T. AlignGraph: Algorithm for secondary de novo genome assembly guided by closely related references. *Bioinformatics* 2014, 30:i319–i328.
286. Kim J, Larkin DM, Cai Q, Asan, Zhang Y, Ge RL, Auvil L et al. Reference-assisted chromosome assembly. *Proc Natl Acad Sci USA* 2013, 110:1785–1790.
287. Chen Y, Negre N, Li Q, Mieczkowska JO, Slattery M, Liu T, Zhang Y et al. Systematic evaluation of factors influencing ChIP-seq fidelity. *Nat Methods* 2012, 9:609–614.
288. Daley T, Smith AD. Predicting the molecular complexity of sequencing libraries. *Nat Methods* 2013, 10:325–327.
289. Diaz A, Nellore A, Song JS. CHANCE: Comprehensive software for quality control and validation of ChIP-seq data. *Genome Biol* 2012, 13:R98.
290. Zhao X, Sandelin A. GMD: Measuring the distance between histograms with applications on high-throughput sequencing reads. *Bioinformatics* 2012, 28:1164–1165.
291. Diaz A, Park K, Lim DA, Song JS. Normalization, bias correction, and peak calling for ChIP-seq. *Stat Appl Genet Mol Biol* 2012, 11:Article 9.
292. Rozowsky J, Euskirchen G, Auerbach RK, Zhang ZD, Gibson T, Bjornson R, Carriero N, Snyder M, Gerstein MB. PeakSeq enables systematic scoring of ChIP-seq experiments relative to controls. *Nat Biotechnol* 2009, 27:66–75.
293. Zhang Y, Liu T, Meyer CA, Eeckhoute J, Johnson DS, Bernstein BE, Nusbaum C et al. Model-based analysis of ChIP-Seq (MACS). *Genome Biol* 2008, 9:R137.
294. Ji H, Jiang H, Ma W, Johnson DS, Myers RM, Wong WH. An integrated software system for analyzing ChIP-chip and ChIP-seq data. *Nat Biotechnol* 2008, 26:1293–1300.
295. Shen L, Shao NY, Liu X, Maze I, Feng J, Nestler EJ. diffReps: Detecting differential chromatin modification sites from ChIP-seq data with biological replicates. *PLoS One* 2013, 8:e65598.
296. Manser P, Reimers M. A simple scaling normalization for comparing ChIP-Seq samples. *PeerJ PrePrints* 2014, 1.
297. Shao Z, Zhang Y, Yuan GC, Orkin SH, Waxman DJ. MAnorm: A robust model for quantitative comparison of ChIP-Seq data sets. *Genome Biol* 2012, 13:R16.
298. Nair NU, Sahu AD, Bucher P, Moret BM. ChIPnorm: A statistical method for normalizing and identifying differential regions in histone modification ChIP-seq libraries. *PLoS One* 2012, 7:e39573.
299. Taslim C, Wu J, Yan P, Singer G, Parvin J, Huang T, Lin S, Huang K. Comparative study on ChIP-seq data: Normalization and binding pattern characterization. *Bioinformatics* 2009, 25:2334–2340.
300. Kharchenko PV, Tolstorukov MY, Park PJ. Design and analysis of ChIP-seq experiments for DNA-binding proteins. *Nat Biotechnol* 2008, 26:1351–1359.
301. Landt SG, Marinov GK, Kundaje A, Kheradpour P, Pauli F, Batzoglou S, Bernstein BE et al. ChIP-seq guidelines and practices of the ENCODE and modENCODE consortia. *Genome Res* 2012, 22:1813–1831.
302. Heinz S, Benner C, Spann N, Bertolino E, Lin YC, Laslo P, Cheng JX, Murre C, Singh H, Glass CK. Simple combinations of lineage-determining transcription factors prime cis-regulatory elements required for macrophage and B cell identities. *Mol Cell* 2010, 38:576–589.

303. Xu H, Handoko L, Wei X, Ye C, Sheng J, Wei CL, Lin F, Sung WK. A signal-noise model for significance analysis of ChIP-seq with negative control. *Bioinformatics* 2010, 26:1199–1204.
304. Mortazavi A, Williams BA, McCue K, Schaeffer L, Wold B. Mapping and quantifying mammalian transcriptomes by RNA-Seq. *Nat Methods* 2008, 5:621–628.
305. Boyle AP, Guinney J, Crawford GE, Furey TS. F-Seq: A feature density estimator for high-throughput sequence tags. *Bioinformatics* 2008, 24:2537–2538.
306. Tuteja G, White P, Schug J, Kaestner KH. Extracting transcription factor targets from ChIP-Seq data. *Nucleic Acids Res* 2009, 37:e113.
307. Feng X, Grossman R, Stein L. PeakRanger: A cloud-enabled peak caller for ChIP-seq data. *BMC Bioinformatics* 2011, 12:139.
308. Valouev A, Johnson DS, Sundquist A, Medina C, Anton E, Batzoglou S, Myers RM, Sidow A. Genome-wide analysis of transcription factor binding sites based on ChIP-Seq data. *Nat Methods* 2008, 5:829–834.
309. Song Q, Smith AD. Identifying dispersed epigenomic domains from ChIP-Seq data. *Bioinformatics* 2011, 27:870–871.
310. Zang C, Schones DE, Zeng C, Cui K, Zhao K, Peng W. A clustering approach for identification of enriched domains from histone modification ChIP-Seq data. *Bioinformatics* 2009, 25:1952–1958.
311. Jothi R, Cuddapah S, Barski A, Cui K, Zhao K. Genome-wide identification of in vivo protein-DNA binding sites from ChIP-Seq data. *Nucleic Acids Res* 2008, 36:5221–5231.
312. Nix DA, Courdy SJ, Boucher KM. Empirical methods for controlling false positives and estimating confidence in ChIP-Seq peaks. *BMC Bioinformatics* 2008, 9:523.
313. Rashid NU, Giresi PG, Ibrahim JG, Sun W, Lieb JD. ZINBA integrates local covariates with DNA-seq data to identify broad and narrow regions of enrichment, even within amplified genomic regions. *Genome Biol* 2011, 12:R67.
314. Shin H, Liu T, Manrai AK, Liu XS. CEAS: cis-Regulatory element annotation system. *Bioinformatics* 2009, 25:2605–2606.
315. Zhu LJ, Gazin C, Lawson ND, Pages H, Lin SM, Lapointe DS, Green MR. ChIPpeakAnno: A Bioconductor package to annotate ChIP-seq and ChIP-chip data. *BMC Bioinformatics* 2010, 11:237.
316. Chen X, Xu H, Yuan P, Fang F, Huss M, Vega VB, Wong E et al. Integration of external signaling pathways with the core transcriptional network in embryonic stem cells. *Cell* 2008, 133:1106–1117.
317. Chen L, Wang C, Qin ZS, Wu H. A novel statistical method for quantitative comparison of multiple ChIP-seq datasets. *Bioinformatics* 2015, 31:1889–1896.
318. Xu H, Wei CL, Lin F, Sung WK. An HMM approach to genome-wide identification of differential histone modification sites from ChIP-seq data. *Bioinformatics* 2008, 24:2344–2349.
319. Hon G, Ren B, Wang W. ChromaSig: A probabilistic approach to finding common chromatin signatures in the human genome. *PLoS Comput Biol* 2008, 4:e1000201.
320. Liang K, Keles S. Detecting differential binding of transcription factors with ChIP-seq. *Bioinformatics* 2012, 28:121–122.
321. Stark R, Brown G. DiffBind: Differential binding analysis of ChIP-Seq peak data. In R package version 2011, 100.

322. Taslim C, Huang T, Lin S. DIME: R-package for identifying differential ChIP-seq based on an ensemble of mixture models. *Bioinformatics* 2011, 27:1569–1570.
323. Schweikert G, Cseke B, Clouaire T, Bird A, Sanguinetti G. MMDiff: Quantitative testing for shape changes in ChIP-Seq data sets. *BMC Genomics* 2013, 14:826.
324. Welch RP, Lee C, Imbriano PM, Patil S, Weymouth TE, Smith RA, Scott LJ, Sartor MA. ChIP-Enrich: Gene set enrichment testing for ChIP-seq data. *Nucleic Acids Res* 2014. doi: 10.1093/nar/gku463.
325. Liu T, Ortiz JA, Taing L, Meyer CA, Lee B, Zhang Y, Shin H et al. *Cistrome: An integrative platform for transcriptional regulation studies. Genome Biol* 2011, 12:R83.
326. Machanick P, Bailey TL. MEME-ChIP: Motif analysis of large DNA datasets. *Bioinformatics* 2011, 27:1696–1697.
327. Thomas-Chollier M, Herrmann C, Defrance M, Sand O, Thieffry D, van Helden J. *RSAT peak-motifs*: Motif analysis in full-size ChIP-seq datasets. *Nucleic Acids Res* 2012, 40:e31.
328. Kulakovskiy IV, Boeva VA, Favorov AV, Makeev VJ. Deep and wide digging for binding motifs in ChIP-Seq data. *Bioinformatics* 2010, 26:2622–2623.
329. Mahony S, Benos PV. STAMP: A web tool for exploring DNA-binding motif similarities. *Nucleic Acids Res* 2007, 35:W253–258.
330. Gupta S, Stamatoyannopoulos JA, Bailey TL, Noble WS. Quantifying similarity between motifs. *Genome Biol* 2007, 8:R24.
331. Bailey TL, Machanick P. Inferring direct DNA binding from ChIP-seq. *Nucleic Acids Res* 2012, 40:e128.
332. Grant CE, Bailey TL, Noble WS. FIMO: Scanning for occurrences of a given motif. *Bioinformatics* 2011, 27:1017–1018.
333. Ernst J, Kellis M. Discovery and characterization of chromatin states for systematic annotation of the human genome. *Nat Biotechnol* 2010, 28:817–825.
334. Klein HU, Schafer M, Porse BT, Hasemann MS, Ickstadt K, Dugas M. Integrative analysis of histone ChIP-seq and transcription data using Bayesian mixture models. *Bioinformatics* 2014 (Epub ahead of print).
335. Standards and Guidelines for Whole Genome Shotgun Bisulfite Sequencing, http://www.roadmapepigenomics.org/protocols.
336. Ziller MJ, Hansen KD, Meissner A, Aryee MJ. Coverage recommendations for methylation analysis by whole-genome bisulfite sequencing. *Nat Methods* 2015, 12(3):230–232.
337. Meissner A, Gnirke A, Bell GW, Ramsahoye B, Lander ES, Jaenisch R. Reduced representation bisulfite sequencing for comparative high-resolution DNA methylation analysis. *Nucleic Acids Res* 2005, 33:5868–5877.
338. Nautiyal S, Carlton VE, Lu Y, Ireland JS, Flaucher D, Moorhead M, Gray JW et al. High-throughput method for analyzing methylation of CpGs in targeted genomic regions. *Proc Natl Acad Sci USA* 2010, 107:12587–12592.
339. Varley KE, Mitra RD. Bisulfite Patch PCR enables multiplexed sequencing of promoter methylation across cancer samples. *Genome Res* 2010, 20:1279–1287.
340. Deng J, Shoemaker R, Xie B, Gore A, LeProust EM, Antosiewicz-Bourget J, Egli D et al. Targeted bisulfite sequencing reveals changes in DNA methylation associated with nuclear reprogramming. *Nat Biotechnol* 2009, 27:353–360.
341. Ivanov M, Kals M, Kacevska M, Metspalu A, Ingelman-Sundberg M, Milani L. In-solution hybrid capture of bisulfite-converted DNA for targeted bisulfite sequencing of 174 ADME genes. *Nucleic Acids Res* 2013, 41:e72.

342. Harris RA, Wang T, Coarfa C, Nagarajan RP, Hong C, Downey SL, Johnson BE et al. Comparison of sequencing-based methods to profile DNA methylation and identification of monoallelic epigenetic modifications. *Nat Biotechnol* 2010, 28:1097–1105.
343. Nair SS, Coolen MW, Stirzaker C, Song JZ, Statham AL, Strbenac D, Robinson MD, Clark SJ. Comparison of methyl-DNA immunoprecipitation (MeDIP) and methyl-CpG binding domain (MBD) protein capture for genome-wide DNA methylation analysis reveal CpG sequence coverage bias. *Epigenetics* 2011, 6:34–44.
344. Booth MJ, Branco MR, Ficz G, Oxley D, Krueger F, Reik W, Balasubramanian S. Quantitative sequencing of 5-methylcytosine and 5-hydroxymethylcytosine at single-base resolution. *Science* 2012, 336:934–937.
345. Flusberg BA, Webster DR, Lee JH, Travers KJ, Olivares EC, Clark TA, Korlach J, Turner SW. Direct detection of DNA methylation during single-molecule, real-time sequencing. *Nat Methods* 2010, 7:461–465.
346. Laszlo AH, Derrington IM, Brinkerhoff H, Langford KW, Nova IC, Samson JM, Bartlett JJ, Pavlenok M, Gundlach JH. Detection and mapping of 5-methylcytosine and 5-hydroxymethylcytosine with nanopore MspA. *Proc Natl Acad Sci USA* 2013, 110:18904–18909.
347. Schreiber J, Wescoe ZL, Abu-Shumays R, Vivian JT, Baatar B, Karplus K, Akeson M. Error rates for nanopore discrimination among cytosine, methylcytosine, and hydroxymethylcytosine along individual DNA strands. *Proc Natl Acad Sci USA* 2013, 110:18910–18915.
348. Trim Galore!, http://www.bioinformatics.babraham.ac.uk/projects/trim_galore/.
349. Hansen KD, Langmead B, Irizarry RA. BSmooth: From whole genome bisulfite sequencing reads to differentially methylated regions. *Genome Biol* 2012, 13:R83.
350. Liang F, Tang B, Wang Y, Wang J, Yu C, Chen X, Zhu J, Yan J, Zhao W, Li R. WBSA: Web service for bisulfite sequencing data analysis. *PLoS One* 2014, 9:e86707.
351. Xi Y, Li W. BSMAP: Whole genome bisulfite sequence MAPping program. *BMC Bioinformatics* 2009, 10:232.
352. Coarfa C, Yu F, Miller CA, Chen Z, Harris RA, Milosavljevic A. Pash 3.0: A versatile software package for read mapping and integrative analysis of genomic and epigenomic variation using massively parallel DNA sequencing. *BMC Bioinformatics* 2010, 11:572.
353. Xi Y, Bock C, Muller F, Sun D, Meissner A, Li W. RRBSMAP: A fast, accurate and user-friendly alignment tool for reduced representation bisulfite sequencing. *Bioinformatics* 2012, 28:430–432.
354. Lim JQ, Tennakoon C, Li G, Wong E, Ruan Y, Wei CL, Sung WK. BatMeth: Improved mapper for bisulfite sequencing reads on DNA methylation. *Genome Biol* 2012, 13:R82.
355. Krueger F, Andrews SR. Bismark: A flexible aligner and methylation caller for Bisulfite-Seq applications. *Bioinformatics* 2011, 27:1571–1572.
356. Harris EY, Ponts N, Le Roch KG, Lonardi S. BRAT-BW: Efficient and accurate mapping of bisulfite-treated reads. *Bioinformatics* 2012, 28:1795–1796.
357. Chen PY, Cokus SJ, Pellegrini M. BS Seeker: Precise mapping for bisulfite sequencing. *BMC Bioinformatics* 2010, 11:203.

358. Guo W, Fiziev P, Yan W, Cokus S, Sun X, Zhang MQ, Chen PY, Pellegrini M. BS-Seeker2: A versatile aligning pipeline for bisulfite sequencing data. *BMC Genomics* 2013, 14:774.
359. Pedersen B, Hsieh TF, Ibarra C, Fischer RL. MethylCoder: Software pipeline for bisulfite-treated sequences. *Bioinformatics* 2011, 27:2435–2436.
360. Kunde-Ramamoorthy G, Coarfa C, Laritsky E, Kessler NJ, Harris RA, Xu M, Chen R, Shen L, Milosavljevic A, Waterland RA. Comparison and quantitative verification of mapping algorithms for whole-genome bisulfite sequencing. *Nucleic Acids Res* 2014, 42:e43.
361. Lin X, Sun D, Rodriguez B, Zhao Q, Sun H, Zhang Y, Li W. BSeQC: Quality control of bisulfite sequencing experiments. *Bioinformatics* 2013, 29:3227–3229.
362. Benoukraf T, Wongphayak S, Hadi LH, Wu M, Soong R. GBSA: A comprehensive software for analysing whole genome bisulfite sequencing data. *Nucleic Acids Res* 2013, 41:e55.
363. Akalin A, Kormaksson M, Li S, Garrett-Bakelman FE, Figueroa ME, Melnick A, Mason CE. methylKit: A comprehensive R package for the analysis of genome-wide DNA methylation profiles. *Genome Biol* 2012, 13:R87.
364. Washington University EpiGenome Browser, http://epigenomegateway.wustl.edu/browser/.
365. Kent WJ, Zweig AS, Barber G, Hinrichs AS, Karolchik D. BigWig and BigBed: Enabling browsing of large distributed datasets. *Bioinformatics* 2010, 26:2204–2207.
366. Dorff KC, Chambwe N, Zeno Z, Simi M, Shaknovich R, Campagne F. GobyWeb: Simplified management and analysis of gene expression and DNA methylation sequencing data. *PLoS One* 2013, 8:e69666.
367. Jiang P, Sun K, Lun FM, Guo AM, Wang H, Chan KC, Chiu RW, Lo YM, Sun H. Methy-Pipe: An integrated bioinformatics pipeline for whole genome bisulfite sequencing data analysis. *PLoS One* 2014, 9:e100360.
368. Sun D, Xi Y, Rodriguez B, Park HJ, Tong P, Meong M, Goodell MA, Li W. MOABS: Model based analysis of bisulfite sequencing data. *Genome Biol* 2014, 15:R38.
369. Zhang Y, Liu H, Lv J, Xiao X, Zhu J, Liu X, Su J et al. QDMR: A quantitative method for identification of differentially methylated regions by entropy. *Nucleic Acids Res* 2011, 39:e58.
370. Statham AL, Strbenac D, Coolen MW, Stirzaker C, Clark SJ, Robinson MD. Repitools: An R package for the analysis of enrichment-based epigenomic data. *Bioinformatics* 2010, 26:1662–1663.
371. Halachev K, Bast H, Albrecht F, Lengauer T, Bock C. EpiExplorer: Live exploration and global analysis of large epigenomic datasets. *Genome Biol* 2012, 13:R96.
372. McLean CY, Bristor D, Hiller M, Clarke SL, Schaar BT, Lowe CB, Wenger AM, Bejerano G. GREAT improves functional interpretation of cis-regulatory regions. *Nat Biotechnol* 2010, 28:495–501.
373. Schloss PD, Handelsman J. Metagenomics for studying unculturable micro-organisms: Cutting the Gordian knot. *Genome Biol* 2005, 6:229.
374. Yarza P, Ludwig W, Euzeby J, Amann R, Schleifer KH, Glockner FO, Rossello-Mora R. Update of the All-Species Living Tree Project based on 16S and 23S rRNA sequence analyses. *Syst Appl Microbiol* 2010, 33:291–299.
375. Delmont TO, Robe P, Clark I, Simonet P, Vogel TM. Metagenomic comparison of direct and indirect soil DNA extraction approaches. *J Microbiol Methods* 2011, 86:397–400.

376. BMTagger, http://biowulf.nih.gov/apps/bmtagger.html.
377. Schmieder R, Edwards R. Fast identification and removal of sequence contamination from genomic and metagenomic datasets. *PLoS One* 2011, 6:e17288.
378. Zhou J, Sun D, Childers A, McDermott TR, Wang Y, Liles MR. Three novel virophage genomes discovered from yellowstone lake metagenomes. *J Virol* 2015, 89:1278–1285.
379. Beja O, Aravind L, Koonin EV, Suzuki MT, Hadd A, Nguyen LP, Jovanovich SB et al. Bacterial rhodopsin: Evidence for a new type of phototrophy in the sea. *Science* 2000, 289:1902–1906.
380. Rho M, Wu YW, Tang H, Doak TG, Ye Y. Diverse CRISPRs evolving in human microbiomes. *PLoS Genet* 2012, 8:e1002441.
381. Lai B, Ding R, Li Y, Duan L, Zhu H. A de novo metagenomic assembly program for shotgun DNA reads. *Bioinformatics* 2012, 28:1455–1462.
382. Laserson J, Jojic V, Koller D. Genovo: De novo assembly for metagenomes. *J Comput Biol* 2011, 18:429–443.
383. Afiahayati, Sato K, Sakakibara Y. An extended genovo metagenomic assembler by incorporating paired-end information. *PeerJ* 2013, 1:e196.
384. Namiki T, Hachiya T, Tanaka H, Sakakibara Y. MetaVelvet: An extension of Velvet assembler to de novo metagenome assembly from short sequence reads. *Nucleic Acids Res* 2012, 40:e155.
385. Afiahayati, Sato K, Sakakibara Y. MetaVelvet-SL: An extension of the Velvet assembler to a de novo metagenomic assembler utilizing supervised learning. *DNA Res* 2014. doi: 10.1093/dnares/dsu041.
386. Peng Y, Leung HC, Yiu SM, Chin FY. Meta-IDBA: A de Novo assembler for metagenomic data. *Bioinformatics* 2011, 27:i94–101.
387. Peng Y, Leung HC, Yiu SM, Chin FY. IDBA-UD: A de novo assembler for single-cell and metagenomic sequencing data with highly uneven depth. *Bioinformatics* 2012, 28:1420–1428.
388. Wu YW, Rho M, Doak TG, Ye Y. Stitching gene fragments with a network matching algorithm improves gene assembly for metagenomics. *Bioinformatics* 2012, 28:i363–i369.
389. Boisvert S, Raymond F, Godzaridis E, Laviolette F, Corbeil J. Ray Meta: Scalable de novo metagenome assembly and profiling. *Genome Biol* 2012, 13:R122.
390. Haider B, Ahn TH, Bushnell B, Chai J, Copeland A, Pan C. Omega: An overlap-graph de novo assembler for metagenomics. *Bioinformatics* 2014, 30:2717–2722.
391. McHardy AC, Martin HG, Tsirigos A, Hugenholtz P, Rigoutsos I. Accurate phylogenetic classification of variable-length DNA fragments. *Nat Methods* 2007, 4:63–72.
392. Patil KR, Roune L, McHardy AC. The PhyloPythiaS web server for taxonomic assignment of metagenome sequences. *PLoS One* 2012, 7:e38581.
393. Teeling H, Waldmann J, Lombardot T, Bauer M, Glockner FO. TETRA: A web-service and a stand-alone program for the analysis and comparison of tetranucleotide usage patterns in DNA sequences. *BMC Bioinformatics* 2004, 5:163.
394. Diaz NN, Krause L, Goesmann A, Niehaus K, Nattkemper TW. TACOA: Taxonomic classification of environmental genomic fragments using a kernelized nearest neighbor approach. *BMC Bioinformatics* 2009, 10:56.
395. Brady A, Salzberg SL. Phymm and PhymmBL: Metagenomic phylogenetic classification with interpolated Markov models. *Nat Methods* 2009, 6:673–676.

396. Chan CK, Hsu AL, Halgamuge SK, Tang SL. Binning sequences using very sparse labels within a metagenome. *BMC Bioinformatics* 2008, 9:215.
397. Zheng H, Wu H. Short prokaryotic DNA fragment binning using a hierarchical classifier based on linear discriminant analysis and principal component analysis. *J Bioinform Comput Biol* 2010, 8:995–1011.
398. Huson DH, Mitra S, Ruscheweyh HJ, Weber N, Schuster SC. Integrative analysis of environmental sequences using MEGAN4. *Genome Res* 2011, 21:1552–1560.
399. Huson DH, Auch AF, Qi J, Schuster SC. MEGAN analysis of metagenomic data. *Genome Res* 2007, 17:377–386.
400. Gerlach W, Junemann S, Tille F, Goesmann A, Stoye J. WebCARMA: A web application for the functional and taxonomic classification of unassembled metagenomic reads. *BMC Bioinformatics* 2009, 10:430.
401. Gerlach W, Stoye J. Taxonomic classification of metagenomic shotgun sequences with CARMA3. *Nucleic Acids Res* 2011, 39:e91.
402. Monzoorul Haque M, Ghosh TS, Komanduri D, Mande SS. SOrt-ITEMS: Sequence orthology based approach for improved taxonomic estimation of metagenomic sequences. *Bioinformatics* 2009, 25:1722–1730.
403. Liu B, Gibbons T, Ghodsi M, Treangen T, Pop M. Accurate and fast estimation of taxonomic profiles from metagenomic shotgun sequences. *BMC Genomics* 2011, 12 Suppl 2:S4.
404. Leung HC, Yiu SM, Yang B, Peng Y, Wang Y, Liu Z, Chen J, Qin J, Li R, Chin FY. A robust and accurate binning algorithm for metagenomic sequences with arbitrary species abundance ratio. *Bioinformatics* 2011, 27:1489–1495.
405. Mohammed MH, Ghosh TS, Singh NK, Mande SS. SPHINX—An algorithm for taxonomic binning of metagenomic sequences. *Bioinformatics* 2011, 27:22–30.
406. Rusch DB, Halpern AL, Sutton G, Heidelberg KB, Williamson S, Yooseph S, Wu D et al. The Sorcerer II Global Ocean Sampling expedition: Northwest Atlantic through eastern tropical Pacific. *PLoS Biol* 2007, 5:e77.
407. Davenport CF, Neugebauer J, Beckmann N, Friedrich B, Kameri B, Kokott S, Paetow M et al. Genometa—A fast and accurate classifier for short metagenomic shotgun reads. *PLoS One* 2012, 7:e41224.
408. Niu B, Zhu Z, Fu L, Wu S, Li W. FR-HIT, a very fast program to recruit metagenomic reads to homologous reference genomes. *Bioinformatics* 2011, 27:1704–1705.
409. Rho M, Tang H, Ye Y. FragGeneScan: Predicting genes in short and error-prone reads. *Nucleic Acids Res* 2010, 38:e191.
410. Zhu W, Lomsadze A, Borodovsky M. Ab initio gene identification in metagenomic sequences. *Nucleic Acids Res* 2010, 38:e132.
411. Noguchi H, Taniguchi T, Itoh T. MetaGeneAnnotator: Detecting species-specific patterns of ribosomal binding site for precise gene prediction in anonymous prokaryotic and phage genomes. *DNA Res* 2008, 15:387–396.
412. Hoff KJ, Lingner T, Meinicke P, Tech M. Orphelia: Predicting genes in metagenomic sequencing reads. *Nucleic Acids Res* 2009, 37:W101–105.
413. Lowe TM, Eddy SR. tRNAscan-SE: A program for improved detection of transfer RNA genes in genomic sequence. *Nucleic Acids Res* 1997, 25:955–964.
414. Grissa I, Vergnaud G, Pourcel C. CRISPRFinder: A web tool to identify clustered regularly interspaced short palindromic repeats. *Nucleic Acids Res* 2007, 35:W52–57.
415. Segata N, Waldron L, Ballarini A, Narasimhan V, Jousson O, Huttenhower C. Metagenomic microbial community profiling using unique clade-specific marker genes. *Nat Methods* 2012, 9:811–814.

416. Wu M, Eisen JA. A simple, fast, and accurate method of phylogenomic inference. *Genome Biol* 2008, 9:R151.
417. Kerepesi C, Banky D, Grolmusz V. AmphoraNet: The webserver implementation of the AMPHORA2 metagenomic workflow suite. *Gene* 2014, 533:538–540.
418. Wu M, Scott AJ. Phylogenomic analysis of bacterial and archaeal sequences with AMPHORA2. *Bioinformatics* 2012, 28:1033–1034.
419. Sharpton TJ, Riesenfeld SJ, Kembel SW, Ladau J, O'Dwyer JP, Green JL, Eisen JA, Pollard KS. PhylOTU: A high-throughput procedure quantifies microbial community diversity and resolves novel taxa from metagenomic data. *PLoS Comput Biol* 2011, 7:e1001061.
420. Darling AE, Jospin G, Lowe E, Matsen FA, Bik HM, Eisen JA. PhyloSift: Phylogenetic analysis of genomes and metagenomes. *PeerJ* 2014, 2:e243.
421. Li W. Analysis and comparison of very large metagenomes with fast clustering and functional annotation. *BMC Bioinformatics* 2009, 10:359.
422. Arumugam M, Harrington ED, Foerstner KU, Raes J, Bork P. SmashCommunity: A metagenomic annotation and analysis tool. *Bioinformatics* 2010, 26:2977–2978.
423. Treangen TJ, Koren S, Sommer DD, Liu B, Astrovskaya I, Ondov B, Darling AE, Phillippy AM, Pop M. MetAMOS: A modular and open source metagenomic assembly and analysis pipeline. *Genome Biol* 2013, 14:R2.
424. Overbeek R, Begley T, Butler RM, Choudhuri JV, Chuang HY, Cohoon M, de Crecy-Lagard V et al. The subsystems approach to genome annotation and its use in the project to annotate 1000 genomes. *Nucleic Acids Res* 2005, 33:5691–5702.
425. Franzosa EA, Morgan XC, Segata N, Waldron L, Reyes J, Earl AM, Giannoukos G et al. Relating the metatranscriptome and metagenome of the human gut. *Proc Natl Acad Sci USA* 2014, 111:E2329–2338.
426. Abubucker S, Segata N, Goll J, Schubert AM, Izard J, Cantarel BL, Rodriguez-Mueller B et al. Metabolic reconstruction for metagenomic data and its application to the human microbiome. *PLoS Comput Biol* 2012, 8:e1002358.
427. Ye Y, Doak TG. A parsimony approach to biological pathway reconstruction/inference for genomes and metagenomes. *PLoS Comput Biol* 2009, 5:e1000465.
428. Liu B, Pop M. MetaPath: Identifying differentially abundant metabolic pathways in metagenomic datasets. *BMC Proc* 2011, 5 Suppl 2:S9.
429. Paulson JN, Stine OC, Bravo HC, Pop M. Differential abundance analysis for microbial marker-gene surveys. *Nat Methods* 2013, 10:1200–1202.
430. Segata N, Izard J, Waldron L, Gevers D, Miropolsky L, Garrett WS, Huttenhower C. Metagenomic biomarker discovery and explanation. *Genome Biol* 2011, 12:R60.
431. White JR, Nagarajan N, Pop M. Statistical methods for detecting differentially abundant features in clinical metagenomic samples. *PLoS Comput Biol* 2009, 5:e1000352.
432. Parks DH, Tyson GW, Hugenholtz P, Beiko RG. STAMP: Statistical analysis of taxonomic and functional profiles. *Bioinformatics* 2014, 30:3123–3124.
433. Rodriguez-Brito B, Rohwer F, Edwards RA. An application of statistics to comparative metagenomics. *BMC Bioinformatics* 2006, 7:162.
434. Jain M, Fiddes IT, Miga KH, Olsen HE, Paten B, Akeson M. Improved data analysis for the MinION nanopore sequencer. *Nat Methods* 2015, 12:351–356.
435. Ohshiro T, Matsubara K, Tsutsui M, Furuhashi M, Taniguchi M, Kawai T. Single-molecule electrical random resequencing of DNA and RNA. *Sci Rep* 2012, 2:501.

436. Chin CS, Alexander DH, Marks P, Klammer AA, Drake J, Heiner C, Clum A et al. Nonhybrid, finished microbial genome assemblies from long-read SMRT sequencing data. *Nat Methods* 2013, 10:563–569.
437. FALCON, https://github.com/PacificBiosciences/falcon.
438. DePristo MA, Banks E, Poplin R, Garimella KV, Maguire JR, Hartl C, Philippakis AA et al. A framework for variation discovery and genotyping using next-generation DNA sequencing data. *Nat Genet* 2011, 43:491–498.
439. Reid JG, Carroll A, Veeraraghavan N, Dahdouli M, Sundquist A, English A, Bainbridge M et al. Launching genomics into the cloud: Deployment of Mercury, a next generation sequence analysis pipeline. *BMC Bioinformatics* 2014, 15:30.
440. Zhao S, Prenger K, Smith L, Messina T, Fan H, Jaeger E, Stephens S. Rainbow: A tool for large-scale whole-genome sequencing data analysis using cloud computing. *BMC Genomics* 2013, 14:425.
441. Schadt EE, Linderman MD, Sorenson J, Lee L, Nolan GP. Computational solutions to large-scale data management and analysis. *Nat Rev Genet* 2010, 11:647–657.

Index

F

G