OXFORD

Briefings in Bioinformatics, 17(5), 2016, 786–795

doi: 10.1093/bib/bbv088
Advance Access Publication Date: 5 October 2015
Paper

A Comparison of Base-calling Algorithms for Illumina Sequencing Technology

Ashley Cacho, Ekaterina Smirnova, Snehalata Huzurbazar and Xinping Cui

Corresponding author: Xinping Cui, Department of Statistics and Institute for Integrative Genome Biology, University of California, Riverside.
Tel: +1 951 827 2563; Fax: +1 951 827 3286; E-mail: xinping.cui@ucr.edu

Abstract

Recent advances in next-generation sequencing technology have yielded increasing cost-effectiveness and higher throughput produced per run, in turn, greatly influencing the analysis of DNA sequences. Among the various sequencing technologies, Illumina is by far the most widely used platform. However, the Illumina sequencing platform suffers from several imperfections that can be attributed to the chemical processes inherent to the sequencing-by-synthesis technology. With the enormous amounts of reads produced, statistical methodologies and computationally efficient algorithms are required to improve the accuracy and speed of base-calling. Over the past few years, several papers have proposed methods to model the various imperfections, giving rise to accurate and/or efficient base-calling algorithms. In this article, we provide a comprehensive comparison of the performance of recently developed base-callers and we present a general statistical model that unifies a large majority of these base-callers.

Key words: base-calling; Illumina; next-generation sequencing

Introduction

Nearly four decades ago, the first widely used sequencing method was developed by Frederick Sanger with a group of colleagues [1]. A few years following the development of the Sanger sequencing method, the first genome, that of bacteriophage φX174, was sequenced completely. The Sanger method dominated for the next two decades, producing DNA sequences of up to ~1000 base pairs in length; however, this 'first-generation' technology was costly and slow. Over the past decade, breakthroughs in alternative sequencing technologies have greatly influenced the analysis of DNA sequences. This remarkable technological advancement is opening up a new era for the biological sciences and related fields. If the cost of whole genome sequencing can be pushed down to approximately $1000, DNA sequencing may become part of routine medical diagnostics [2]; the positive implications of which are endless. However, to bring this promise of whole genome sequencing to fruition, several immediate challenges pertinent to data acquisition and analysis must be overcome. First and foremost, it is important to obtain accurate base-calls for high-throughput sequences. Over the past few years, there has been a growing interest in base-calling because it will have direct and immediate impact on assembly, polymorphism detection and downstream analyses. Table 1 shows a list of available base-callers along with some practical notes. In this article, we introduce a general model with various base-callers as special cases and provide a comparison of the base-calling algorithms using the bacteriophage φX174 and human genomes.

A review paper by Ledergerber and Dessimoz [17] summarizes some of the base-calling algorithms before 2011. The paper performed a comparison of the Alta-Cyclic, Rolexa, Ibis, BayesCall, naiveBayesCall and Bustard base-callers using metrics of overall error rate, base-calling time and quality score

Ashley Cacho is a PhD Candidate in Applied Statistics at the Universityof California Riverside. She is interested in statistics and bioinformatics.
Ekaterina Smirnova is a post-doc at the University of Wyoming, Department of Statistics. She is interested in reflecting the effect of bias introduced at each step of next generation sequencing pipeline on interpretation of statistical analysis results and developing error correction methods.
Snehalata Huzurbazar is Associate Professor of Statistics at the University of Wyoming. Previously she spent 2 years as the Deputy Director of the Statistical and Applied Mathematical Sciences Institute in North Carolina where she developed the 2014-15 SAMSI research program on Bioinformatics.
Xinping Cui is a Professor in the Department of Statistics, University of California, Riverside. Her research focuses on statistical methods for NGS data analysis including base calling, variant calling, metagenomic binning; multiple testing; high-dimensional clustering and classification and system biology.

Submitted: 28 May 2015; **Received (in revised form):** 5 August 2015

Table 1. The summary of the base-calling procedures available for the Illumina platform

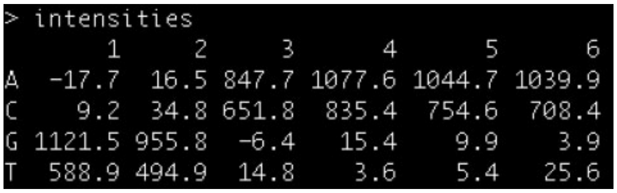
Name	Year	Input	Quality measure	Model-type	Practical notes
BlindCall [3]	2014	CIF	None	Blind deconvolution	Requires Matlab
freeIbis [4]	2013	INT or CIF with Bustard reads	Phred	SVM	
Softy [5]	2013	INT	Probability	Parametric	
AYB [6]	2012	CIF	Phred	Nonparametric	
OnlineCall [7]	2012	INT	Probability	Parametric	
BM-BC [8]	2012	INT	Unknown	Parametric	Unsuccessful installation[a]
ParticleCall [9]	2012	INT	Probability	Parametric	Unsuccessful installation[a]
TotalReCaller [10]	2011	INT or CIF	None	Parametric w/ alignment	
naiveBayesCall [11]	2010	INT	Probability	Parametric	
Srfim [12]	2009	INT or CIF	Phred	Parametric	
BayesCall [13]	2009	CIF	Probability	Parametric	
Ibis [14]	2009	INT or CIF with Bustard reads	Phred	SVM	Not open-source
Rolexa [15]	2008	INT or CIF	Phred	Parametric	
Alta-Cyclic [16]	2008	INT with Bustard reads	Unknown	SVM	Requires Sun-grid engine
Bustard	N/A	N/A	Phred	Nonparametric	Not open-source

The older Illumina Genome Analyzer outputs intensities in the INT text format while the newer platforms output the intensities as binary cluster intensity files (CIF). For every base-call, there is a quality measure attached that measures the probability of a correct base-call or logarithmically related to the base-call error probability.
[a]Considerable efforts were made without success in installing the software, with several email exchanges with the authors.

accuracy for the Illumina platform using a data set of 286 847 reads of length 51 from φX174. However, they were not able to successfully install some software programs, owing to the fact that some base-callers are no longer maintained while others require specific computing systems. The paper also discusses base-calling on the Roche 454 platform but provide no performance comparisons. In addition to what was previously reviewed, we also included Srfim, OnlineCall, All Your Base (AYB), Softy, freeIbis and BlindCall. We also faced problems with installation and implementation of some base-callers. In particular, ParticleCall, BM-BC and the first version of TotalReCaller that was obtained while it was still open-source for academic purposes could not be implemented successfully, despite considerable efforts. The current version of TotalReCaller could not be obtained owing to proprietary issues. The Alta-Cyclic base-caller requires a Sun-grid engine, which we do not have access to, and so it was excluded from the analysis. Since the survey in [17], several new base-calling algorithms have emerged, with many focusing on speed while maintaining accuracy.

Base-calling

Among several competing commercial platforms, a widely used sequencer is the Illumina platform. Sequencing on this platform begins with the library preparation step, followed by bridge amplification and, finally, sequencing-by-synthesis. DNA samples that will contain several copies of a genome are acquired from an organism and randomly fragmented into shorter pieces through the commonly used sonification method. Adaptors are then ligated to each end of the double-stranded DNA fragments, and to generate enough amounts of DNA material to sequence, the fragments go through several polymerase chain reaction amplification cycles. The double-stranded DNA fragments are denatured, and the resulting single strands are covalently bound to the dense lawn of oligo primers found on the eight-lane glass flow cell. Each of these single-stranded DNA fragments serve as a template strand. To generate enough signal for sequencing, the templates are copied several times through a bridge amplification process where ~1000 copies are generated to form a tightly knit cluster of identical template strands. The sequencing-by-synthesis process is conducted in parallel for each cluster, thus sequentially building the complementary strands. This happens in cycles where for each cycle, there is an addition of DNA polymerase and molecules consisting of fluorescently labeled terminating bases with an attached reversible terminator. These terminators allow the process to be carried out sequentially by preventing several fluorescently labeled bases to incorporate all at once. After the incorporation of a single base, a laser excites each of the clusters to allow for the emission of fluorescence, which will be captured by charge-coupled device imaging. The reversible terminators are removed to allow for the incorporation of the next base in the next sequencing cycle [18].



```
> intensities
       1      2      3      4      5      6
A  -17.7   16.5  847.7 1077.6 1044.7 1039.9
C    9.2   34.8  651.8  835.4  754.6  708.4
G 1121.5  955.8   -6.4   15.4    9.9    3.9
T  588.9  494.9   14.8    3.6    5.4   25.6
```

Figure 1. Intensity matrix for the first six cycles of a single read.

At the end of each sequencing cycle, a set of four images in the optimal wavelengths is taken for each of the fluorophores to capture the emitted fluorescence. The image processing produces an intensity quadruple where each value represents the intensity for the corresponding nucleotide channels A, C, G and T. Figure 1 shows an example of intensity output for the first six cycles of a single cluster. For more information on this sequence-by-synthesis process, helpful videos on the Illumina Web site are available (http://www.illumina.com/technology/next-generation-sequencing/sequencing-technology.html). The output for each of the Illumina sequencing machines includes intensities over all reads and cycles, cluster positions as (x, y) coordinates and the base-calls by the built-in base-caller, Bustard. The process of inferring the base from the intensity quadruple is what is known as 'base-calling'. Ideally, the channel in which the maximum intensity occurs would be the base that is 'called'. However, the chemical processes involved in sequencing are imperfect, leading to errors in base-calling.

Owing to the technology, the Illumina platform suffers from several major biases including phasing (lagging) and

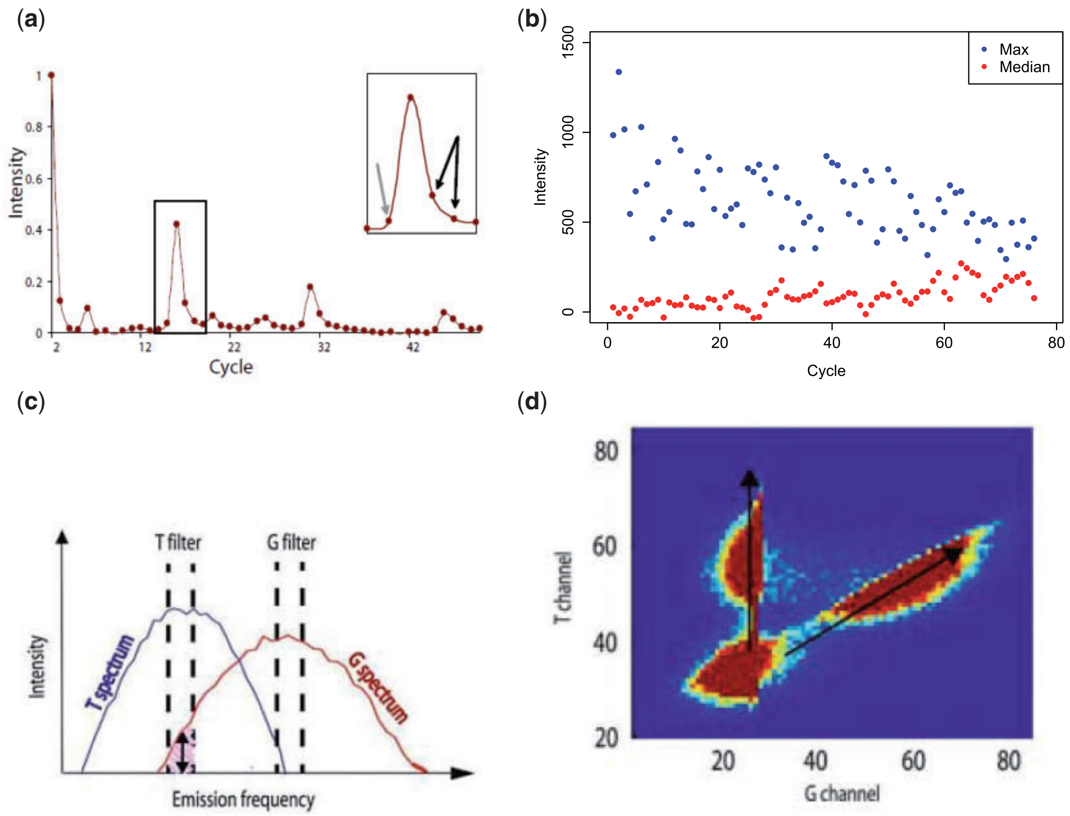


Figure 2. Commonly modeled base-calling errors for the Illumina platform. (A) Scaled C intensity channels versus cycle of a single read. A spike indicates a potential C nucleotide occurs at that position. Phasing can be seen as an anticipation signal in the cycle before a C (left arrow) and subsequent cycles after (right arrows) [16]. (B) Maximum intensity (signal) and median intensity (noise) plotted against cycle. (C) Intensity versus fluorophore emission spectrum. The spectrum of the G fluorophore bleeds (pink shading) into the optimal spectrum of the T filter. Thus, when a G fluorophore is excited, a T signal will also be detected [19]. (D) Two-dimensional histogram of intensity data of the T channel versus G channel. The G fluorophores (right arrow) transmit to the to T channel, hence the positive linearity. However, the T fluorophores do not transmit to the G channel [19].

prephasing (leading), signal decay and cross-talk. These biases can be visualized through Figure 2. Phasing is the phenomenon that occurs during the sequencing process when a strand in a given cluster of amplified DNA templates fails to incorporate or synthesize the next base in the read because some of the enzymes fail to work simultaneously. The read starts lagging behind the rest of the reads in the cluster and may continue synthesizing or become completely inactive. Because of this, photographs of the fluorescence emissions are not accurate, leading to errors in the base-calls. Prephasing occurs when two bases are synthesized in a single sequencing cycle and prove to have similar consequences as of phasing. Figure 2A shows the scaled intensities over the C channel of a single read. Where there is a spike in the scaled C channel intensity, a C is expected. If the chemistry in sequencing was perfect, all other bases would produce 0 intensity values. Thus, phasing can be seen as an anticipation signal in the cycle before a C (left arrow) and subsequent cycles after (right arrows). As the complementary strand continues to be synthesized, some sequencing material may be lost, which causes a decrease in overall signal and increase in noise. This phenomenon is known as signal decay; see Figure 2B where the maximum of the intensity quadruple (signal) and the median of the remaining three intensities (noise) are plotted for each cycle. Cross-talk is the phenomenon that occurs when fluorophore emission spectra overlap, which causes a positive correlation between the intensities in those channels. In Figure 2C, the emission spectrum of G bleeds into the T filter (pink shading). Thus, when the G fluorophore is excited, signal from T is also captured, see Figure 2D for a two-dimensional histogram of intensity data of the T channel versus G channel. When this occurs, signal from G (right arrow) transmits to the T channel, hence the positive linearity. However, the T fluorophores do not transmit to the G channel. A similar phenomenon occurs with both fluorophores of the bases A and C, transmitting to the C and A channels, respectively. Several algorithms seek to explicitly model these biases to generate more accurate base-calls.

A unified base-calling model

Several statistical approaches have emerged within the past few years aimed at generating more accurate base-calls. The techniques used in the algorithms range from parametric to nonparametric, and (statistical) model-based to completely empirical machine learning methods. Figure 3 provides an overview of different modeling techniques currently being used.

To provide better insight into the different modeling techniques, we first define the following unified model for base-calling as

$$\begin{aligned} Z_i - B &= Y_i \\ &= MX_iPD + E_i \end{aligned} \qquad (1)$$

Refer to the top of Table 2 for the definitions of the matrices and dimensions. Most parametric and nonparametric

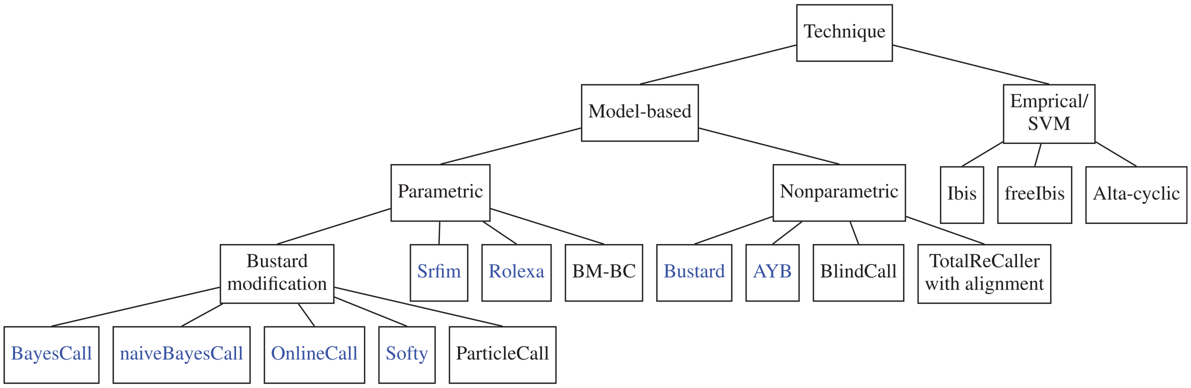


Figure 3. The different modeling techniques currently being used. The base-callers highlighted in blue will be shown to be special cases of the general statistical model and we refer the reader to the Supplementary Material for the specification of the remaining methods.

Table 2. Top: notation that will be used to describe the general statistical model. Bottom: real intensity data in Figure 1 along with notation

Notation	Dimension	Definition
$\mathbf{Z}_i$	4xJ	Intensities before Illumina correction
$\mathbf{B}$	4xJ	Background correction
$\mathbf{Y}_i$	4xJ	Observed intensities
$\mathbf{M}$	4x4	Cross-talk matrix
$\mathbf{X}_i$	4xJ	True (latent) intensities or nucleotide indicators
$\mathbf{P}$	JxJ	Phasing/prephasing matrix
$\mathbf{D}$	JxJ	Signal decay matrix
$\mathbf{E}_i$	4xJ	Error term matrix

Observed intensities Y_i

	1	2	3	4	5	6
A	Y_{i1A}	Y_{i2A}	Y_{i3A}	Y_{i4A}	Y_{i5A}	Y_{i6A}
C	Y_{i1C}	Y_{i2C}	Y_{i3C}	Y_{i4C}	Y_{i5C}	Y_{i6C}
G	Y_{i1G}	Y_{i2G}	Y_{i3G}	Y_{i4G}	Y_{i5G}	Y_{i6G}
T	Y_{i1T}	Y_{i2T}	Y_{i3T}	Y_{i4T}	Y_{i5T}	Y_{i6T}
	↓					
	1	2	3	4	5	6
A	−17.70	16.50	847.70	1077.60	1044.70	1039.90
C	9.20	34.80	651.80	835.40	754.60	708.40
G	1121.50	955.80	−6.40	15.40	9.90	3.90
T	588.90	494.90	14.80	3.60	5.40	25.60

model-based methods are special cases of the general model, while the machine learning-based methods are not. The following indices will be used; read/cluster $i = 1, 2, \ldots, N$, cycle $j = 1, 2, \ldots, J$, and channel $k = A, C, G, T$ or 1, 2, 3, 4, respectively. The $\mathbf{Z}_i$'s are the intensities before Illumina's background correction $\mathbf{B}$ is performed. Unfortunately, these raw intensities, namely $\mathbf{Z}_i$'s, are not retrievable. However, the $\mathbf{Y}_i$'s are the observed intensities and interest is in obtaining the $\mathbf{X}_i$'s, the true (latent) values. The rows and columns of both $\mathbf{Y}_i$ and $\mathbf{X}_i$ represent the channel and cycle, respectively. Depending on the model, the columns of $\mathbf{X}_i$ either represent the true intensities or serve as nucleotide indicators where, for example, a nucleotide C would be represented as $(0, 1, 0, 0)^T$. The $\mathbf{M}$, $\mathbf{P}$ and $\mathbf{D}$ matrices model the cross-talk, phasing/prephasing, and signal decay, respectively, and the error term matrix is represented by $\mathbf{E}_i$. Observed intensity matrix $\mathbf{Y}_i$ for Figure 1 is shown at the bottom of Table 2.

Several existing base-calling algorithms can be written as special cases of this unified model. Here, we start with Bustard, which is the most frequently used base-caller, as it is readily available with the Illumina platform. We then describe in detail the Srfim model as an example of the parametric model-based methods, AYB model as a representation for the nonparametric model-based techniques and Ibis/freeIbis for the machine learning-based methods. We refer the reader to the Supplementary Material for the specification of the other base-callers.

Bustard

The Illumina sequencing platform has a built-in base-caller called Bustard. It is important to understand the underlying model because it is the most widely used base-caller, and several base-calling algorithms were built using Bustard as their base. For the model underlying Bustard, consider Equation (1) with

$$\mathbf{M} = \begin{pmatrix} 1 & m_{12} & m_{13} & m_{14} \\ m_{21} & 1 & m_{23} & m_{24} \\ m_{31} & m_{32} & 1 & m_{34} \\ m_{41} & m_{42} & m_{43} & 1 \end{pmatrix}$$

where each component of the cross-talk matrix m_{rs} indicates the amount of observed intensity in channel s generated by the signal from nucleotide r; for each r, s = A, C, G, T (or r, s = 1, 2, 3, 4). Li and Speed [20] suggest an iterative method to estimate the elements of the **M** matrix that is based on considering the intensities as linear combinations of the cross-talk bias and the fluorophore dye concentrations. The approach is to estimate each element m_{rs} and m_{sr} by considering only those channels, r and s. The intensities of the first component of the pair are binned between two chosen quantiles. For those values whose first component falls into a given bin, take the pair having a minimum value in the second component. Of these points that were taken, fit an L-1 regression of the second component against the first component to obtain the estimate of the slope. Treat the second component as the first and repeat the previous step to obtain the estimate of the slope. Under certain criteria, these slope estimates will estimate m_{rs} and $m_{sr,}$ respectively. For further details, we refer the reader to [20].

To model phasing and prephasing, the matrix **P** in Equation (1) is obtained from the transition probability matrix **Q** ($J \times J$), which models the position of the terminator at position u to v with elements

$$Q_{uv} = \begin{cases} p & \textit{if } v = u \\ 1 - p - q & \textit{if } v = u + 1 \\ q & \textit{if } v = u + 2 \\ 0 & \textit{otherwise} \end{cases}$$

where p is the probability of phasing, q is the probability of prephasing and 1-p-q is the probability of normal incorporation. The (u,v)th element of the t-step transition probability matrix $\mathbf{Q}^t$ denotes the probability that a template strand at position u moves to position v after t cycles. Thus, the (v,t)th element of the phasing/prephasing matrix **P** is given by

$$P_{vt} = \left[\mathbf{Q}^t\right]_{0,v} = p * P_{v-1,t} + (1 - p - q) * P_{v-1,t-1} + q * P_{v-1,t-2}$$

for v = 2, ... , J and t = 1, ... , J, where the P_{11} element is 1-p-q, P_{12} element is q and the remaining columns of the first row are 0. The phasing p and prephasing q are estimated as two channel-independent parameters from the increasing correlation of intensities in the first few cycles of the sequencing run [13]. The loss in signal, modeled by $\mathbf{D} = \left[diag\left(\frac{\overline{W}_1}{\overline{W}_1}, \frac{\overline{W}_1}{\overline{W}_2}, \ \ldots, \ \frac{\overline{W}_1}{\overline{W}_J}\right)\right]^{-1}$, is addressed with renormalizing the concentrations by taking the average of cross-talk corrected intensities, and using it as the normalizing factor where $\overline{W}_i = \sum_{i=1}^{N} Y'_{ijA} + Y'_{ijC} + Y'_{ijG} + Y'_{ijT}$ and $\mathbf{Y}'_i = \mathbf{M}^{-1}\mathbf{Y}_i$ [13]. In performing the base-calls, the Bustard model does not consider the error term $\mathbf{E}_{i.}$ Solving for $\mathbf{X}_i$ in Equation (1) yields $\mathbf{X}_i = \mathbf{M}^{-1}\mathbf{Y}_i\mathbf{D}^{-1}\mathbf{P}^{-1}$. Finally, the row index of the largest entry for each of the columns of $\mathbf{X}_i$ is the base that is called.

Srfim

A fully parametric base-calling model written by Bravo and Irizarry in 2009 is implemented in a software program called Short-Read Filtering and Intensity Modeling (Srfim) [12]. The approach here is entirely different from the Bustard modeling techniques. Consider Equation (1) as

$$\mathbf{Y}_i^* = \mathbf{M}\mathbf{X}_i\mathbf{P}\mathbf{D} + \mathbf{E}_i$$

where, $\mathbf{Y}_i^* = \log(\mathbf{Y}_i)$, with log(.) the component-wise function. The **M** matrix is the same as in Bustard. The Srfim model does not model phasing and prephasing so the **P** matrix is the identity matrix. Hence, Equation (1) can be considered as

$$\mathbf{Y}'_i = \mathbf{X}_i\mathbf{D} + \mathbf{E}'_i$$

The underlying mechanism for generating $\mathbf{X}_i\mathbf{D}$ is based on a fixed-effects mixture model. Conditional on a latent nucleotide indicator π_{ijk}, which is defined as 1 if the nucleotide in the ith read, at the jth cycle is k and 0 otherwise, the jth column of $\mathbf{X}_i\mathbf{D}$ is of the form $\mathbf{U}_j\boldsymbol{\beta}_{ij}{}^k$ where the design matrix $\mathbf{U}_j$ is

$$\mathbf{U}_j = \begin{pmatrix} 1 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & j-1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & j-1 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & j-1 & 0 \\ 0 & 0 & 0 & 1 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & j-1 \end{pmatrix}$$

and the $\boldsymbol{\beta}_{ij}{}^k$'s are 12 × 1 vectors of fixed effects. Thus, conditional on $\pi_{ijk} = 1$, the jth column of $\mathbf{Y}'_i$ is modeled as

$$\mathbf{Y}'_{ij} = \mathbf{U}_j\boldsymbol{\beta}_{ij}^k + \mathbf{E}'^k_{ij}$$

where the jth column of $\mathbf{E}'_i$ is modeled as $\mathbf{E}'^k_{ij} \sim N_4(0, \Sigma_i^k)$, and thus, $\mathbf{Y}'_{ij} \sim N_4(\mathbf{U}_j\boldsymbol{\beta}_{ij}^k, \Sigma_i^k)$. For example, suppose $\pi_{ijA} = 1$, then $\boldsymbol{\beta}_{ij}^A = (\mu_{Aj\alpha}, \mu_{Cj\delta}, \mu_{Gj\delta}, \mu_{Tj\delta}, \ \alpha_{0i}, \beta_{0i}, \beta_{0i}, \beta_{0i}, \alpha_{1i}, \beta_{1i}, \ \beta_{1i}, \beta_{1i})^T$ and $\Sigma_i^A = diag(\sigma_{\alpha i}^2, \sigma_{\delta i}^2, \sigma_{\delta i}^2, \sigma_{\delta i}^2)$. The α's and δ's in both $\boldsymbol{\beta}_{ij}{}^k$ and Σ_i^k represent the signal and noise models, respectively. The μ's represent the cycle-channel effect and, coupled with the fixed effects with 0 subscripts, serve as the intercepts of the linear model. The subscripts with 1 represent the slopes of the linear model. A more common way to denote a mixture distribution is

$$\mathbf{Y}'_{ij} \sim \sum_{k \in \{A,C,G,T\}} \pi_{ijk} N_4(\mathbf{U}_j\boldsymbol{\beta}_{ij}^k, \Sigma_i^k)$$

The parameters are estimated using the Expectation-Maximization (EM) algorithm, and the estimates of the mixture probabilities π_{ijk} are used in base-calling.

AYB

The AYB [6] base-caller is nonparametric in that no distributional assumptions are made on the observed intensities. The AYB model incorporates variation in cross-talk that occurs on a cycle-wise basis for the whole sequence, with phasing rates that allow for flexibility in accounting for sequence-specific errors, so that Equation (1) becomes

$$\mathbf{Y}_i^* = \lambda_i\mathbf{M}\mathbf{X}_i\mathbf{P}\mathbf{D} + \mathbf{E}_i$$

where $\mathbf{Y}_i^* = \mathbf{Y}_i - \mathbf{N}$ and **N** is another background correction that is considered as the systematic noise, while $\mathbf{E}_i$ is the random noise. The cross-talk matrix **M** is of the same form as the one used in the Bustard, BayesCall, naiveBayesCall, OnlineCall, Softy and Srfim models (see Supplementary Material). The $\mathbf{X}_i$ matrix is defined such that the jth column can be considered a nucleotide indicator, a unit vector with a 1 in one of the entries and the rest 0s. A parameter λ_i is introduced to describe the amount of light emitted by a cluster i, which is proportional to the number of molecules in the given cluster. The (u,v)th element of the phasing matrix **P** describes the relative proportion of fluorophore-labeled nucleotides bound to position v of the template sequence during cycle u. The signal decay is incorporated into the phasing matrix by scaling the columns such that each sums to the proportion of molecules in the cluster expected to still continue synthesizing bases. Thus, the signal decay matrix **D** is a scaling matrix. The error term $\mathbf{E}_i$ has expectation zero and an unassumed covariance structure. The estimation procedure for the parameters is carried out through an Iteratively reWeighted Least Squares algorithm where the weights are chosen from the Cauchy function. The posterior probabilities obtained in the estimation procedure are used to call bases.

Ibis and freeIbis

Rather than trying to build a model that explains all of the biases found in the Illumina sequencing technology, of which our understanding may still be incomplete, and having to adjust the model for different sequencing platforms, the Improved Base Identification System (Ibis) [14] method uses a statistical learning scheme. Ibis uses multi-class support vector machines (SVMs) with polynomial kernels corresponding to each cycle. The input of each SVM is a 12-feature vector consisting of the raw intensities at the current, preceding and succeeding cycles. During this training stage, the optimal hyperplane for the separable patterns between the bases is found. To train the cycle-dependent SVMs, a set of training data created from alignment of Bustard reads to a known reference genome is required. During sequencing, it is recommended to include a control sample DNA from the species with the known reference genome. Although it is possible to run both Ibis and freeIbis without a control, the results are better with a spiked-in control such as a sample from φX174. It is assumed that the sequence of the known reference genome is the correct one. An open-source version of Ibis called freeIbis [4] was created, which uses a different multi-class SVM software that is open source. Another improvement is on the produced sequence quality scores, which are calibrated using empirically observed scores.

Materials and methods

For our comparisons across base-callers, we used a fairly standard data set; specifically, sample DNA from bacteriophage φX174 that was sequenced on a full sequencing lane composed of 100 tiles on the Illumina Genome Analyzer II, producing reads of length 76. For each tile, there are between 70 000 and 80 000 reads, and so a full lane comprises approximately 8 million reads. A second data set consisting of one tile containing ~30 000 reads of length 101 from a human sample with φX174 spiked-in for control and sequenced using a HiSeq platform was also used in the analysis. This data set is a subset of the data used in the AYB paper. The size of the text intensity file for each tile in the φX174 data set is ~131M and for human, 56M, uncompressed. We ran all algorithms with default options. The SVMs for Ibis and freeIbis were trained using the first tile for φX174. Through helpful correspondence with Kircher, there should be at least 10k control sequences found within the sample to obtain meaningful results. The HiSeq sample was found to have only 1.5% of the needed control sequences. Thus, Ibis and freeIbis have been excluded from the comparison of the HiSeq data because it would not be a fair comparison, as there is a lack of control sequences in the sample. After performing the different base-calling algorithms, alignment to the known reference genome of φX174, which comprises 5386 nucleotides or Human Genome build 19, restricted to only chromosomes 1–22, X, Y and mitochondrial was run using BWA [19] with default options allowing up to four mismatches. AYB, Rolexa, Srfim, Ibis and freeIbis present output as FastQ files, and so these can be directly used in further data analyses. For BayesCall, naïve BayesCall, OnlineCall and both Softy algorithms (Forward-Backward (FB) and Soft output Viterbi algorithm (SOVA)), FastQ-like files are output where the qualities are reported as probabilities. For use in further data analyses, these probabilities must be converted to quality scores.

To evaluate the performance of these base-callers, we used three metrics; alignment rate, error rate and discrimination ability. The alignment rate is defined to be the ratio of reads that align the reference to the total number of reads that were base-called. A high alignment rate is important for downstream analysis so that more information is available. The error rate is defined to be the ratio of the number of mismatches on those reads that aligned to the number of bases in the aligned reads. To distinguish the bad and good base-calls, a quality score is provided with each base-call. It is common to use the Phred quality score defined to be $Q = -10\log_{10}P$, where P is the probability of an incorrect base-call, and in this comparison, we converted all quality scores to Phred. To define discrimination ability, we sort the bases aligned to the reference genome by their quality scores in increasing order, and compute the number of correct matches up to each Phred score weighted by the alignment rate for that Phred score.

Results

Table 3 compares the running time per tile in minutes. There seems to be three groups in running times: those that run under a minute, approximately 5 min and much longer. BlindCall, Ibis, freeIbis and OnlineCall all run under a minute. Rolexa and both

Table 3. Average run times per tile for φX174 and total run time for a single tile of human. We used two of Softy's implementations, SOVA and FB

Base-caller	φX174 Time (min) per tile	HiSeq Time
BlindCall	0.21	0.17
Ibis	0.27	N/A
freeIbis	0.39	N/A
OnlineCall	0.51	0.21
Srfim	3.91	2.03
AYB	4.73	5.54
SoftySOVA	5.02	2.88
SoftyFB	5.71[a]	3.80
Rolexa	104.49	22.29
naiveBayesCall	708.84	74.86
BayesCall	898.71	99.91

[a] The per-tile time was reported because the Softy FB algorithm was not able to base-call tile 3.

BayesCall algorithms are computationally expensive owing to their respective implementations of the EM algorithm used for estimation of parameters. The BayesCall models also have a large number of parameters to estimate because they incorporate a cycle-dependent parameter into their model (see Supplementary Material). Table 4 compares the alignment rates, which is defined to be the proportion of base-called reads that align back to the reference genome. The alignment rates all seem comparable, with BayesCall having the highest alignment rate at 91.38% in φX174 and 84.45% in human. Nearly half of the base-callers improve on Bustard's alignment rate in both sets. Table 4 also shows the overall error rate, which is defined to be the ratio of the number of mismatches on those reads that were aligned to the reference, to the total number of bases of the aligned reads. The error rates are also comparable in both sets, with AYB having the lowest error rate at 0.41% in φX174 and 0.40% in human. In both data sets, OnlineCall and Rolexa have considerably lower alignment and higher error rates.

As mentioned above, Ibis and freeIbis were excluded from the human data set. Although these have been excluded, we report results of Ibis base-calls from AYB's analysis to get a sense of the performance; see Table 5. Ibis was trained on the first lane out of the total of eight. As you can see, Ibis performs well with enough control φX174 reads found in the sample.

Figures 4 and 5 compare error rates over cycle and tile, respectively. In Figure 4, no base-caller clearly performs the best across all cycles. For the φX174 data (Figure 4A and B), in the first ∼40 cycles, freeIbis and Ibis have the lowest error rates and BlindCall, BayesCall, naiveBayesCall, Softy FB and Softy SOVA have similar performance to Bustard. After about cycle 45, AYB has the lowest error rate, and BayesCall, freeIbis, Ibis, naiveBayesCall, Softy FB and Softy SOVA have similar performance, but still have slightly lower error rates than Bustard. In the first ∼5 cycles of the human data set, all base-callers except for BayesCall show some inconsistent error rates. After ∼65 cycles, both Softy implementations have lower error rates than Bustard, while AYB has the lowest. In Figure 5, AYB clearly has the lowest error rate across all tiles, and BayesCall, freeIbis, Ibis, naiveBayesCall, Softy FB and Softy SOVA all have similar performance but still maintain an advantage over Bustard. BlindCall has similar error rates across tile as Bustard. Srfim does not show consistent error rates across tiles. Rolexa has high error rates because the algorithm identifies ambiguous bases and codes them with IUPAC symbols instead of calling one of the four nucleotide bases.

Figure 6 assesses discrimination ability of the quality scores produced by each base-caller. The distributed BlindCall software does not provide quality scores with their base-calls, and so it was excluded from Figure 6. Considerable efforts were made to contact the authors, and it was discovered that BlindCall software requires an external aligner not included in their software to obtain the quality scores. The Rolexa R package does not provide a way to obtain quality scores, and so it was also excluded from Figure 6. A good quality assignment method would maintain high discrimination ability as the probability of making a correct base-call approaches one. Both implementations of BayesCall and naiveBayesCall maintain high discrimination ability in φX174 and human because exact matches to the reference occur at high-quality scores. The parametric models Srfim, Softy (FB and SOVA), BayesCall, naiveBayesCall and OnlineCall maintain higher discrimination ability over the empirical models Ibis, freeIbis and Bustard in both data sets because meaningful posterior probabilities are obtained from the estimation procedures. AYB maintains higher discrimination ability over Bustard in the φX174 data set but not in the human. The nonparametric Illumina built-in base-caller Bustard tends to assign smaller quality scores to the correctly called bases in the φX174 data set. While its alignment rate is not much higher than the other base-calling methods considered in this study (see Table 1), the number of mismatches in aligned reads is considerably higher. The phred quality scores above 35 produced by the SVM-based methods freeIbis and Ibis are not reliable, as most of the reads with these quality scores did not align back to the reference genome.

Table 4. The alignment rates are reported along with some notes

Base-caller	φX174		HiSeq	
	Alignment rate	Error rate	Alignment rate	Error rate
BayesCall[a]	0.9138	0.0045	0.8445	0.0049
AYB	0.9033	0.0041	0.8276	0.0040
freeIbis	0.8982	0.0044	N/A	N/A
Ibis	0.8970	0.0045	N/A	N/A
naiveBayesCall[b]	0.8965	0.0044	0.8204	0.0046
SoftyFB[b]	0.8964[c]	0.0044[c]	0.8098	0.0045
SoftySOVA[b]	0.8963	0.0044	0.8147	0.0047
BlindCall	0.8864	0.0049	0.8140	0.0048
Bustard	0.8826	0.0050	0.8162	0.0047
Srfim	0.8733	0.0060	0.8003	0.0048
OnlineCall[d]	0.8303	0.0113	0.7050	0.0137
Rolexa	0.6922	0.0202	0.4833	0.0290

[a] Some reads were not included in the output fastq file, for an unknown reason.

[b] Some probabilities were reported as -nan so these reads were removed.

[c] The rate is out of 99 tiles because it was not able to base-call tile 3.

[d] Base-called reads that were less than 76 or 101 cycles were removed. Reads with probabilities greater than 1 were removed. The error rate is calculated over all tiles.

Table 5. Alignment and error rates for AYB's reported results of Ibis base-calls on the human HiSeq data

	Alignment rate	Error rate
Ibis	0.8222	0.0042

Ibis was trained with the first lane of the eight lanes, and here we report results of the first tile.

Discussion

As high-throughput sequencing technologies advance, the need for statistical methodologies to handle such immense amounts of data is imperative. We have reviewed several base-calling algorithms for the Illumina sequencing platform, as accurate base-calling is key to further data analysis.

With the introduction of the general base-calling framework, each method becomes easier to assess and understand. Each modeling technique has its advantages and disadvantages, namely for model-based methods, clear interpretations for the parameters exist, and for machine learning methods, adaptation to other sequencing platforms is much easier. For the model-based methods, the best way to model the biases remains to be determined. Most of the methods use the **M** matrix proposed by Li and Speed to deal with cross-talk, but there does not seem to be a consensus with modeling phasing, prephasing

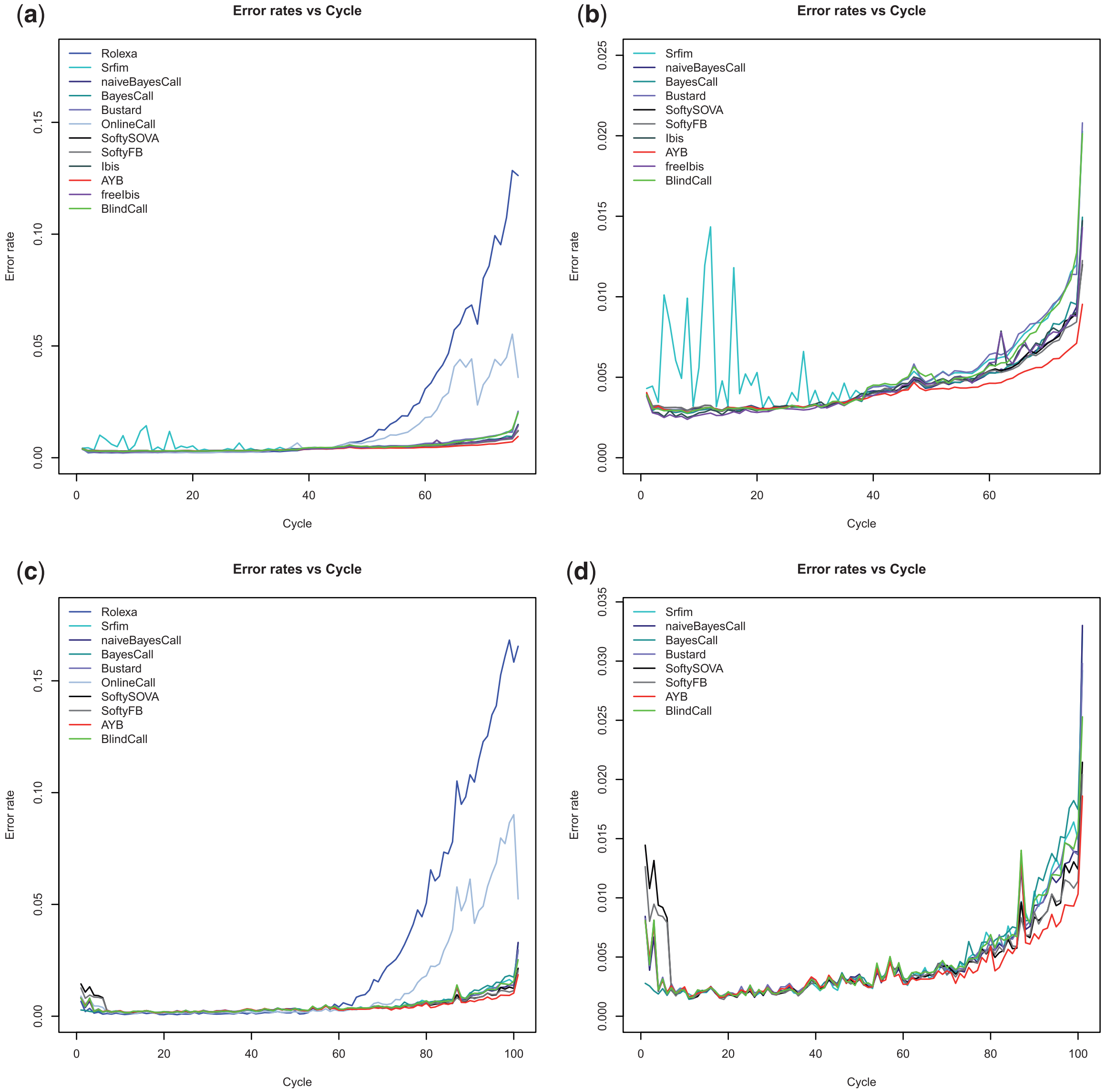


Figure 4. Error rate across cycle. (**A**) error rate across cycle for base-callers for φX174, (**B**) error rate across cycle for base-callers excluding Rolexa, OnlineCall and Srfim for φX174, (**C**) error rate across cycle for base-callers for human, (**D**) error rate across cycle for base-callers excluding Rolexa and OnlineCall.

and signal decay. The SVM-based models, Ibis and freeIbis, do not apply any bias corrections but are able to maintain low error rates. One small disadvantage to Ibis and freeIbis is that they require a control sequence. Because Srfim has inconsistent error rates at the beginning cycles and both Rolexa and OnlineCall have higher error rates at the end, these base-callers are not recommended. The nonparametric model-based method of AYB has the lowest error rate across tile and for the latter parts of the sequencing cycles.

As with any method in high-throughput sequencing, computationally efficient algorithms must be considered. There is a trade-off between having low error rates and fast base-calling algorithms. There are three groups of running times per tile: those that run under a minute, ~5 min and anything considerably more than 5 min. The more recent base-calling techniques focus on maximizing computational efficiency while maintaining low error rates. In terms of speed, BayesCall, naiveBayesCall and Rolexa would not be recommended. Both machine learning methods, Ibis and freeIbis, as well as BlindCall are the fastest algorithms.

In general, alignment rate is important to ensure that as much of the available data can be used. The alignment rates are all comparable apart from OnlineCall and Rolexa. Approximately half of the base-callers considered in both data sets improve on Bustard, with BayesCall having the highest alignment rate. Although BayesCall has the highest alignment rates, it is not recommended because of the total time taken for base-calling.

Sequencing is usually not the end goal of any high-throughput sequencing project. The next logical step after sequencing is base-calling. Thus, results of base-calling will affect any downstream analysis. In particular, the quality of the bases affects the analysis through the direct use such as in variant calling or through quality filtering such as in BS-Seq, which requires high-quality bases. Quality filtering of bases may lead to the removal of a large amount of bases that do not meet the

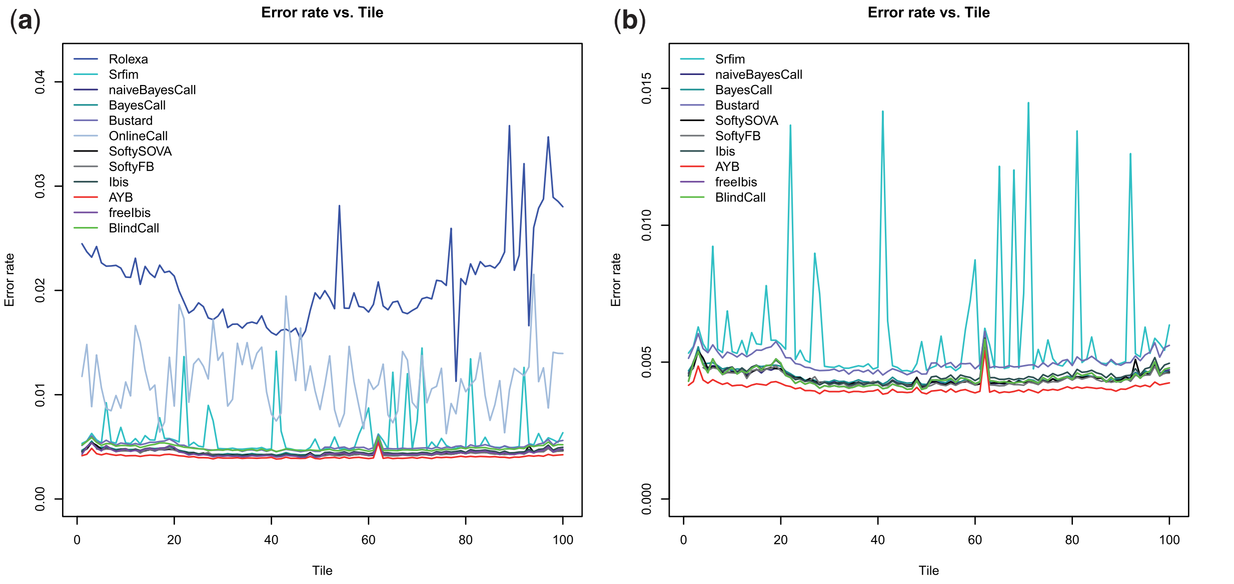


Figure 5. Error rate across tile. (**A**) Error rate across tile for all base-callers on φX174. (**B**) Error rate across tile for base-callers excluding Rolexa and OnlineCall on φX174

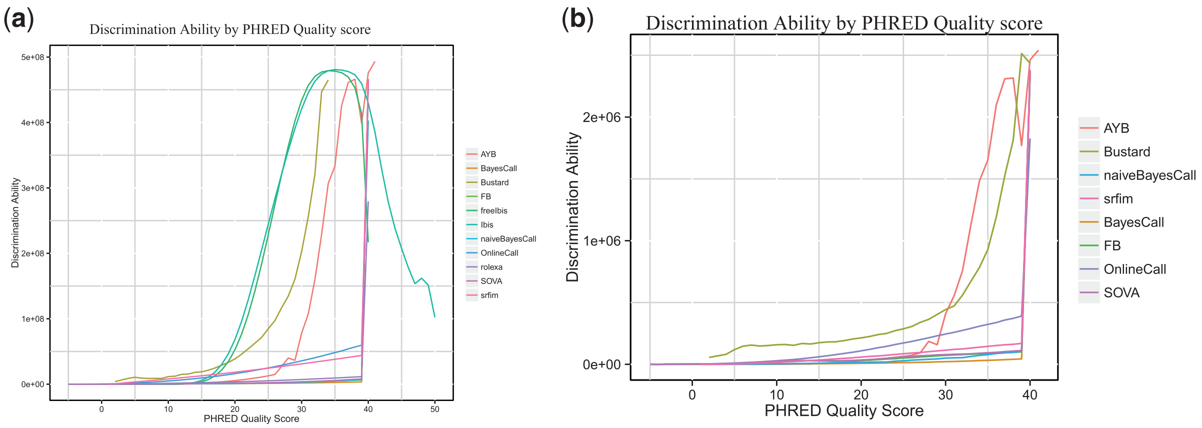


Figure 6. The discrimination ability is the number of exact matches to the reference as a function of Phred quality. (**A**) φX174 and (**B**) human.

quality standards. The parametric models Srfim, Softy (FB and SOVA), BayesCall, naiveBayesCall and OnlineCall have better quality score assignments because of the clear interpretations of posterior probabilities that result from estimation not found in machine learning methods. AYB maintains a high discrimination ability over the other nonparametric models and machine learning methods because the sequencing events are explicitly modeled.

There is no clear best base-caller in terms of all the performance metrics considered in this analysis. However, Ibis, freeIbis and AYB seem to outperform the rest of the base-callers and are recommended for base-calling. They are comparable in alignment and error rates but differ in base-calling time and quality score assignments. Ibis runs faster than AYB but AYB has higher discrimination ability. If a control sequence is not available or if the downstream analysis requires high-quality bases, then we would recommend AYB, but if the speed of base-calling is important to the researcher, then we would recommend Ibis or freeIbis. Base-calling is an important step after sequencing that must be carefully considered, as it can influence the analysis of sequencing through increased accuracy.

Key Points

- Base-calling is the process of inferring the nucleotide base from the intensity signals.
- The general statistical model that unifies model-based methods helps to understand the different techniques to handle the biases found in sequencing.
- Several alternative base-calling methods for the Illumina platform have been shown to outperform the built-in base-callers.
- AYB, freeIbis and Ibis are all recommended base-callers.
- For runs without a control or if the downstream analyses requires high-quality bases, AYB is recommended.
- For computational time, we recommend either freeIbis or Ibis.

Supplementary Data

Supplementary data are available online at http://bib.oxfordjournals.org/.

Acknowledgments

The authors thank the Institute for Integrative Genome Biology Bioinformatics Facility at University of California, Riverside for providing the bioinformatics cluster.

This material was based on work partially supported by the National Science Foundation under Grant DMS-1127914 to the Statistical and Applied Mathematical Sciences Institute. Any opinions, findings, and conclusions or recommendations expressed in this material are those of the author(s) and do not necessarily reflect the views of the National Science Foundation.

Funding

National Science Foundation ATD-1222718 for X.C. and A.C., and DMS-1100615 for S.H. and E.S.

References

1. Sanger F, Nicklen S, Coulson AR. DNA sequencing with chain-terminating inhibitors. *J Mol Biol* 1977;**94**(3):441–8.
2. Chan, EY. Advances in sequencing technology. *Mutat Res* 2005;**573**:13–40.
3. Ye, C, Hsiao, C, Corrada-Bravo, H. BlindCall: ultra-fast base-calling of high-throughput sequencing data by blind deconvolution. *Bioinform* 2014;**30**(9):1214–9.
4. Renaud, G, Kircher, M, Stenzel U, *et al*. FreeIbis: an efficient basecaller with calibrated quality scores for Illumina sequencers. *Bioinformatics* 2013;**29**:1208–9.
5. Das, S, Vikalo, H. Base calling for high-throughput short-read sequencing: dynamic programming solutions. *BMC Bioinformatics* 2013;**14**:129.
6. Massingham, T, Goldman, N. All your base: a fast and accurate probabilistic approach to base calling. *Genome Biol* 2012;**13**:R13.
7. Das, S, Vikalo, H. Onlinecall: fast online parameter estimation and base calling for illumine<5002>s next-generation sequencing. *Bioinformatics* 2012;**28**:1677–83.
8. Ji, Y, Mitra, R, Quintana, F, *et al*. BM-BC: a Bayesian method of base calling for Solexa sequence data. *BMC Bioinformatics* 2012;**13**:S6.
9. Shen, X, Vikalo, H. Particlecall: a particle filter for base calling in next-generation sequencing systems. *BMC Bioinformatics* 2012;**13**:160.
10. Menges, F, Narzisi, G, Mishra, B. Totalrecaller: improved accuracy and performance via integrated alignment and base-calling. *Bioinformatics* 2011;**27**:2330–7.
11. Kao, WC, Song YS. NaiveBayesCall: an efficient model-based base-calling algorithm for high-throughput sequencing. *Lect Notes Comput Sci* 2012;**6044**: 233–47.
12. Corrada-Bravo, H, Irizarry, RA. Model-based quality assessment and base-calling for second-generation sequencing data. *Biometrics* 2009;**3**:665–74.
13. Kao, WC, Stevens, K, Song YS. BayesCall: a model-based base-calling algorithm for high-throughput short-read sequencing. *Genome Res* 2009;**19**:1884–95.
14. Kircher, M, Stenzel, U, Kelso, J. Improved base calling for the Illumina Genome analyzer using machine learning strategies. *Genome Biol* 2009;**10:**R83.1–.9.
15. Rougemont, J, Amzallag, A, Iseli, C. Probabilistic base calling of Solexa sequencing data. *BMC Bioinformatics* 2008;**9:**431.
16. Erlich, Y, Mitra, PP, delaBastide, M, *et al*. Alta-cyclic: a self-optimizing base caller for next-generation sequencing. *Nat Methods* 2008;**5:**679–82.
17. Ledergerber, C, Dessimoz, C. Base-calling for next-generation sequencing platforms. *Brief Bioinform* 2011;**12**:489–97.
18. Illumina, Inc. *Illumina Sequencing Technology: Highest Data Accuracy, Simple Workflow, and A Broad Range of Applications*. 2010. Springer New York Dordrecht, Heidelberg London, http://www.illumina.com/documents/products/techspotlights/techspotlight_sequencing.pdf
19. Sheikh, MA, Erlich, Y. Chapter 5: base-calling for bioinformaticians. In: *Bioinformatics for High Throughput Sequencing*. Springer Link, 2012.
20. Li, L, Speed, T. An estimate of the crosstalk matrix in four-dye fluorescence-based DNA sequencing. *Electrophoresis* 1999;**20**:1433–42.
21. Li, H, Durbin, R. Fast and accurate short read alignment with burrows-wheeler transform. *Bioinformatics* 2009;**25**:1754–60.