BIOINFORMATICS ORIGINAL PAPER

Vol. 29 no. 18 2013, pages 2292–2299
doi:10.1093/bioinformatics/btt381

Gene expression

Advance Access publication July 2, 2013

TIGAR: transcript isoform abundance estimation method with gapped alignment of RNA-Seq data by variational Bayesian inference

Naoki Nariai[1,*], Osamu Hirose[2], Kaname Kojima[1] and Masao Nagasaki[1]

[1]Department of Integrative Genomics, Tohoku Medical Megabank Organization, Tohoku University, Seiryo-machi, Aoba-ku, Sendai, Miyagi, 980-8575, Japan and [2]Faculty of Electrical and Computer Engineering, Institute of Science and Engineering, Kanazawa University, Kakuma, Kanazawa, Ishikawa, 920-1192, Japan

Associate Editor: Martin Bishop

ABSTRACT

Motivation: Many human genes express multiple transcript isoforms through alternative splicing, which greatly increases diversity of protein function. Although RNA sequencing (RNA-Seq) technologies have been widely used in measuring amounts of transcribed mRNA, accurate estimation of transcript isoform abundances from RNA-Seq data is challenging because reads often map to more than one transcript isoforms or paralogs whose sequences are similar to each other.

Results: We propose a statistical method to estimate transcript isoform abundances from RNA-Seq data. Our method can handle gapped alignments of reads against reference sequences so that it allows insertion or deletion errors within reads. The proposed method optimizes the number of transcript isoforms by variational Bayesian inference through an iterative procedure, and its convergence is guaranteed under a stopping criterion. On simulated datasets, our method outperformed the comparable quantification methods in inferring transcript isoform abundances, and at the same time its rate of convergence was faster than that of the expectation maximization algorithm. We also applied our method to RNA-Seq data of human cell line samples, and showed that our prediction result was more consistent among technical replicates than those of other methods.

Availability: An implementation of our method is available at http://github.com/nariai/tigar

Contact: nariai@megabank.tohoku.ac.jp

Supplementary Information: Supplementary data are available at *Bioinformatics* online.

Received on February 14, 2013; revised on June 11, 2013; accepted on June 27, 2013

1 INTRODUCTION

Alternative splicing is a biological process in which an exon can be either included or excluded, or there can be several splice sites so that it allows a gene to have multiple forms of proteins. A recent study suggested that >90% of human genes undergo alternative splicing, and that many alternative splicing events vary among tissues (Wang *et al.*, 2008). It has also been reported that some aberrant splicing results in human diseases, such as cystic fibrosis and spinal muscular atrophy (Garcia-Blanco *et al.*, 2004). Hence, it is important to identify transcript isoforms that are expressed in particular tissues under some conditions, and not in others.

To quantify transcription levels, RNA-Seq that uses high-throughput sequencing of cDNA has been widely used. Advantages of RNA-Seq over other conventional methods, such as DNA microarrays, include precise measurements of levels of transcripts and splice patterns because the sequence-based approach directly determines mRNA sequences (Wang *et al.*, 2009). Probabilistic approaches have been proposed to compute the relative frequencies of alternative splice isoforms for each gene, but these methods do not handle reads that map to multiple gene loci, and hence do not estimate global transcript abundances for each isoform (Jiang and Wong, 2009; Katz *et al.*, 2010). In most cases, reads are often shorter than a full transcript, and sequences of alternatively spliced transcript isoforms and paralogs are often similar to each other (Mortazavi *et al.*, 2008). Hence, short reads generated by RNA-Seq do not always map uniquely to reference cDNAs, which lead to inaccurate estimation of transcript isoform abundances.

Several approaches have been proposed to resolve ambiguous reads that map to multiple transcript isoforms (multi-map reads). One *ad hoc* approach is to allocate fractions of multimapped reads to target transcript isoforms equally, which is implemented as the default option in Cufflinks (Trapnell *et al.*, 2010). Another approach is to allocate fractions of the reads in proportion to the coverage of uniquely mapped reads divided by the length of the transcript isoforms ('rescue' method) (Mortazavi *et al.*, 2008), which is implemented as the '-u' option in the latest version of Cufflinks. Statistical methods that use a generative model of RNA-Seq data have been proposed, in which the transcript isoform abundance is estimated as a latent random variable by the expectation maximization (EM) algorithm (Li and Dewey, 2011; Li *et al.*, 2010; Nicolae *et al.*, 2011). Although Li *et al.* showed that the statistical methods performed better than the rescue method (Mortazavi *et al.*, 2008), their methods cannot handle the gapped alignment of reads, i.e. an alignment with insertions into reference sequences or deletions from reference sequences. When reads generated from high-throughput sequencers become longer with some insertion or deletion errors, it will be more difficult to map the reads to reference sequences without allowing gapped alignment. Hence, the limitation of methods that are unable to handle gapped alignments of reads is a huge drawback when more sophisticated mapping tools such as Bowtie2

*To whom correspondence should be addressed.

(Trapnell *et al*., 2012) or Novoalign (www.novocraft.com) are considered.

Also, the EM algorithm used in these statistical methods tends to overfit the model parameters to the RNA-Seq data, i.e. the methods try to estimate as many transcript isoforms as possible, as long as the likelihood function increases. It is problematic, for example, when there are many transcript isoforms whose sequences are similar to the one that is actually expressed, and the read sequences contain base substitution, insertion and deletion errors. As a result of the maximum likelihood estimation, the method might predict false positives ('spurious' transcript isoforms), and hence, the transcript abundances estimated for true isoforms are affected. One strategy to avoid such overfitting is to consider a model selection framework based on the marginal likelihood (Cooper and Herskovits, 1992), where a trade-off of matching of sequence reads and the number of isoforms can be considered. However, this involves integrating over all latent variables as well as model parameters, which is usually difficult and intractable to compute.

In this article, we propose a statistical method named TIGAR, a transcript isoform abundance estimation method with gapped alignment of RNA-Seq data by variational Bayesian (VB) inference. The proposed method can handle gapped alignment of reads to reference sequences, allowing mismatches, insertions and deletions. Our VB inference approach resolves the issue of overfitting by optimizing the number of transcript isoforms under the model selection framework (Beal and Ghahramani, 2006). The VB inference approximates full posterior distributions by assuming the factorization of latent variables and model parameters (Jordan *et al*., 1999). An approximation of the posterior distribution is obtained via an iterative EM-like algorithm with a closed-form solution in each iteration, and its convergence is guaranteed (Attias, 1999). We show from simulation experiments that the proposed method successfully predicts transcript isoform abundances more accurately than other comparable methods, and at the same time, its rate of convergence is faster than that of the EM algorithm. We also apply our method to RNA-Seq data of human cell line samples, and show that the prediction result obtained by the proposed method is more consistent than those obtained by other methods.

2 METHODS

2.1 Generative model of RNA-seq data

We use the directed graphical model (Bayesian network) for a generative model of RNA-seq data (Fig. 1), which is an extended version of the models proposed in the past literature (Li *et al*., 2010) so that gapped alignment of reads can be handled. For simplicity, we describe the model that handles single-end reads in the following sections. For details of handling paired-end reads in the model, see the Supplementary Material. The model generates N independent and identically distributed reads of length L, and each read n is represented by the random variable R_n. Each read is associated with four latent variables T_n, S_n, O_n and A_n, which represent the transcript isoform choice, transcript start position, read orientation and alignment state, respectively. Each read is also associated with a random variable Q_n, which represents the set of quality scores at all positions of the read. The abundance of each transcript isoform (a fraction of the total mRNA) is represented by the parameter vector $\boldsymbol{\theta} = (\theta_0, \ldots, \theta_T)'$, where $\sum_{t=0}^{T} \theta_t = 1$ and T is the number of transcript isoforms (e.g. a total number of cDNA sequences in the RefSeq database). Here, $\boldsymbol{\theta}_0$ represents the noise isoform abundance, from which reads that do not map to any other known isoforms are generated. The random variable $T_n = t$ indicates that read n is generated from the transcript isoform t. The random variable $S_n = s$ indicates that read n is generated from the start position s of the transcript isoform. The random variable $O_n = 0$ indicates that read n is generated from the sense strand (read n maps to the sense strand of the reference sequence), whereas $O_n = 1$ indicates read n is generated from the anti-sense strand (the reverse complement of read n maps to the sense strand of the reference sequence). The random variable A_n represents the alignment state (either alignment match, insertion or deletion at each alignment position) between read n and reference t, encoded in the CIGAR string format (http://samtools.sourceforge.net) by mapping tools. For example, if $A_n =$ '10M3D15M', then the first 10 positions of read n are matches or mismatches with the reference sequence, the next three are nucleotide deletions from the reference sequence and the last 15 are matches or mismatches with the reference sequence. The random variable Q_n represents the set of quality scores at all position of read n, indicated by Phred quality scores (Cock *et al*., 2010).

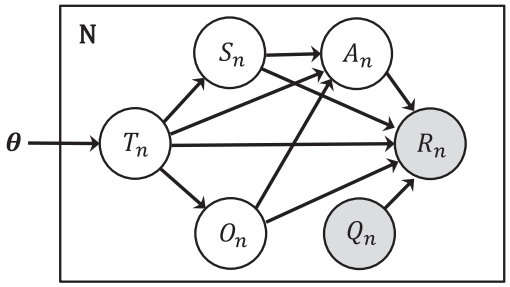

Fig. 1. The generative model for RNA-Seq data used by TIGAR. The transcript isoform abundance, transcript isoform choice, read start position, orientation, alignment state, quality score and sequence of read n are represented by $\boldsymbol{\theta}, T_n, S_n, O_n, A_n, Q_n$ and R_n, respectively

We impose the conditional independence assumptions as indicated in Figure 1, and the conditional likelihood for this model is decomposed as the product of conditional probabilities:

$$P(T_n, S_n, O_n, A_n, R_n, Q_n|\boldsymbol{\theta}) = P(T_n|\boldsymbol{\theta})P(S_n|T_n)P(O_n|T_n)P(A_n|T_n, S_n, O_n) \times P(R_n|T_n, S_n, O_n, A_n, Q_n)P(Q_n).$$

$P(T_n|\boldsymbol{\theta})$ represents the probability that read n is generated from transcript isoform $T_n = t$ given the parameter vector. We compute the transcript isoform choice given the parameter vector as $P(T_n = t|\boldsymbol{\theta}) = \theta_t$.

$P(S_n|T_n)$ represents the probability of the start position of read n given the transcript isoform choice. We assume a uniform distribution of start positions (Li *et al*., 2010) and calculate $P(S_n = s|T_n = t) = 1/l_t$ if mRNAs have poly(A) tails, and $P(S_n = s|T_n = t) = 1/(l_t - L + 1)$ if mRNAs do not have poly(A) tails, where l_t is the length of the transcript isoform t.

$P(O_n|T_n)$ represents the probability of the orientation of read n given the transcript isoform choice. For a strand specific protocol, it can be set as $P(O_n = 0|T_n = t) = 1$ and $P(O_n = 1|T_n = t) = 0$. Otherwise, it can be set as $P(O_n = 0|T_n = t) = P(O_n = 1|T_n = t) = 0.5$ or can be estimated empirically from the RNA-Seq data.

$P(A_n|T_n, S_n, O_n)$ represents the probability of the alignment state of read n given the transcript isoform choice, start position and orientation of read n. Based on the CIGAR string format, we construct the alignment state sequence $a[x]$ at each alignment position x as either 'M' (match or mismatch), 'I' (insertion) or 'D' (deletion). Then the conditional probability is calculated as follows:

$$P(A_n|T_n, S_n, O_n) = start(a[1]) \prod_{x=1}^{X} trans(a[x], a[x+1]),$$

where *start* is the probability of starting with $a[1]$ (from which the alignment state of read n starts), *trans* is the probability of transition from the previous alignment state to the current state and X is the total number of alignment positions. $trans(a[X], a[X+1]) = trans(a[X], \text{“}END\text{”})$ is the probability of ending with $a[X]$.

$P(R_n|T_n, S_n, O_n, A_n, Q_n)$ represents the probability of the sequence of read n, given the transcript isoform choice, start position, orientation, alignment state and quality score of the read. Based on the alignment state $a[x]$ defined earlier in the text, the read sequence character $r[x]$, either 'A', 'T', 'G', 'C' or '-' (deletion from the reference sequence), and the corresponding quality score $q[x]$, either one of the Phred quality score, or "-" (deletion from the reference sequence) are constructed. In a similar fashion, the reference sequence character $c[x]$, either 'A', 'T', 'G' ,'C' or '-' (insertion into the reference sequence) is constructed according to the alignment states of the read and the reference sequence. To summarize the latent variables T_n, S_n,O_n and A_n, we introduce an indicator random variable Z_{ntsoa}, where Z_{ntsoa} is equal to 1 if $(T_n, S_n, O_n, A_n) = (t, s, o, a)$ of read n and 0 otherwise. We calculate $P(R_n|T_n, S_n, O_n, A_n, Q_n)$ when $Z_{ntsoa} = 1$ as:

$$P(R_n|Z_{ntsoa} = 1, Q_n) = \prod_{x=1}^{X} emit(r[x], q[x], c[x], a[x]),$$

where *emit* is the emission probability calculated according to the current alignment state $a[x]$ as follows:

$$\begin{cases} emit(r[x], q[x], c[x], \text{“M”}) = subst(r[x], q[x], c[x]), \\ emit(r[x], q[x], c[x], \text{“I”}) = insert(r[x]), \\ emit(r[x], q[x], c[x], \text{“D”}) = delete(c[x]), \end{cases}$$

where *subst* is a substitution matrix constructed for each Phred quality score, *insert* is a position independent insertion probability constructed for each nucleotide character and *delete* is a position independent deletion probability constructed for each nucleotide character. For example, *subst* ('A', 'I', 'C') is the probability that we observe the nucleotide character 'A' at the aligned position of the read, given that the nucleotide character of the reference sequence is 'C' and the Phred quality score of the corresponding position of the read is 'I'.

$P(Q_n)$ represents the prior probability of the quality score and we calculate this as a constant value.

If $T_n = 0$, then the random variables S_n, O_n and A_n are assumed to be conditionally independent of T_n. Hence, we set the conditional probability of reads generated from a noise isoform as follows:

$$P(R_n|T_n = 0) = \prod_{x=1}^{L} \beta(read[x]),$$

where $read[x]$ is a nucleotide character at position x of read n, and β is a position independent background distribution (Li *et al.*, 2010).

All of the aforementioned functions, such as transition and emission probabilities, can be either calculated in advance or estimated as parameters from the RNA-Seq data. TIGAR estimates these parameters along with $\boldsymbol{\theta}$ during the first 10 iterations of the variational Bayesian inference algorithm described later in the text.

2.2 Variational Bayesian inference method

Our goal is to estimate the model parameter vector $\boldsymbol{\theta} = (\theta_0, \ldots, \theta_T)'$, which represents the transcript isoform abundances (the fraction of each isoform abundance among the total mRNA). The maximum likelihood estimate of the parameter can be calculated by the EM algorithm (see the Supplementary Material for details). On the other hand, we propose a VB inference approach for approximation of full Bayesian inference by factorization assumptions of latent variables and model parameters (Jordan *et al.*, 1999), in which model parameters and latent variables are treated as random variables.

In Bayesian inference, prior distribution is given to the model parameters, and integration of the parameters is considered. As the prior distribution of the parameter $P(\boldsymbol{\theta})$, we use the Dirichlet distribution:

$$P(\boldsymbol{\theta}) = \frac{1}{C}\prod_{t=0}^{T} \theta_t^{\alpha_t - 1},$$

where $\alpha_t > 0$ is a hyperparameter, C is a constant and $\sum_{t=0}^{T} \theta_t = 1$. If $\alpha_t \geq 1$, then $\alpha_t - 1$ can be interpreted as the effective count of prior reads that are assigned to transcript isoform t. If $\alpha_t < 1$, the prior favors that some of the transcript isoform abundances to be 0 (Biship, 2006). In this article, we assume that there is no prior knowledge about the transcript isoform abundances, and we set a single hyperparameter α_0 for all transcript isoforms. From the iterative procedure between the variational Bayesian E (VBE) and M (VBM) steps described later in text, the posterior distribution over the model parameter $\boldsymbol{\theta}$ and the latent variable $\boldsymbol{Z}$ is optimized.

2.3 Variational Bayesian inference algorithm

Given a hyperparameter α_0 for the prior distribution of $\boldsymbol{\theta}$ and RNA-Seq data $\boldsymbol{R}$, a lower bound on the model log marginal likelihood is optimized iteratively with the following procedures:

Step 1. **Initialization**

For each transcript isoform t, initialize $\alpha_t = \alpha_0$.

Step 2. **VBE step**

Using the current estimate of $E_{\boldsymbol{\theta}}[\theta_t]$, compute $E_{\boldsymbol{Z}}[Z_{ntsoa}]$.

Step 3. **VBM step**

Using the current estimate of $E_{\boldsymbol{Z}}[Z_{ntsoa}]$, compute $E_{\boldsymbol{\theta}}[\theta_t]$.

Step 4. **Check for a stopping criterion**

If any of the $E_{\boldsymbol{\theta}}[\theta_t]$ has been changed more than a pre-specified threshold, return to Step 2.

Each cycle of the VB inference algorithm increases the lower bound on the log marginal likelihood of the model, and convergence under a stopping criterion of the aforementioned procedure is guaranteed (Attias, 1999). For a stopping criterion, a relative change of 10^{-3} for an isoform whose abundance parameter $\theta_t > 10^{-7}$ is used as the threshold for each $E_{\boldsymbol{\theta}}[\theta_t]$ (Li and Dewey, 2011).

In Step 2, $E_{\boldsymbol{Z}}[Z_{ntsoa}]$ is calculated by using the current estimate of the posterior distribution over $\boldsymbol{\theta}$

$$E_{\boldsymbol{Z}}[Z_{ntsoa}] = \begin{cases} \frac{\rho_{ntsoa}}{\sum_{(t's'o'a') \in \pi_n} \rho_{nt's'o'a'}}, (t, s, o, a) \in \pi_n \\ 0, \ otherwise \end{cases},$$

where π_n is defined as the set of all (t, s, o, a) tuples for possible alignments of read n, and

$$\begin{aligned} \log \rho_{ntsoa} = {} & E_{\boldsymbol{\theta}}[\log \theta_t] + \log P(S_n|T_n) + \log P(O_n|T_n) \\ & + \log P(A_n|T_n, S_n, O_n) + \log P(R_n|Z_{ntsoa} = 1, Q_n), \\ E_{\boldsymbol{\theta}}[\log \theta_t] = {} & \psi(\alpha_t) - \psi\Big(\sum_t \alpha_t\Big), \end{aligned}$$

where ψ is the digamma function.

In Step 3, $E_{\boldsymbol{\theta}}[\theta_t]$ is calculated by using the current estimate of the posterior distribution over $\boldsymbol{Z}$ as follows:

$$E_{\boldsymbol{\theta}}[\theta_t] = \frac{\alpha_t}{\sum_t \alpha_t},$$

where

$$\alpha_t = \alpha_0 + \sum_{n,s,o,a} E_{\boldsymbol{Z}}[Z_{ntsoa}].$$

For details of derivations of the equations and how to calculate the lower bound of the log marginal likelihood, see the Supplementary Material.

3 RESULTS

3.1 Simulation experiments

To assess the performance of the proposed method in quantifying transcript isoforms, we prepared synthetic single-end RNA-Seq data from human reference cDNA sequences in the RefSeq database (Pruitt *et al.*, 2007). Because it has been shown that expression levels roughly follow log-normal distributions (Nicolae *et al.*, 2011), 10 000 transcript isoforms were randomly chosen such that their transcript abundance parameters follow a log-normal distribution. It turned out that these isoforms were generated from 7771 gene locus in total, of which 1586 genes have more than one transcript isoform. For this experiment, 3.0 million, 2.1 million, 1.4 million, 1.05 million and 0.42 million mRNA reads of 35, 50, 75, 100 and 250 bp, respectively, were synthetically generated independently from the cDNA sequences in the RefSeq database. Start positions of reads were chosen randomly across each transcript. The total number of nucleotides (the total throughput of base pairs) was the same for each experiment. These reads were generated with 1% substitution, 1% insertion and 1% deletion errors randomly at each nucleotide position of the read sequences. We used the Bowtie2 software (Langmead *et al.*, 2009) to align the reads to the reference sequences with the '–very-sensitive' option specified. Then we estimated transcript isoform abundances with TIGAR by variational Bayesian inference and the EM algorithm, called TIGAR (VB) and TIGAR (EM), respectively. To compare quantification results with TIGAR, we used RSEM v1.1.20 (Li and Dewey, 2011) that implements the EM algorithm after mapping with Bowtie. We also used Cufflinks (Trapnell *et al.*, 2010) with the '-u' and TopHat2 '-G' and '–no-novel-juncs' options with using Bowtie2, so that it handles multimap reads and does not predict novel transcripts. To choose an appropriate hyperparameter α_0 for TIGAR (VB), we calculated the lower bound of the log marginal likelihood of the model given α_0 (see the Supplementary Material for the calculation in detail). The lower bound of the log marginal likelihood of the model was maximized when $\alpha_0 = 0.1$ for the 3.0 million reads of 35 bp (Fig. 2). Similarly, $\alpha_0 = 0.1$ was also selected for all the read sets of 50, 75, 100 and 250 bp because it maximized the lower bound of the log marginal likelihood for all cases.

Figure 3 shows the number of transcript isoforms predicted with TIGAR (VB) for varying α_0, and TIGAR (EM). Please see the Supplementary Figure S1 for the zoomed-in region of the first 1000 iterations. We can see from the results that TIGAR (VB) converged faster than TIGAR (EM), although the total predicted number of isoforms resulted in different levels according to the α_0. Figure 4 shows the number of iterations until convergence at the different read lengths with TIGAR (VB), TIGAR (EM) and RSEM. It turned out that TIGAR (VB) required the least number of iterations until convergence, regardless of the read length. The only difference between VB inference and the EM algorithm in terms of computation cost at each iteration step is the evaluation of $E_{\theta}[\log \theta_t]$ in Step 2 of the

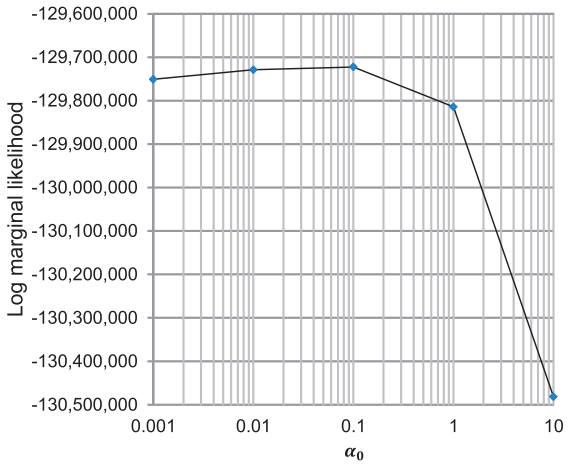

Fig. 2. Choosing the hyperparameter α_0 by calculating the lower bound of the log marginal likelihood of the model. In this case, $\alpha_0 = 0.1$ is selected as the maximizer

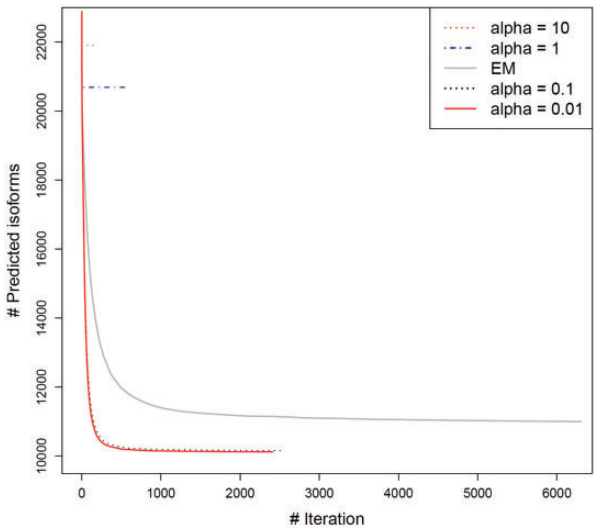

Fig. 3. The number of predicted transcript isoforms at each iteration step by VB inference for given α_0, and by the EM algorithm. TIGAR (VB) converged faster than with TIGAR (EM). The smaller α_0 resulted in prediction of fewer transcript isoforms

inference algorithm. The computation cost of this calculation is $O(T)$, where T is the total number of isoforms. The total computation cost in Step 2 is $O(M + T)$, where M is the total number of mappings. Because the number of reads is usually much larger than the total number of transcripts, $O(M + T) \approx O(M)$, and hence, the same as the EM algorithm. Figure 5 shows the total CPU time with TIGAR (VB) and that of TIGAR (EM) for varying read lengths. We can see that the fewer iterations until convergence resulted in less computation time.

The transcript abundance was measured as fragments per kilobase of exon per million mapped fragments (FPKM) (Trapnell *et al.*, 2010). We kept transcript isoform predictions whose

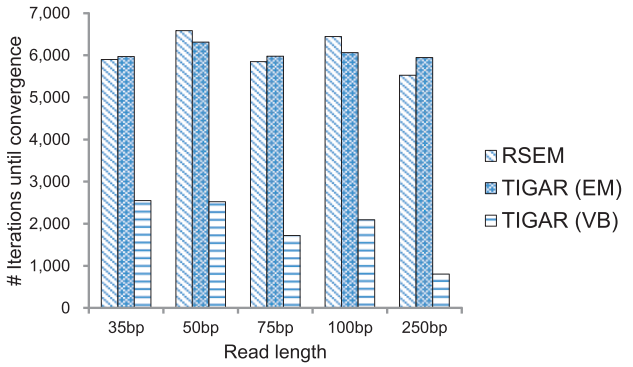

Fig. 4. Number of iterations until convergence with TIGAR (VB), TIGAR (EM) and RSEM

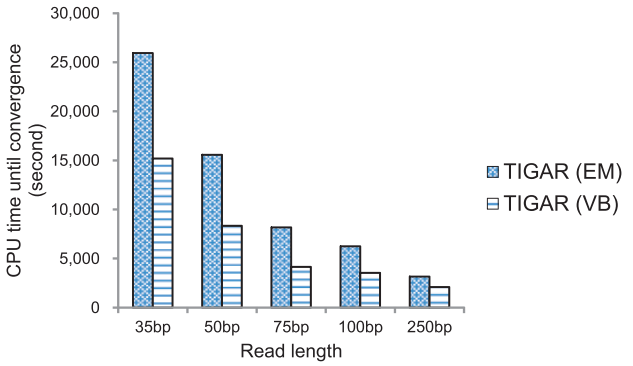

Fig. 5. The total CPU time until convergence with TIGAR (VB) and TIGAR (EM)

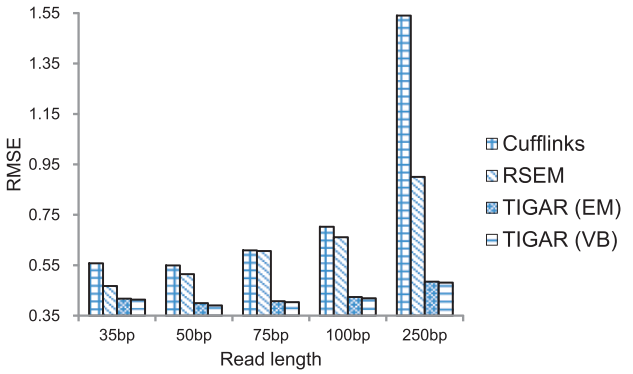

Fig. 6. Root mean square error of the abundance predicted with TIGAR (VB), TIGAR (EM), RSEM and Cufflinks

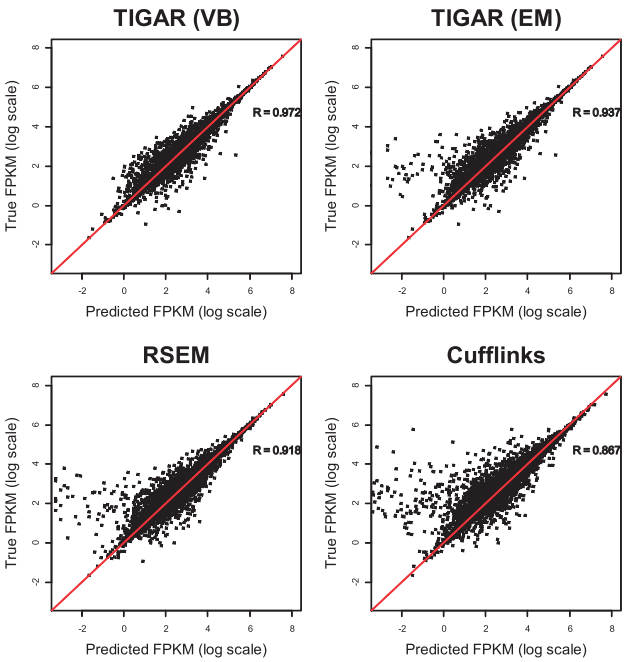

Fig. 7. Scatter plots and the Pearson correlation coefficient of FPKM predicted with TIGAR (VB) (top-left), TIGAR (EM) (top-right), RSEM (bottom-left) and Cufflinks (bottom-right)

FPKMs > 0.01. The root mean square error (RMSE) of log(FPKM+1) between predicted and true abundances is calculated for comparison. The Supplementary Figure S2 shows the RMSE obtained by TIGAR (VB) at each α_0 and by TIGAR (EM). It can be seen from the results that TIGAR (VB) performed better than TIGAR (EM), when α_0 was <1. We hypothesized from the results that the beneficial effect of α_0 on the convergence speed came from optimizing the number of transcript isoforms given appropriate prior distributions by VB inference, which in this case also lead to better prediction performance in terms of the RMSE by eliminating spurious predictions in the early iterations.

Figure 6 shows a summary of prediction performances, which is a comparison between true and predicted abundances based on RMSE with varying read lengths. Overall, both TIGAR (VB) and TIGAR (EM) performed well compared with RSEM and Cufflinks. When the length of reads became longer, prediction performances by RSEM and Cufflinks became worse. This is likely because TIGAR (VB) or TIGAR (EM) was able to handle gapped alignments, which resulted in using more reads with insertion or deletion errors. At the same time, TIGAR benefited from longer reads, which were expected to map to the reference sequence more uniquely. When reads were generated with 0.1% substitution errors and without insertion or deletion error, the performances between TIGAR and RSEM were similar (see the Supplementary Fig. S3). The relatively poor performance of Cufflinks might come from the fact that it uses the rescue strategy (Mortazavi *et al.*, 2008), which roughly corresponds to one iteration of the EM algorithm. In addition, Cufflinks uses a mapping software called TopHat2 (Trapnell *et al.*, 2009) to map reads against the reference genome, instead of against cDNA sequences, by chopping reads at prefixed intervals (such as 25 bp), which might lead to false-positive mappings of reads to exons and their junctions. This result suggested that the optimization and elimination of spurious isoforms were successfully performed by VB inference. Figure 7 shows scatter plots of the true and predicted FPKM of each transcript isoform for 35 bp reads (see the Supplementary Figs S4–S7 for 50, 75, 100 and 250 bp, respectively). Because TIGAR (EM), RSEM and

Cufflinks try to predict as many transcript isoforms as possible, they tend to predict more false positives whose sequences were similar to the one that was actually expressed. From the correlation coefficient between the true and predicted FPKM in Figure 7, TIGAR (VB) successfully obtained accurate estimates of FPKM, whereas others underestimated transcription levels of many isoforms.

Similarly, we observed better performances with TIGAR (VB) over other methods when 3.0 million, 2.1 million, 1.4 million, 1.05 million and 0.42 million paired-end reads of 35, 50, 75, 100 and 250 bp, respectively, were synthetically generated with fragment size distributions with a mean $\mu = 600$ and a standard deviation of $\sigma = 40$ (see the Supplementary Figs S8–S11).

3.2 Real data analysis

To evaluate the proposed method with real data, we analyzed two datasets obtained from different sequencing technologies. First, we analyzed single-end RNA-Seq data of human cell line samples (HeLa) that is publicly available from the Ion Community Web site (http://ioncommunity.lifetechnologies.com). In the study, the sequencing was performed with the Ion Total RNA-Seq Kit v2 and the Ion PGM sequencer. For this analysis, 4.5 million reads were obtained. To evaluate the prediction performance of each method using the real data, we prepared technical replicates named Samples 1 and 2, each with 3.0 million reads of 35 bp, and 1.05 million reads of 100 bp, which were randomly obtained from the original data without replacement. We used the Bowtie2 software (Langmead *et al.*, 2009) to align the reads to the cDNA sequences with the '–very-sensitive' option specified. Figure 8 shows scatter plots of FPKM for each transcript isoform predicted by TIGAR (VB and EM), RSEM and Cufflinks. The hyperparameter α_0 was set to 0.1 as a maximizer of the lower bound of the log marginal likelihood for these data. It can be seen from the result that TIGAR (VB) predicted abundances of transcript isoforms more consistently than the other methods. For example, Cufflinks predicted many transcript isoforms whose FPKM values are relatively small (i.e. <1), as shown in Figure 9. Although it will be difficult to validate such tiny abundances of the transcript isoforms, we conclude that TIGAR (VB) was most consistent in estimating transcript isoform abundances between the two technical replicates constructed from the same sample.

Second, we analyzed paired-end RNA-Seq data of K562 human cell line samples, which were generated by the ENCODE Project Consortium and publicly available from the Web site (http://hgdownload.cse.ucsc.edu/goldenPath/hg19/encodeDCC/wgEncodeCaltechRnaSeq/). For this analysis, we analyzed two technical replicates, each with 12.8 and 13.0 million paired-end reads of 75 bp, obtained using the Illumina Genome Analyzer IIx. The software parameters were set the same as the previous experiment. The hyperparameter α_0 was set to 0.1 as a maximizer of the lower bound of the log marginal likelihood for these data. The Supplementary Figure S12 shows scatter plots and the Pearson correlation coefficient of FPKM predicted with TIGAR (VB), TIGAR (EM), RSEM and Cufflinks. The Pearson correlation coefficient of FPKM between the two technical replicates with TIGAR (VB) was 0.975, whereas those with TIGAR (EM), RSEM and Cufflinks were 0.970, 0.969 and 0.954, respectively. In addition to the better prediction performance with TIGAR (VB), its rate of convergence was faster than those with TIGAR (EM) and RSEM. The number of iteration steps until convergence with TIGAR (VB) was 1735, whereas those with TIGAR (EM) and RSEM were 4249 and 4701,

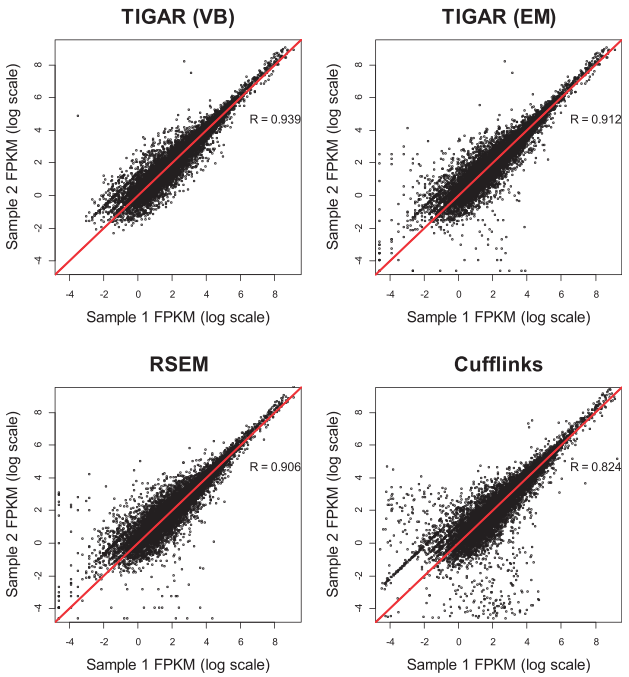

Fig. 8. Scatter plots and the Pearson correlation coefficient of FPKM predicted with TIGAR (VB), TIGAR (EM), RSEM and Cufflinks

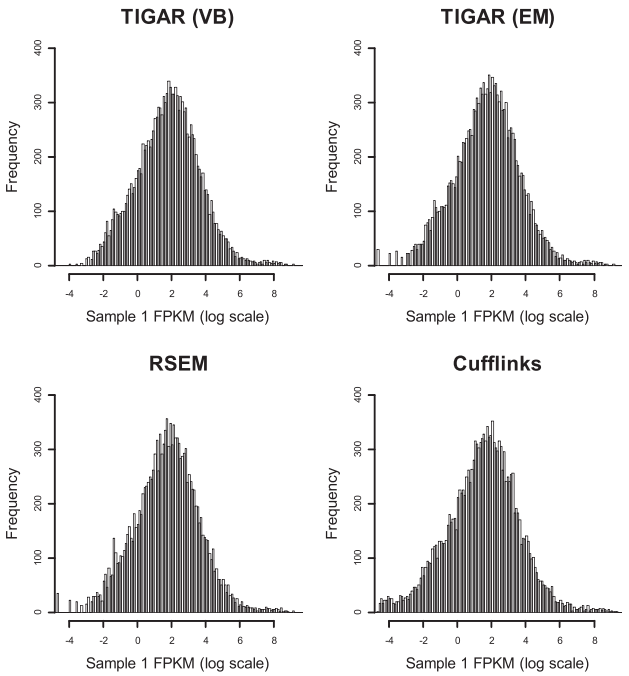

Fig. 9. Histograms of FPKM predicted with TIGAR (VB), TIGAR (EM), RSEM and Cufflinks

respectively. In terms of the CPU time until convergence, TIGAR (VB) was faster by 31.6% compared with the TIGAR (EM).

4 DISCUSSION

In this article, to estimate transcript isoform abundances from RNA-Seq data, we proposed a statistical method, named TIGAR, which can handle gapped alignments of reads against reference sequences. To avoid overfitting, we also proposed VB inference that iteratively optimizes the number of transcript isoforms. We showed through simulation experiments that in quantifying transcript isoform abundances from RNA-Seq data, TIGAR (VB) performed better than the popularly used Cufflinks and RSEM, especially when the length of reads became longer with some insertion and deletion errors. We also showed that the convergence speed of TIGAR (VB) was faster than those of TIGAR (EM) and RSEM, which is likely explained by the optimization of the number of transcript isoforms in the early iterations. We also applied our method to real data from human cell line samples and showed that the prediction result by our method was more consistent than those by other methods in estimating transcript abundances of technical replicates. Notably, TIGAR (VB) made fewer predictions of transcript isoforms whose FPKM were small (e.g. <1), in comparison with predictions by Cufflinks and RSEM, many of which we believe might be false positives. It is also important to consider experimental noise introduced during library preparation under different conditions. To handle such variations among multiple replicates or experimental conditions, methods such as Cuffdiff (Trapnell *et al*., 2012) or EBSeq (Leng *et al*., 2013) can be considered in a downstream analysis. The performance of TIGAR will become more significant when it is applied to RNA-Seq data produced with new types of sequencers, such as the Ion PGM and Pacific Biosciences' RS, whose reads are longer but with relatively many insertion and deletion errors (Quail *et al*., 2012).

Our TIGAR (VB) optimizes the number of transcript isoforms by maximizing the lower bound of log likelihood for RNA-Seq data. On the contrary, other model selection methods such as the Bayesian information criterion (BIC) (Schwarz, 1978) or a sampling-based approach can be considered. However, it has been reported that the variational lower bounds outperform the BIC in finding the correct model structure, while approaching the performance of the much more costly sampling procedure (Beal and Ghahramani, 2003, 2006). Hence, we believe that the proposed method offers the advantage of Bayesian inference to avoid overfitting but keeps the computational speed comparable with that of the EM algorithm.

A useful application of TIGAR (VB) might be the quantification of novel alternative spliced isoforms or fusion transcripts from RNA-Seq data by augmenting reference sequences to include novel splice junctions (Trapnell *et al*., 2009). Another useful application might be fusion transcripts predicted from discordantly mapped mate pairs (Kinsella *et al*., 2011). In such cases, Because the number of possible splice junctions or combinations of exons increases dramatically, the proposed method will be helpful to select a set of plausible transcript isoforms so that predictions can be further validated. Alternatively, if we know in advance from other experiments that certain transcript isoforms expressed more than others, then the proposed method might be able to incorporate the prior information by setting the hyperparameter α_0 as an effective read count. Hence, hyperparameters can be fixed according to prior knowledge or α_0 can be chosen as a maximizer of the lower bound of the log marginal likelihood, as described in the article.

The current model cannot distinguish between sequencing errors and the alignment mismatches because of single nucleotide polymorphisms (SNPs) or structural variations inherent in samples. It will be useful to incorporate structural variation information in the dbSNP (http://www.ncbi.nlm.nih.gov/projects/SNP/). Our future work will include investigations of topics mentioned earlier in text.

ACKNOWLEDGEMENTS

The super-computing resource was provided by Human Genome Center, Institute of Medical Science, University of Tokyo.

Funding: This work was supported (in part) by MEXT Tohoku Medical Megabank Project.

Conflict of Interest: none declared.

REFERENCES

Attias,H. (1999) Inferring parameters and structure of latent variable models by variational bayes. In: *Proceedings of the Fifteenth Conference on Uncertainty in Artificial Intelligence*. pp. 21–30. Morgan Kaufmann, San Francisco, CA.

Beal,M.J. and Ghahramani,Z. (2003) The variational Bayesian EM algorithm for incomplete data: with application to scoring graphical model structures. *Bayesian Stat*., 7.

Beal,M.J. and Ghahramani,Z. (2006) Variational Bayesian learning of directed graphical models with hidden variables. *Bayesian Anal*., **1**, 793–831.

Biship,C.M. (2006) *Pattern Recognition and Machine Learning*. Science+Business Media, LLC, New York, NY, USA.

Cock,P.J. *et al*. (2010) The Sanger FASTQ file format for sequences with quality scores, and the Solexa/Illumina FASTQ variants. *Nucleic Acids Res*., **38**, 1767–1771.

Cooper,G. and Herskovits,E. (1992) A Bayesian method for the induction of probabilistic networks from data. *Mach. Learn*., **9**, 309–347.

Garcia-Blanco,M.A. *et al*. (2004) Alternative splicing in disease and therapy. *Nat. Biotechnol*., **22**, 535–546.

Jiang,H. and Wong,W.H. (2009) Statistical inferences for isoform expression in RNA-Seq. *Bioinformatics*, **25**, 1026–1032.

Jordan,M.I. *et al*. (1999) An introduction to variational methods for graphical models. *Mach. Learn*., **37**, 183–233.

Katz,Y. *et al*. (2010) Analysis and design of RNA sequencing experiments for identifying isoform regulation. *Nat. Methods*, **7**, 1009–1015.

Kinsella,M. *et al*. (2011) Sensitive gene fusion detection using ambiguously mapping RNA-Seq read pairs. *Bioinformatics*, **27**, 1068–1075.

Langmead,B. *et al*. (2009) Ultrafast and memory-efficient alignment of short DNA sequences to the human genome. *Genome Biol*., **10**, R25.

Leng,N. *et al*. (2013) EBSeq: an empirical Bayes hierarchical model for inference in RNA-seq experiments. *Bioinformatics*, **29**, 1035–1043.

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

Li,B. *et al*. (2010) RNA-Seq gene expression estimation with read mapping uncertainty. *Bioinformatics*, **26**, 493–500.

Mortazavi,A. *et al*. (2008) Mapping and quantifying mammalian transcriptomes by RNA-Seq. *Nat. Methods*, **5**, 621–628.

Nicolae,M. *et al*. (2011) Estimation of alternative splicing isoform frequencies from RNA-Seq data. *Algorithms Mol. Biol*., **6**, 9.

Pruitt,K.D. *et al.* (2007) NCBI reference sequences (RefSeq): a curated non-redundant sequence database of genomes, transcripts and proteins. *Nucleic Acids Res.*, **35**, D61–D65.

Quail,M. *et al.* (2012) A tale of three next generation sequencing platforms: comparison of ion torrent, Pacific biosciences and Illumina MiSeq sequencers. *BMC Genomics*, **13**, 341.

Schwarz,G. (1978) Estimating the dimension of a model. *Ann. Stat.*, **6**, 461–464.

Trapnell,C. *et al.* (2009) TopHat: discovering splice junctions with RNA-Seq. *Bioinformatics*, **25**, 1105–1111.

Trapnell,C. *et al.* (2012) Differential gene and transcript expression analysis of RNA-seq experiments with TopHat and Cufflinks. *Nat. Protoc.*, **7**, 562–578.

Trapnell,C. *et al.* (2010) Transcript assembly and quantification by RNA-Seq reveals unannotated transcripts and isoform switching during cell differentiation. *Nat. Biotechnol.*, **28**, 511–515.

Wang,E.T. *et al.* (2008) Alternative isoform regulation in human tissue transcriptomes. *Nature*, **456**, 470–476.

Wang,Z. *et al.* (2009) RNA-Seq: a revolutionary tool for transcriptomics. *Nat. Rev. Genet.*, **10**, 57–63.