PseudoLasso: Leveraging Read Alignment in Homologous Regions to Correct Pseudogene Expression Estimates via RNASeq

Chelsea J.-T. Ju
Department of Computer Science
University of California, Los Angeles
California 90095, USA
chelseaju@ucla.edu

Zhuangtian Zhao
Department of Computer Science
University of California, Los Angeles
California 90095, USA
zztt5511@gmail.com

Wei Wang
Department of Computer Science
University of California, Los Angeles
California 90095, USA
weiwang@cs.ucla.edu

ABSTRACT

Pseudogenes have long been considered to be nonfunctional segments in the genome, but recent studies have provided evidence to support their novel regulatory roles in biological processes. With the growing interests in pseudogene research, scientists rely on RNA sequencing technology to estimate expression level of pseudogenes at different tissues or cell lines. The major challenge of RNASeq on pseudogene quantification falls in the high sequence similarity between pseudogenes and their homologous parents. Reads can be ambiguously aligned to multiple homologous regions. In this article, we present PseudoLasso, a genome-wide approach to accurately estimate the abundance of pseudogenes and their parents, and correctly align reads to their origins. Our approach focuses on learning read alignment behaviors, and leveraging this knowledge for abundance estimation and alignment correction. Compared to the read count estimates reported by TopHat2, PseudoLasso is able to provide estimates with a reduced error rate of 10-fold.

Categories and Subject Descriptors

G.3 [**Probability and Statistics**]: Correlation and regression analysis; J.3 [**Life and Medical Sciences**]: Biology and genetics

General Terms

Bioinformatics

Keywords

Pseudogene, L1 Regularization, RNASeq

1. INTRODUCTION

BCB'14, September 20–23, 2014, Newport Beach, CA, USA.

http://dx.doi.org/10.1145/2649387.2649447.

Prior to the ENCODE project, pseudogenes had hitherto been considered to be dysfunctional genomic segments, and were only of interests to suggest evolutionary evidence of ancient molecules encoded by the genome. Recent studies have revealed that many pseudogenes are transcribed and play different important regulatory roles in biological processes [6, 12, 17, 23, 24]. This rising evidence has broadened the range of interests in pseudogene-related research, which includes elucidating their functional impacts and quantifying their transcript abundance. Currently, genome-wide studies of pseudogenes have been extensively focused on identifying their genomic locations. The knowledge from these studies are comprehensively integrated into publicly available databases, such as the GENCODE pseudogene resource [15] and Yale pseudogene resource [7]. On the other hand, genome-wide analyses of pseudogene expression remain challenging due to the highly sequence similarity of their homologous protein-coding genes.

In this article, we first present the novel biological functions of pseudogenes. These findings have fostered the growing attentions in pseudogene expression profiling. We then discuss the RNA Sequencing technology to estimate transcript abundance, and address its challenges on pseudogene analysis. Since the bottleneck lies on high sequence similarity between a pair of pseudogene and its protein-coding gene, we propose a linear regression model to learn the behavior of reads from any given gene aligning to different regions with sequence homology. Subsequently, we leverage this behavior to reassign read alignments to acquire a more accurate transcripts quantification.

1.1 Pseudogene

Pseudogenes arise from duplication of a set of protein-coding genes, and they can be categorized into two forms, unprocessed and processed, based on their copying mechanisms [21]. Unprocessed pseudogenes derive from a direct genomic duplication of gene, and thus retain the intron-exon structure. Processed pseudogenes are the products of retrotransposition, where the mRNA transcripts of original genes are reverse transcribed and reintegrated into the genome at new locations. As a result, processed pseudogenes lose the introns and the 5' -end promoter sequence. Regardless of the mechanism, they all bear a substantial amount of mutations over time, which may cause deficiencies in gene expression

and protein coding. For this reason, they are often labeled as "junk" DNA.

Referencing the ENCODE project, Zheng *et al.*[23] performed an extensive analysis to demonstrate that at least a fifth of the 201 ENCODE pseudogenes were transcribed. In addition, many gene-by-gene functional assays through knockdown and overexpression experiments have suggested that pseudogenes may interact with functional protein coding genes, and regulate different biochemical processes in cells [5, 6]. Pink *et al.*[16] have summarized these potential mechanisms into three models:

1. Antisense transcripts generated from pseudogenes can combine with sense-stranded mRNA from a homologous coding gene, and interfere with the translation of the coding gene.

2. Pseudogene transcripts can be processed into small interfering RNA (siRNA), and silence the expression of other functional genes.

3. The sense strand of pseudogene transcripts can behave as a competing endogenous RNA (ceRNA) to its homologous functional gene in *cis*. It competes with the functional gene for a shared *trans*-acting molecule, either a stability factor or a miRNA for degradation, and thus regulates the stability of the functional gene.

Advancing to genome-wide studies of pseudogenes, microarray and the next generation sequencing (NGS) technology provide a high-throughput screening for gene expression profiling. However, the design of the microarray chip presents a limitation for studying pseudogenes, as many pseudogene probes are often missing from commercially available arrays [16]. RNA sequencing yields a more unbiased approach, but current aligners fall short on correctly assigning reads among high sequence similarity regions.

1.2 RNASeq Challenges

RNASeq experiments start with converting purified and sheared RNA molecules to cDNA. cDNA fragments are sequenced and read out by sequencers, producing millions of short reads, ranging from 50bp to 300bp in length. The fragment can be read from one end only, single-end sequencing, or from both ends, paired-end sequencing. Following a series of data analysis, the amount of reads from a transcript can be used to estimate the expression abundance.

Data analysis for most of the current RNASeq pipeline can be divided into three stages [4]: read mapping, transcriptome reconstruction, and expression quantification. Millions of short reads are first aligned either to a reference transcriptome or a genome. In transcriptome reconstruction, overlapped alignments from the first stage are aggregated and assembled into transcripts. The abundance of each transcript is quantified based on the number of mapped reads, and normalized to account for transcript size and total number of mapped reads. Statistical analysis can be further applied to identify significant changes in gene expression across different experiments. Intuitively, each stage has a cascading effect, and the outcome from read mapping can predominantly change the results of any downstream analysis. Thus, we focus on the computational challenges of read mapping.

Paralogous and homologous genes contain low-complexity sequences within their families, and as a result, reads from these genes may align equally well to multiple places, known as the "multireads". Similarly, pseudogenes share a high sequence similarity with the functional genes they are derived from. In the case of processed pseudogenes, which contain uninterrupted sequence without introns, reads from spliced transcripts of their functional genes can yield a better alignment at these pseudogene regions. In Figure 1, we use simulated data to illustrate this issue on the Tuxedo Pipeline [19]. Paired-end reads are generated from one of the human transcripts for diacylglycerol kinase (*DGKZ*), ENST00000421244, with two different coverages, 10X and 30X. Reads are aligned by TopHat2 [8], and the abundance is estimated by Cufflinks [20]. PGOHUM00000248578 is the processed pseudogenes of this transcript, and is not expressed in our simulation. Cufflink reports a relatively high abundance in terms of FPKM for this processed pseudogene compared to *DGKZ*. As we observe from the read alignments, most of the reads are assigned to the pseudogene.

1.3 Related Work

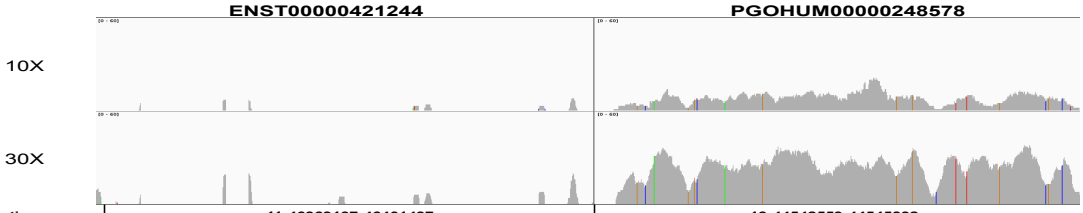
Sequence similarity poses the biggest challenge in RNASeq analysis for pseudogene expression quantification. To address the multiple alignment issue, one of the common approaches is to discard all multireads. Tonner *et al.* [18] developed a special method to quantify the expression levels of ribosomal protein pseudogenes. Their method only kept uniquely mapped read for abundance estimation. This idea is easy to apply, but tosses out information that is critical for quantification step. Consequently, it creates a bias toward genes or pseudogenes with a more unique sequence in the genome. Another approach is to assign the multireads based on the mapping quality. Nevertheless, there may be more than one alignment that shares the same quality score due to sequence similarity. Other approaches include assigning the multireads to the best locus based on local coverage estimation from uniquely mapped reads [13], or based on probabilistic models [10, 14]. However, none of these approaches leverages the relationship of read alignment among homologous regions.

Here, we present PseudoLasso, an analysis pipeline that keeps all multireads and correctly reassigns both unique reads and multireads between pseudogenes and their homologous regions. Our approach is based on an observation that reads from a given gene are linearly distributed to different genomic loci (Section 2). Since the homologous regions are sparse throughout the genome, this relation can be modeled by linear regression with L1 regularization (Section 3). Once the read count of each region is accurately estimated, reads are reassigned to regions in which the observed read count is lower than the estimation. Results show that PseudoLasso is able to accurately estimate the expression abundance and correctly assign reads among homologous loci (Section 4).

2. APPROACH

We adopt the terminology from Yale pseudogene knowledgebase (*Pseudogene.org*), and denote the functional genes where pseudogenes derived from as parent genes. Pseudogenes are represented by their pseudogene IDs (prefix PGOHUM), and other genes are represented by their Ensembl gene IDs [2].

We first examine the read alignments of TopHat2 between homologous regions, specifically between the regions of a pseudogene and its parent. A set of 100 parent transcripts



Location	11:46369137-46401497		13:44542559-44545332	
Experiment	10X	30X	10X	30X
Expected Count	109	348	0	0
Expected FPKM	2923	3105	0	0
Reported Count	9	15	153	159
Reported FPKM	160	305	3519	3673

Figure 1: FPKM and fragment counts reported by Cufflink after aligning reads from the *DGKZ* transcript (ENST00000421244) using TopHat2. Two different coverages, 10X and 30X, are used in the simulation. The peaks show the amount of reads (y-axis) covering across two genomic locations (x-axis), one for its pseudogene, and the other for itself. A sub-region of ENST00000421244 is displayed in this figure. PGOHUM00000248578 is not expressed at all, but is reported to have high FPKMs. On the other hand, ENST00000421244 is reported to have only 5% and 10% of the expected FPKMs.

that contain only one pseudogene are selected. The selection also ensure that there is no isoform in the set, and these 100 parent-pseudogene pairs do not overlap with others in their genomic locations. Paired-end reads are simulated for parent genes only, with different coverages and abundances (see method for data preparation). All the alignments, including multireads are used to estimate the number of fragments mapped to the parent genes themselves, and to their pseudogenes. Figure 2 shows an example of fragment counts between four pairs of regions, and these regions contain two processed pseudogenes (PGOHUM00000263241 - ENST00000580191, PGOHUM00000251182 - ENST00000512485) and two unprocessed pseudogenes (PGOHUM00000242685 - ENST00000527626, PGOHUM00000249020 - ENST00000388957). The observed number of fragments are plotted against the expected number of fragments from the parents. For a given gene, a linear relationship is observed in fragment count between two homologous regions, and the ratio is consistent across different coverages and abundances. However, the slope is different for each gene, implying that the ratio varies from gene to gene.

We further investigate the effect of read length on these 100 parents using 75bp and 100bp. Two of them are randomly chosen to demonstrate this in Figure 3. The slopes vary with different read length for both ENST00000580191 and ENST00000527626. Hence, with respect to the multiple alignment issue, the linearity characteristic holds with different read depths, but is sensitive to read lengths.

Based on these observations, mapped reads are linearly distributed among homologous regions. The total number of reads for a gene can be estimated from the reads aligned to itself and its homologous loci with different weights. We model this relationship through a linear regression model, and the weight coefficients are trained from simulated data for a specific read length. Next, we formalize the mathematical notations of this model. Table 1 summarizes the symbols we used in this article.

2.1 Model

Let y_i be the expected number of reads for a gene i among n genes $(0 < i \leq n)$; x_{ij} be the observed number of reads generated from gene i aligned to a locus j $(0 < j \leq m)$, where m is the total number of loci mapped by all reads in an experiment. Each locus j is associated with a weight β_i contributing to the prediction. Number of reads for gene i can be estimated by a weighted sum of observed counts across all loci. We use the "ˆ" symbol to represent estimated variables.

$$\hat{y}_i = x_{i1}\beta_1 + x_{i2}\beta_2 + ... + x_{im}\beta_m + \epsilon, \epsilon \sim N(0,1) \quad (1)$$

Putting all n genes and m loci in a matrix form $(n \leq m)$, we have

$$\hat{Y}_{n\times 1} = X_{n\times m}\beta_{m\times 1} + \epsilon_{n\times 1} \quad (2)$$

Y is a vector that contains the "ground truth" for n genes, and we define X as the distribution matrix to keep track of the aligned loci for each gene. We assume errors follow a normal distribution with mean of zero.

DEFINITION 1. **(Distribution Matrix)** Let R be a set of reads in the experiment, $R = \{r_1, r_2, \cdots r_p\}$. G be a set of genes, $G = \{g_1, g_2, \cdots, g_n\}$, and L be a set of genomic loci, $L = \{l_1, l_2, \cdots, l_m\}$. X is a $n \times m$ distribution matrix with n genes and m loci. We define $R_{g_i l_j} \subset R$ to denote a subset of reads, such that $r_k \in R_{g_i l_j}$ if and only if $\phi(r_k, g_i, l_j) = 1$. $\phi(r_k, g_i, l_j)$ is an indicating function as described in Equation (3). Each value in the distribution matrix represents the number of reads from gene g_i aligned to locus l_j, and thus $x_{ij} = |R_{g_i l_j}|$. We use the cardinality notation to represent the number of element in a set.

$$\phi(r_k, g_i, l_j) = \begin{cases} 1 & \text{if read } r_k \text{ comes from gene } g_i \\ & \text{and is aligned to locus } l_j \\ 0 & \text{otherwise} \end{cases} \quad (3)$$

A good estimation can be evaluated by minimizing the difference between the predicted value $\hat{y}_i$ and the expected value y_i, which leads to our objective function,

$$\underset{\beta}{\operatorname{argmin}} \|Y - X\beta\|^2 \quad (4)$$

For any given gene, there are only a few homologous loci across the genome. Therefore, fragments from a gene are

aligned to only a few regions, yielding a sparse distribution matrix. This sparsity can be enhanced by *L1* regularization, and used as a constraint to augment our objective function. Finally, these coefficients (β) can be estimated as described in Equation (5).

$$\hat{\beta} = \underset{\beta}{\operatorname{argmin}} \|Y - X\beta\|^2 + \lambda\|\beta\|_1 \quad (5)$$

2.2 Distribution Matrix Reconstruction

In real RNASeq data, we do not know the distribution matrix. Instead, we only know the total number of reads aligned to each genomic locus after the alignment step. In other words, we can only obtain a vector of column summation with respect to the distribution matrix from the real data. According to Equation (4), we need to reconstruct this distribution matrix first from the real data in order to estimate the read count for each gene.

Given only the summation of each column j, $\sum_{i=1}^{n} x_{ij}$, there are infinite solutions for x_{ij}. However, we observed earlier that the amount of reads at x_{ij} is proportional to the expected number of reads y_i for gene g_i across different replicates. Hence, this ratio is constant and can be inferred from simulated data. We rewrite x_{ij} as $z_{ij}c_i$, where c_i is a variable for gene i, and z_{ij} corresponds to the constant ratio related to c_i. We define a pseudo matrix for these ratios $\{z_{ij}\}$, which is a transformation of the distribution matrix.

DEFINITION 2. **(Pseudo Matrix)** The pseudo matrix $Z_{n\times m}$ is defined by the proportion of reads mapped to locus l_j out of the total number of reads for a given gene g_i. Let Y be a vector of expected number of reads for genes G, $Y = \{y_1, y_2, \cdots, y_n\}$. y_i is defined as $|R_{g_i}|$, such that $r_k \in R_{g_i}$ if and only if $\psi(r_k, g_i) = 1$. $\psi(r_k, g_i)$ is another indicating function described in Equation (6). Each value is computed by $z_{ij} = \frac{x_{ij}}{y_i}$.

$$\psi(r_k, g_i) = \begin{cases} 1 & \text{if read } r_k \text{ comes from gene } g_i \\ 0 & \text{otherwise} \end{cases} \quad (6)$$

After the substitution, we reduce the unknown variables down to $\{c_1, c_2, \cdots, c_n\}$. Since there are n variables and m equations, and $n \leq m$, the variable c_i can be estimated with the following objective function and constraint.

$$\arg \min_{\{c_1, c_2, \cdots, c_n\}} \|\sum_{i=1}^{n} x_{ij} - \sum_{i=1}^{n} \alpha_{ij} c_i\|^2, \text{s.t. } c_i \geq 0, \forall i \quad (7)$$

If the pseudo matrix is not full rank, it implies that the read distribution profiles are linearly dependent among a subset of genes. The corresponding c_i for these genes can have more than one solution. In reality, it happens to a group of genes that have extremely high sequence similarities, and as a result, their distribution profiles are indistinguishable from each other. We calculate the pairwise correlation for each gene based on the pseudo matrix. The highest correlation coefficient refers to the correlation of itself, so the second highest correlation coefficient is selected to reflect the likelihood of gene g_i being linear dependent to any others in the gene set G. Hence, a high correlation coefficient indicates a low confidence on the c_i we estimated from Equation (7).

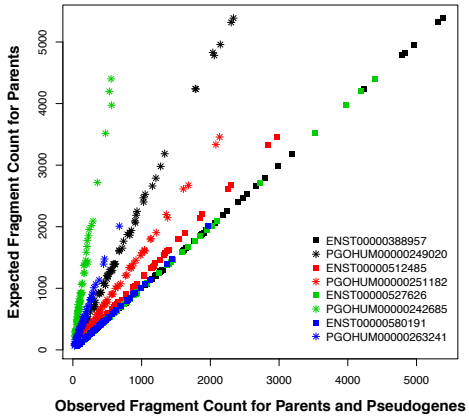


Figure 2: Comparing the number of fragments mapped to a functional parent gene and its pseudogene. Paired-end reads of 100bp are simulated for four functional genes (ENST-) from 60 sets of experiments. The expected number of reads from a parent gene (y-axis) is plotted against the number of reads mapped to either itself or its pseudogene (x-axis). The parent-pseudogene pair is indicated by the same color.

3. IMPLEMENTATION

We present PseudoLasso, which learns the behavior of reads distributed among homologous regions. Leveraging that knowledge, it estimates read abundance of each gene, and reassigns reads to the corresponding loci. The final read alignments are outputted in BAM format. The workflow is illustrated in Figure 4. The training stage uses simulated reads from a set of pseudogenes and their parents to learn the pseudo matrix and the lasso coefficients. In the validation stage, lasso coefficients and pseudo matrix are evaluated using a different set of simulated experiments. We describe our method below, starting with data preparation.

3.1 Data Preparation

RNA sequencing simulator from GeneScissors [22], RNAseqSim, is used to generate paired-end reads for a list of genes with fragment size ranging from 100bp to 400bp. Gene definitions are based on the annotation from Ensembl release 74 of Human Genome GRCh37. Each transcript is assigned with an abundance level, and the simulator uniformly samples fragments up to the given abundance. In order to mark the origin of each read, the software is modified to inherit the gene ID in read names. Ten different levels of read coverage are used to imitate low (5X, 7X, and 10X), medium (13X, 15X, 17X, and 20X) and deep (23X, 27X, and 30X) sequencing. Transcript abundance is assigned either with a fix number across all transcripts (4A, 6A, 8A)[1], or with three different sets of random numbers (from 3A to 20A) for each transcript. In total, the combination yields 60 sets of data. Six data sets are randomly chosen for validation, and the remaining sets serve as replicates during training.

[1]To distinguish from 'X' in coverage, the letter 'A' is used to denote the magnitude of abundance level. 4A, 6A, and 8A mean that transcripts are expressed 4,6,8 times respectively.

Table 1: Mathematical Notations

Symbol	Description
$G = \{g_1, g_2, \cdots, g_n\}$	a set of gene g_i , where $1 < i \leq n$
$L = \{l_1, l_2, \cdots, l_m\}$	a set of mapped loci l_j , where $1 < j \leq m$ and $n \leq m$
$Y = \{y_1, y_2, \cdots, y_n\}$	read counts or fragment counts y_i for gene g_i
$R = \{r_1, r_2, \cdots, r_p\}$	a set of reads in an experiment
$R_{g_i} \in R$	a subset of reads come from gene g_i
$R_{g_i l_j} \in R_{g_i}$	a subset of reads come from gene g_i and aligned to locus l_j
$X_{n \times m}$	an n by m distribution matrix
$Z_{n \times m}$	an n by m pseudo matrix that is a transformation of $X_{n \times m}$.
$\beta = \{\beta_1, \beta_2, \cdots, \beta_m\}$	a vector of weights indicate the contribution of loci l_j toward $\hat{Y}$
$C = \{c_1, c_2, ..., c_n\}$	a set of variables associated with gene i for distribution matrix reconstruction
$U = \{u_1, u_2, ..., u_n\}$	number of missing reads in locus l_j
$V = \{v_1, v_2, ..., v_m\}$	v_j represents a set of leftover uniquely aligned reads for locus l_j after Algorithm 1
$\psi(r_k, g_i)$	an indicating function for read r_k comes from gene g_i
$\phi(r_k, g_i, l_j)$	an indicating function for read r_k comes from gene g_i and aligned to locus l_j

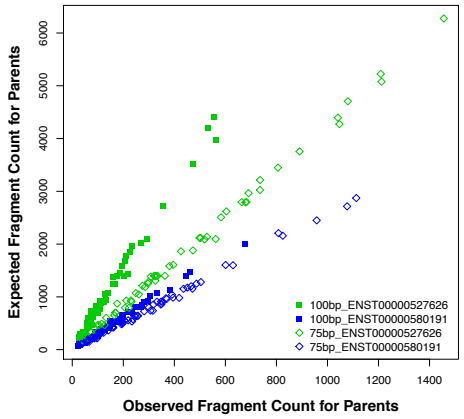


Figure 3: Read length effect on fragment counts. Paired-end reads are simulated from 60 sets of experiments, with read length of 75bp and 100bp for ENST00000580191 and ENST00000527626. The expected number of fragment (y-axis) is plotted against the observed number of fragment (x-axis). Read lengths are marked by different symbols, and genes are depicted by different colors.

3.2 Read Alignment

Reads are aligned to the reference genome using TopHat2. With the default settings, multiple alignments are reported up to 20 records, and these multireads are kept for analysis.

In simulated data, the name of each read is tagged with a gene ID of its origin, thus expected number of reads for a gene can be counted directly by the number of unique read names containing this tag. We use gene IDs instead of transcript IDs to simplify the distribution matrix. These counts provide the "ground truth" of read count for all genes. For each gene, we iterate through all alignment records to construct the distribution matrix. Each aligned locus is matched to a gene based on the span of genomic coordinates. We use Samtools [11] to retrieve the alignment information for mapped reads, and Bedtools to facilitate the gene matching.

3.3 Coefficient Training

We organize the distribution matrix X so that the first n columns represent the genomic locus of n genes. The diagonal values (x_{ii}) in the first half of the matrix (up to the n^{th} column) describe the read count of self-alignment. We assume all of the pseudogenes and their homologous parents are expressed, and the sparsity is enforced on the unexpressed loci, which correspond to $\{l_{n+1}, l_{n+2}, \cdots, l_m\}$.

As denoted in Equation (5), lasso regression is solved via coordinate descent, which is implemented in the *GLMNET* package by Friedman *et al.* [3]. The penalty strength parameter is adjusted to suppress the unexpressed loci.

3.4 Distribution Matrix Reconstruction

The distribution matrix is constructed by iterating through all ID-tagged reads in simulation, but can only be inferred through pseudo matrix in real data. Taking account of the variations among data sets, we use the average values of pseudo matrix from the training sets. Based on Equation (7), it is a non-negative least-squares constraints problem [9], and can be solved by a default function in Matlab, *lsqnonneg*.

3.5 Alignment Correction

After we obtain the estimated read count for each gene, we sort through all the read alignments and assign them to the location of these genes up to the estimated amount. The algorithm involves three stages. Uniquely mapped reads are retained in the first stage, and multireads are assigned to the most likely gene region in the second stage. Reconstructed distribution matrix is used to guide the realignment of the remaining reads in the third stage.

The first stage is summarized in Algorithm 1. The algorithm collects all the uniquely mapped reads that fall within the genomic locus l_j. These reads are sorted from good to bad based on the sequence quality (MAPQ) and whether the mate read is properly mapped for paired-end sequencing. This sorting procedure is described in Algorithm 2. Since the distribution matrix is ordered, each locus $\{l_1, l_2, \cdots, l_n\}$ is associated with a gene g_i, and the top $\hat{y}_i$ uniquely mapped reads are kept for locus l_i. The remaining loci $\{l_{n+1}, l_{n+2}, \cdots, l_m\}$ are assumed to be not expressed, with zero estimated read count, $\{\hat{y}_{n+1}, \hat{y}_{n+2}, \cdots, \hat{y}_m\} = 0$. For locus l_j, we use count u_j to keep track of the difference between estimated abundance and the number of kept reads. If the remaining count u_j reaches zero, then locus l_j is resolved. If the number of uniquely mapped reads exceeds

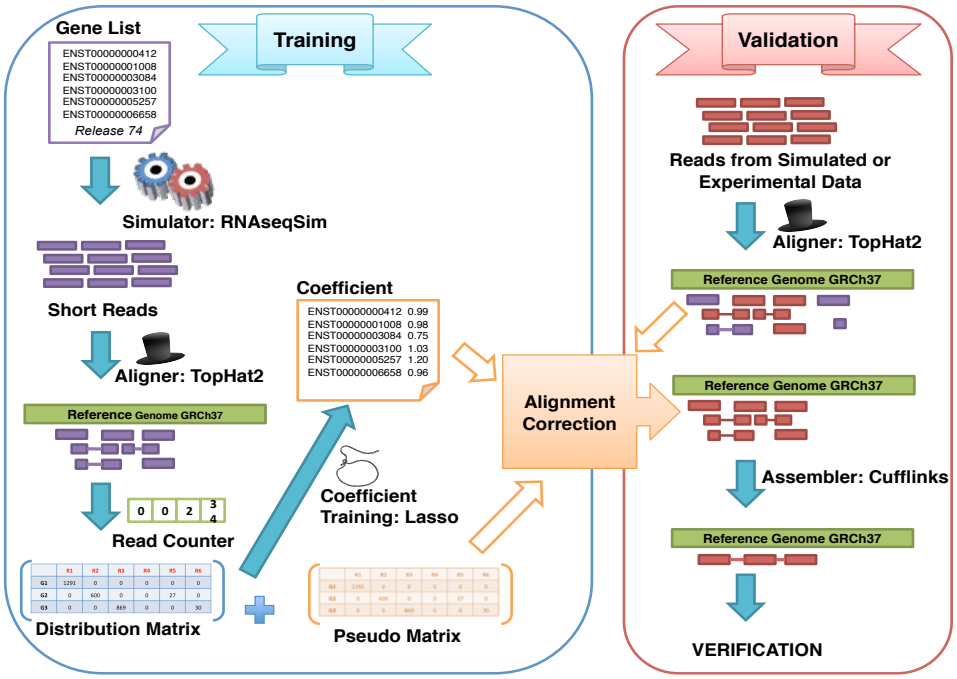


Figure 4: The workflow of PseudoLasso

the estimated abundance $\hat{y_j}$, the leftover reads v_j are written to a different file and are subjected to be reassigned to a different locus in the third stage.

In the second stage, Algorithm 3, the "unresolved loci" are sorted based on their remaining counts. Unassigned multireads are retrieved for these loci and sorted using the criteria mentioned above. We start the correction from a locus with the least amount of remaining counts, and assign the top u_j multireads to locus l_j. Once a multiread has been assigned, it is removed from the multireads pool. The remaining count is updated after each assignment, and if it reaches zero, the corresponding locus is resolved.

The remaining unresolved loci do not have enough aligned reads, and thus require realignment of leftover reads from the homologous loci. For gene g_i, homologous loci can be uncovered from the distribution matrix, where $x_{ij}(i \neq j)$ is greater than zero for locus l_j. We sort x_{ij} in descending order for each gene g_i, and retrieve the leftover reads from l_j. These reads are aligned back to the sequence spanned loci l_i using Blastn [1]. The top u_i alignments with e-value $\leq 1.00E-05$ are assigned to g_i and converted to BAM format. u_i is updated again after each assignment. The process is repeated for each homologous loci l_j until u_i reaches zero or there are no more leftover reads from homologous loci. This step is described in Algorithm 4.

The final list of alignments is acquired by removing the records from $DList$ and adding the newly assigned alignments from $SList$ to the original results reported by TopHat2. The corrected alignments are stored in BAM format, and can be processed for downstream analyses, such as transcriptome reconstruction and statistical analysis.

3.6 Validation

A separate set of replicates is used to validate our estimated read count for pseudogenes and parent genes. The prediction error is evaluated by the relative error with re-

Algorithm 1 Retain Uniquely Mapped Read

Input: R, L, $\hat{Y}$
Output: U, V
Let $\{\hat{y}_{n+1}, \hat{y}_{n+2}, \cdots, \hat{y}_m\} = 0$
for $l_j \in L$ **do**
  Let $u_j = \hat{y}_j$, $v_j = \emptyset$
  Let R_{l_j} = set of reads aligned to locus j
  R'_{l_j} = **QualitySort**(R_{l_j})
  for $r_k \in R'_{l_j}$ **do**
    if r_k is an uniquely mapped read **then**
      if $u_j > 0$ **then**
        $u_j = u_j - 1$
      else
        Add r_k to v_j
      end if
    end if
  end for
end for

Algorithm 2 QualitySort Procedure

Input: R_{l_j}
Output: Sorted R_{l_j} based on read quality
Let $Qscore = \{0, 0, \cdots, 0\}_{1 \times |R_{l_j}|}$
for $r_k \in R_{l_j}$ **do**
  if r_k is paired and mate is properly paired **then**
    $Qscore_k$ = MAPQ(r_k) + 100
  else
    $Qscore_k$ = MAPQ(r_k)
  end if
end for
for $r_k, r_{k+1} \in R_{l_j}$ **do**
  if $Qscore_{k+1} > Qscore_k$ **then**
    swap(r_{k+1}, r_k)
  end if
end for

Algorithm 3 Select Multiread

Input: R, L, U
Output: $DList$, $SList$, U

Let M = a list of multireads with $m_k = (readName, alignment)$
$U' = ascending\_sort(U)$
for $u_j \in U'$ **do**
  Let $M_{l_j} \in (M \cap R_{l_j})$
  $M'_{l_j} = \textbf{QualitySort}(M_{l_j})$
  while $u_j > 0$ and $M'_{l_j} \neq \emptyset$ **do**
    $r_k = M'_{l_j}(1)$
    $u_j = u_j - 1$
    Add $r_k.alignment$ to $SList$
    for $r_a \in M$ **do**
      if $r_a.readName == r_k.readName$ **then**
        Add $r_a.alignment$ to $DList$
        Remove r_a from M
        if $r_a \in M'_{l_j}$ **then**
          Remove r_a from M'_{l_j}
        end if
      end if
    end for
  end while
end for

Algorithm 4 Realign Reads from Homologous Loci

Input: V, X, U
Output: $DList$, $SList$

$U' = ascending\_sort(U)$
for $u_i \in U'$ **do**
  if $u_i > 0$ **then**
    $x'_{i.} = descending\_sort(x_{i.})$
    for $x'_{ij} \in x'_{i.}$ **do**
      if $x'_{ij} \neq 0$ and $v_j \neq \emptyset$ **then**
        if $u_i > 0$ **then**
          for $r_k \in v_j$ **do**
            $b = \textbf{Blastn}(r_k, sequence(l_i))$
            if $b.evalue < 1.00E-05$ **then**
              $u_i = u_i - 1$
              $bam = \textbf{BLAST2BAM}(b)$
              Add bam to $SList$
              Add r_k to $DList$
              Remove r_k from v_j
            end if
          end for
        end if
      end if
    end for
  end if
end for

Table 2: Prediction Error for 128 Gene Set

Data Set	Overall Error
Validation 1	0.0091
Validation 2	0.0093
Validation 3	0.0070
Validation 4	0.0037
Validation 5	0.0041
Validation 6	0.0038

spect to the true count. We compare our read count and FPKM to the traditional Tuxedo pipeline.

4. RESULTS

According to *Pseudogene.org* knowledgebase (Build 74), there are 16367 pseudogenes and 6251 parent genes located on the canonical chromosomes (Chr1-22, X, Y, and mitochondira). We removed the pseudogenes that overlap with any of the parents in their genomic locations. Our method used paired-end reads with a read length of 100bp, and fragment size from 100bp to 400bp. Thus, pseudogenes with read length less than 100 were discarded, resulting in 14246 candidate pseudogenes and 5979 parent genes.

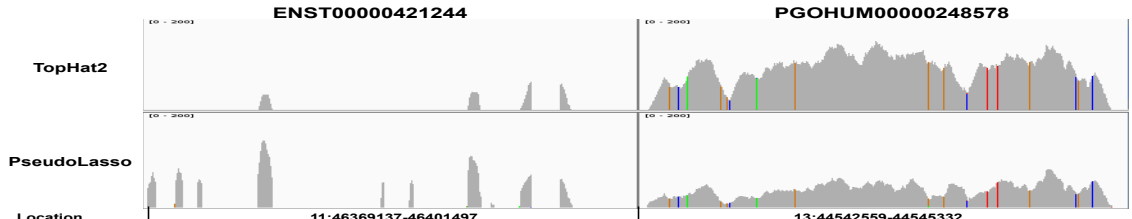
We first examined our approach using a small number of genes where a portion of the homologous pairs were expressed at the same time. The gene set was expanded upon the 100 parents we used earlier to examine the read alignment behavior between homologous regions (Section 2). Additional 8 more parent genes and 10 pairs of pseudogenes along with their homologous parents were randomly chosen from the candidate gene sets, resulting in a total of 128 genes expressed in our initial experiment.

In the second set of experiment, we assumed all of the pseudogenes were expressed at the same time as the parent genes. Since a parent gene could have more than one pseudogene, the 118 parents from the first experiment were associated with 206 pseudogenes, forming a set of 324 genes.

Applying our approach to a larger set of data, we randomly selected 709 genes from the candidate parent gene set. Assuming their pseudogenes were also expressed, 949 pseudogenes were added to the list, resulting a set of 1658 genes.

4.1 Results from 128 Gene Set

Read alignment behavior was modeled using 54 replicates, and validated by other 6 replicates. Relative errors were computed to evaluate the accuracy of our model. Table 2 summarized the average error rate of our predictions for these 128 genes. The low error rates showed that our model was able to recover the distribution matrix and provide an accurate estimation of the fragment count. Carrying on with the alignment correction, we used the *DGKZ* transcript to compare our approach with the Tuxedo pipeline. In Figure 5, reads were generated from the *DGKZ* transcript only, but TopHat2 aligned most of the reads to its pseudogene, PGOHUM00000248578. Our approach used the reconstructed distribution matrix to correct read alignments. The diminished amount of reads at the PGOHUM00000248578 locus along with the increased number of reads at the ENST00000421244 locus showed that our algorithm was able to successfully reassign reads to the correct region.



Location	11:46369137-46401497		13:44542559-44545332	
Experiment	TopHat2	PseudoLasso	TopHat2	PseudoLasso
Expected Count	1366	1366	0	0
Expected FPKM	624	624	0	0
Reported Count	118	297	1356	333
Reported FPKM	343	582	2407	698

Figure 5: Comparing the FPKM and fragment counts reported by Cufflink with TopHat2 and PseudoLasso. Reads are simulated from the *DGKZ* transcript (ENST00000421244) only. TopHat2 aligns most of the reads to it pseudogene, PGOHUM0000024857, whereas PseudoLasso is able to correctly reassign reads back to its origin.

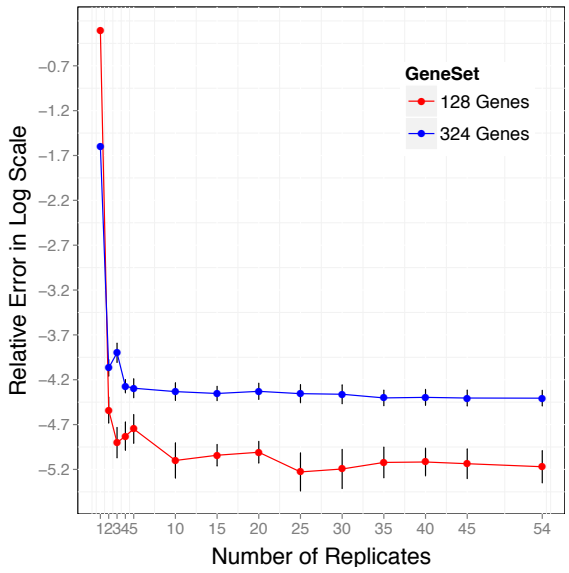


Figure 6: The average error rate of prediction is plotted against different number of replicates for two gene sets. The error rate (y-axis) is in logarithmic scale of base 10. As we increase the number of replicates (x-axis), we see a sharp decrease at 2 replicates, and a stable pattern after 5 replicates. Standard deviations are shown from 6 validations.

4.2 Replicates Determination

In order to scale up our approach to model more homologous genes, we used two small gene sets (128 and 324 gene sets) to determine the number of replicates that was sufficient for training. Six replicates with deep sequencing (30X) were used for validation, and the remaining 54 replicates were randomly selected for training. Figure 6 showed that the average error rate dropped dramatically after 2 replicates, and reached a plateau at 5 replicates. The standard deviations among the 6 validation sets are displayed.

4.3 Results from 1658 Gene Set

Table 3: Prediction Error for 1658 Gene Set

Data Set	Overall Error	Dependency $< 98\%$
Validation 1	1.05E-02	7.80E-03
Validation 2	9.94E-03	7.21E-03
Validation 3	1.04E-02	7.74E-03
Validation 4	1.10E-02	5.68E-03
Validation 5	8.89E-03	7.24E-03
Validation 6	1.23E-02	8.68E-03

We used 5 replicates to model the alignment behavior for the 1658 gene set. Five replicates were randomly selected from the data pool for training, and additional 6 replicates were selected for validation. We compared the estimated fragment count with the expected number of fragments for these 1658 genes. Table 3 showed that the overall errors were below 1.3% for all validations. Among these 1658 genes, there were 11 genes with a likelihood of dependency greater than 98%, which reflects a low confidence on their coefficients estimated by the model. After filtering out these 11 genes, the relative errors dropped below 1% as indicated in Table 3.

We used a subset of genes to demonstrate our fragment count estimation in detail in Tables 4 and 5. Genes with a low likelihood of dependency were accurately predicted across all validation data sets, with errors smaller than $1.34E-03$. On the other hand, the predictions for genes with a high likelihood of dependency ($> 98\%$) were inconsistent among data sets, and tended to present a high error rate ($> 30\%$).

We further compared the errors of fragment count at each locus between TopHat2 and our approach after read assignment. Table 6 showed that our approach performed better than TopHat2 in terms of aligning reads to the correct loci. After removing the genes with a high likelihood of dependency, the error rate was reduced by 10-fold with our approach compared to TopHat2. As demonstrated in Table 7, the fragment count reported by TopHat2 were higher than expected for pseudogene and lower than the true count for parent genes. It was prone to aligning reads from parents to their homologous pseudogenes. On the contrary, our approach was able to correctly align reads among homologous loci.

Table 4: Model Validation for Genes with Low Dependency

Gene Name	ENSG00000023287			ENSG00000034713			ENSG00000064309		
Dependency[2]	0.0013			0.0013			0.0013		
	Truth[3]	Pred[4]	Err[5]	Truth	Pred	Err	Truth	Pred	Err
Validation 1	3268.00	3268.05	1.51E-05	510.00	510.05	1.04E-04	4978.00	4978.04	8.11E-06
Validation 2	4872 .00	4872.07	1.51E-05	766.00	766.08	1.04E-04	7495 .00	7495.06	8.11E-06
Validation 3	6120.00	6120.09	1.51E-05	977.00	977.10	1.04E-04	9935.00	9935.08	8.11E-06
Validation 4	12536.00	12536.19	1.51E-05	2097.00	2097.22	1.04E-04	13088.00	13088.11	8.11E-06
Validation 5	11699.00	11699.17	1.51E-05	368.00	368.04	1.04E-04	16817.00	16817.14	8.11E-06
Validation 6	14960.00	14960.23	1.51E-05	617.00	617.06	1.34E-03	14801.00	14801.12	8.11E-06

Table 5: Model Validation for Genes with High Dependency

Gene Name	ENSG00000176269			PGOHUM00000241986			PGOHUM00000241985			PGOHUM00000263400		
Dependency	0.9884			0.9998			0.9998			0.9819		
	Truth	Pred	Err	Truth	Pred	Err	Truth	Pred	Err	Truth	Pred	Err
Validation 1	459	400.60	0.13	123.00	0	1.00	122.00	245.17	1.01	159.00	325.32	1.05
Validation 2	705	817.00	0.16	172.00	0	1.00	182.00	351.11	0.93	241.00	461.56	0.92
Validation 3	856	1009.31	0.18	199.00	405.61	1.04	205.00	0	1.00	319.00	619.80	0.94
Validation 4	1823	1844.00	0.01	276.00	0	1.00	388.00	665.90	0.72	382.00	932.18	1.44
Validation 5	2211	2429.68	0.10	575.00	619.72	0.08	92.00	48.46	0.47	503.00	945.47	0.88
Validation 6	1211	1298.88	0.07	567.00	0	1.00	169.00	739.55	3.38	482.00	593.30	0.23

5. DISCUSSION AND CONCLUSION

Pseudogenes quantification remains challenging due to the high sequence similarity with their parent genes. Coupling with RNASeq technology for expression profiling, a short read sequence with low complexity may align to multiple places. Current approaches only focus on resolving the multiple mapping problem, either neglecting the multireads or assigning them to the best place based on the local coverage. Nevertheless, none of them considered the alignment relationship between homologous loci nor attempted to correct the falsely aligned reads.

In this article, we present PseudoLasso, a genome-wide approach to correct read alignment among homologous loci, specifically for pseudogenes and their parents. Given a list of gene of interests, PseudoLasso simulates reads for these genes, and constructs a model that describes the mapping behavior (pseudo matrix) of an aligner, e.g.TopHat2, and a vector of weight for each loci contributing to the abundance estimation (β coefficients). Applying the trained model to real data, PseudoLasso reconstructs the read distribution matrix and estimates the gene expression abundance, which further serves as a guideline to select multireads and realign excess reads from homologous loci.

In the effort to correctly assign reads for pseudogenes and their homologous parents, we trained a model on a subset of pseudogenes and their parents. We used 100bp paired-end reads to learn the alignment behavior of TopHat2. The results showed that this approach was able to estimate the abundance with high accuracy, and assign reads to the best locus to meet the estimated abundance.

The alignment behavior is sensitive to read length, and thus a new training is required for a different read length. Currently, many well-studied publicly available RNASeq data are generated with read length of 50bp and 75bp. Therefore, we plan on training a new model using 75bp short reads, and evaluate the pseudogene expression on the Human Body Map 2.0 Project (NCBI GEO accession GSE30611). Our approach can be extended to model all homologous regions across the entire genome, not just limited to the pseudogenes and their parents. In the future, to accommodate for the large amounts of data, we plan on storing the sparse matrices in a compressed format to facilitate computation.

[2]Likelihood of dependency
[3]The expected number of fragment count
[4]Estimated fragment count
[5]Relative error

6. ACKNOWLEDGMENTS

We thank Zhaojun Zhang for his comments and discussion, and Shunping Huang for his assistance in setting up the RNA simulator software, RNAseqSim. This work was funded by NIH R01HG006703, NIH P50 GM076468-08 and NSF IIS-1313606.

7. REFERENCES

[1] S. F. Altschul, W. Gish, W. Miller, E. W. Myers, and D. J. Lipman. Basic local alignment search tool. *Journal of molecular biology*, 215(3):403–10, Oct. 1990.

[2] P. Flicek, M. R. Amode, D. Barrell, K. Beal, K. Billis, S. Brent, D. Carvalho-Silva, P. Clapham, G. Coates, S. Fitzgerald, L. Gil, C. G. Girón, L. Gordon, T. Hourlier, S. Hunt, N. Johnson, T. Juettemann, A. K. Kähäri, S. Keenan, E. Kulesha, F. J. Martin, T. Maurel, W. M. McLaren, D. N. Murphy, R. Nag, B. Overduin, M. Pignatelli, B. Pritchard, E. Pritchard, H. S. Riat, M. Ruffier, D. Sheppard, K. Taylor, A. Thormann, S. J. Trevanion, A. Vullo, S. P. Wilder, M. Wilson, A. Zadissa, B. L. Aken, E. Birney, F. Cunningham, J. Harrow, J. Herrero, T. J. P. Hubbard, R. Kinsella, M. Muffato, A. Parker, G. Spudich, A. Yates, D. R. Zerbino, and S. M. J. Searle. Ensembl 2014. *Nucleic acids research*, 42(Database issue):D749–55, Jan. 2014.

[3] J. Friedman, T. Hastie, and R. Tibshirani. Regularization Paths for Generalized Linear Models via Coordinate Descent. *Journal of statistical software*, 33(1):1–22, Jan. 2010.

[4] M. Garber, M. G. Grabherr, M. Guttman, and C. Trapnell. Computational methods for transcriptome annotation and quantification using RNA-seq. *Nature methods*, 8(6):469–77, June 2011.

Table 6: Comparison of Error Rate Between TopHat2 and PseudoLasso

Data Set	TopHat2		PseudoLasso	
	Overall Error	**Dependency** $< 98\%$	**Overall Error**	**Dependency** $< 98\%$
Validation 1	6.61E-02	5.87E-02	1.02E-02	7.50E-03
Validation 2	6.48E-02	5.75E-02	9.75E-03	7.01E-03
Validation 3	6.61E-02	5.87E-02	1.03E-02	7.57E-03
Validation 4	6.61E-02	5.87E-02	1.09E-02	8.24E-03
Validation 5	6.61E-02	5.87E-02	8.69E-03	7.04E-03
Validation 6	6.61E-02	5.87E-02	1.22E-02	8.56E-03

Table 7: Comparison of Fragment Count Between TopHat2 and PseudoLasso

Pseudogene				Parent Gene			
Gene Name	**Truth**	**TopHat2**	**PseudoLasso**	**Gene Name**	**Truth**	**TopHat2**	**PseudoLasso**
PGOHUM00000250236	1339	5029	1325	**ENSG00000133030**	6429	5528	6430
PGOHUM00000248345	1532	5387	1601	**ENSG00000001630**	11041	9605	10882
PGOHUM00000237670	3200	5711	3228	**ENSG00000001630**	11041	9605	10882
PGOHUM00000244974	3929	4678	3955	**ENSG00000176853**	5326	4997	5311
PGOHUM00000244647	1155	1826	1119	**ENSG00000181163**	4098	2802	4266
PGOHUM00000238769	328	592	335	**ENSG00000106615**	756	623	740

[5] Y. J. Han, S. F. Ma, G. Yourek, Y.-D. Park, and J. G. N. Garcia. A transcribed pseudogene of MYLK promotes cell proliferation. *FASEB journal : official publication of the Federation of American Societies for Experimental Biology*, 25(7):2305–12, July 2011.

[6] P. G. Hawkins and K. V. Morris. Transcriptional regulation of Oct4 by a long non-coding RNA antisense to Oct4-pseudogene 5. *Transcription*, 1(3):165–175, Jan. 2010.

[7] J. E. Karro, Y. Yan, D. Zheng, Z. Zhang, N. Carriero, P. Cayting, P. Harrrison, and M. Gerstein. Pseudogene.org: a comprehensive database and comparison platform for pseudogene annotation. *Nucleic acids research*, 35(Database issue):D55–D60, Jan. 2007.

[8] D. Kim, G. Pertea, C. Trapnell, H. Pimentel, R. Kelley, and S. L. Salzberg. TopHat2: accurate alignment of transcriptomes in the presence of insertions, deletions and gene fusions. *Genome biology*, 14(4):R36, Apr. 2013.

[9] C. L. Lawson and R. J. Hanson. *Solving Least Squares Problems*. Society for Industrial and Applied Mathematics, 1995.

[10] B. Li and C. N. Dewey. RSEM: accurate transcript quantification from RNA-Seq data with or without a reference genome. *BMC bioinformatics*, 12:323, Jan. 2011.

[11] H. Li, B. Handsaker, A. Wysoker, T. Fennell, J. Ruan, N. Homer, G. Marth, G. Abecasis, and R. Durbin. The Sequence Alignment/Map format and SAMtools. *Bioinformatics (Oxford, England)*, 25(16):2078–9, Aug. 2009.

[12] H. Lin, A. Shabbir, M. Molnar, and T. Lee. Stem cell regulatory function mediated by expression of a novel mouse Oct4 pseudogene. *Biochemical and biophysical research communications*, 355(1):111–6, Mar. 2007.

[13] A. Mortazavi, B. A. Williams, K. Mccue, L. Schaeffer, and B. Wold. Mapping and quantifying mammalian transcriptomes by RNA-Seq. 5(7):1–8, 2008.

[14] B. PaÅ§aniuc, N. Zaitlen, and E. Halperin. Accurate estimation of expression levels of homologous genes in RNA-seq experiments. *Journal of computational biology : a journal of computational molecular cell biology*, 18(3):459–68, Mar. 2011.

[15] B. Pei, C. Sisu, A. Frankish, C. Howald, L. Habegger, X. Mu, R. Harte, S. Balasubramanian, A. Tanzer, M. Diekhans, A. Reymond, T. J. Hubbard, J. Harrow, and M. B. Gerstein. The GENCODE pseudogene resource. *Genome Biology*, 13(9):R51, Jan. 2012.

[16] R. C. Pink, K. Wicks, D. P. Caley, E. K. Punch, L. Jacobs, and D. R. F. Carter. Pseudogenes: pseudo-functional or key regulators in health and disease? *RNA (New York, N.Y.)*, 17(5):792–8, May 2011.

[17] Z. Redshaw and A. J. Strain. Human haematopoietic stem cells express Oct4 pseudogenes and lack the ability to initiate Oct4 promoter-driven gene expression. *Journal of negative results in biomedicine*, 9(1):2, Jan. 2010.

[18] P. Tonner, V. Srinivasasainagendra, S. Zhang, and D. Zhi. Detecting transcription of ribosomal protein pseudogenes in diverse human tissues from RNA-seq data. *BMC Genomics*, 13(1):412, Jan. 2012.

[19] C. Trapnell, A. Roberts, L. Goff, G. Pertea, D. Kim, D. R. Kelley, H. Pimentel, S. L. Salzberg, J. L. Rinn, and L. Pachter. Differential gene and transcript expression analysis of RNA-seq experiments with TopHat and Cufflinks. *Nature protocols*, 7(3):562–78, Mar. 2012.

[20] C. Trapnell, B. A. Williams, G. Pertea, A. Mortazavi, G. Kwan, M. J. van Baren, S. L. Salzberg, B. J. Wold, and L. Pachter. Transcript assembly and quantification by RNA-Seq reveals unannotated transcripts and isoform switching during cell differentiation. *Nature biotechnology*, 28(5):511–5, May 2010.

[21] Y. Tutar. Pseudogenes. *Comparative and functional genomics*, 2012:424526, Jan. 2012.

[22] Z. Zhang, S. Huang, J. Wang, X. Zhang, F. Pardo Manuel de Villena, L. McMillan, and W. Wang. GeneScissors: a comprehensive approach to detecting and correcting spurious transcriptome inference owing to RNA-seq reads misalignment. *Bioinformatics (Oxford, England)*, 29(13):i291–9, July 2013.

[23] D. Zheng, A. Frankish, R. Baertsch, P. Kapranov, A. Reymond, S. W. Choo, Y. Lu, F. Denoeud, S. E. Antonarakis, M. Snyder, Y. Ruan, C.-L. Wei, T. R. Gingeras, R. Guigó, J. Harrow, and M. B. Gerstein. Pseudogenes in the ENCODE regions: consensus annotation, analysis of transcription, and evolution. *Genome research*, 17(6):839–51, June 2007.

[24] M. Zou, E. Y. Baitei, A. S. Alzahrani, F. Al-mohanna, N. R. Farid, and B. Meyer. Oncogenic Activation of MAP Kinase by BRAF Pseudogene in Thyroid Tumors 1. 11(1):57–65, 2009.