CAMDA2009

Critical Assessment of Massive Data Anaysis 2009

Oct. 5-6, 2009
Chicago, IL, USA
http://www.camda2009.org

Biomedical Informatics Center, Northwestern University

Sponsers:

From Microarray to Massive -- Ten years of CAMDA

In year 2000, we initiated the CAMDA conference because of the explosion of microarray data (Nature 411: 885, 2001). Now, we have seen an even bigger challenge with the rapid adoption of Next Generation Sequencing (NGS) in biomedical research. Accordingly, a couple of thought leaders including Joaquin Dopazo, David Kreil, and Simon Lin in the 2008 conference suggested changing the CAMDA focus from 'microarray data' to 'massive data'. As such, CAMDA resulted in a brand-new name of "**Critical Assessment of Massive Data Analysis**" in 2009.

For the first time in the history of biology, the massive amount of data is quickly outpacing Moore's Law.

> "The cost of analyzing the large data sets already exceeds the cost of generating them." – Editorial, Nature Methods, 6:623, 2009

The goal of CAMDA is to solve this grand challenge by crowdsourcing. We rely on the collective intelligence of the community to find viable solutions. To make it fun, we have twisted this approach with a competition. Ever year, a common data set is released to the contestants and a best presentation at CAMDA is awarded. We also compile the solutions from each year into a monograph for the distribution of knowledge. Presentations from this year will appear in the open journal of PLOS One as a special issue after peer review.

We hope you all enjoy this exciting meeting and the wonderful autumn in Chicago!

Warren A. Kibbe, PhD
Conference Chair
Northwestern University

Xijin Ge, PhD
CAMDA Planning Committee
South Dakota State University

Simon M. Lin
Conference Co-Chair
Northwestern University

Joaquin Dopazo
CAMDA Planning Committee
CIPF, Valencia, Spain

David Philip Kreil
Conference Co-Chair
Boku University Vienna, Austria

Pan Du
CAMDA Planning Committee
Northwestern University

CAMDA2009 Committee

Conference Chairs: Warren A. Kibbe, Simon Lin and David Kreil

Publication Committee Chair: Pan Du

Steering Committee

Peter Kopp
Center for Genetic Medicine, Feinberg School of Medicine
Northwestern University

Nadereh Jafari
Center for Genetic Medicine, Feinberg School of Medicine
Northwestern University

Scientific Committee

Warren Kibbe
Biomedical Informatics Center,
NUCATS
Northwestern University

Simon Lin
Biomedical Informatics Center,
NUCATS
Northwestern University

David Philip Kreil
Chair of Bioinformatics
Boku University Vienna
Austria

Joaquin Dopazo
Bioinformatics Department
Centro de Investigación Principe Felipe (CIPF)
Valencia, Spain

Pan Du
Biomedical Informatics Center,
NUCATS
Northwestern University

Xijin Ge
Department of Mathematics and Statistics
South Dakota State University

Ruchir Shah
Virtual Center for Computational Biology
SRA International Inc

Denise Scholtens
Department of Preventive Medicine,
Feinberg School of Medicine
Northwestern University

Julie Zhu
Program in Gene Function and Expression; Molecular Medicine
University of Massachusetts Medical School

Jake Chen
Indiana University School of Informatics
Purdue University Dept. of Computer & Information Science

Weida Tong
FDA

Hui Lu
Bioengineering Department
University of Illinois at Chicago

Tim Beissbarth
Bioinformatics, DKFZ,
Heidelberg, Germany

Daniel Berrar
Systems Biology Research Group
School of Biomedical Sciences,
University of Ulster
Northern Ireland

Philippe Broet
University Paris - XI, Paris, France

Ana Conesa
Bioinformatics Department
Centro de Investigación Principe Felipe (CIPF)
Valencia, Spain

Susmita Datta
Department of Bioinformatics and Biostatistics
University of Louisville
Louisville KY 40202, USA

Sandrine Dudoit
Department of Statistics
University of California, Berkeley
Berkeley, USA

Jelle Goeman
Department of Medical Statistics
Leiden University Medical Center,
Leiden
The Netherlands

Seon-Young Kim
Functional Genomics Research Center
Gwahangno, Yuseong-gu, Daejeon
Korea

David Montaner
Bioinformatics Department
Centro de Investigación Principe Felipe (CIPF)
Valencia, Spain

Yves Moreau
K.U. Leuven
ESAT-SCD (Bioinformatics)
Leuven, Belgium

Wei Pan
Division of Biostatistics
School of Public Health, University of Minnesota
Minneapolis, USA

Local Committee:

Warren A. Kibbe, Simon Lin, Pan Du, Gilbert Feng, Joshua Lamb, Dong Fu, Brian Chamberlain

Logistics Committee:

Joshua Lamb (Chair), Tyler Smith

Sponsorship Committee

Damir Herman, University of Arkansas

CAMDA2009 Agenda

Monday, Oct. 5, 2009		
Time	**Speaker**	**Title**
8:30-9:00	*Registration and breakfast (sponsored by Isilon Systems)*	
9:00-9:10	Warren Kibbe	Welcome: Ten Years of CAMDA
9:10-10:00	Martin Morgan (Keynote)	Computational Challenges in the Analysis of Short Read DNA Sequences
10:00-10:50	Mark B Gerstein (Keynote)	Human Genome Annotation
10:50-11:10	*Coffe Break (Sponsored by Isilon Systems)*	
Morning Session (Session Chair: Chun-Yu Liu)		
11:10-11:50	Jean-Francois Pessiot	PeakRegressor identifies composite sequence motifs responsible for STAT1 binding sites and their potential rSNPs
11:50-12:30	Sunduz Keles	A Statistical Framework for the Analysis of ChIP-Seq Data
12:30-14:00	*Lunch / Sponsered talk (13:00 – 14:00) by Isilon Systems*	
Afternoon Session (Session Chair: Denise Scholtens)		
14:00-14:40	Kun Huang	Comparative Analysis of ChIP-seq Data using Mixture Model
14:40-15:20	Pingzhao Hu	Scoring of ChIP-seq experiments by modeling large-scale correlated tests
15:20-15:40	*Coffe Break*	
15:40-16:20	Yanen Li	SeqMapReduce: software and web service for accelerating short sequence mapping
16:20-17:00	Yunlong Liu	STAT1 regulates microRNA transcription in interferon γ – stimulated HeLa cells
17:10-18:00	*Tutorial: 'Analyzing high-throughput short sequences with Bioconductor'*, by **Martin Morgan**	

Tuesday, Oct. 6, 2009		
Time	**Speaker**	**Title**
8:30-9:00	*Breakfast*	
Morning Session (Session Chair: Julie Zhu)		
9:00-9:40	Robert Grossman	Cloud-based Services for Large Scale Analysis of Sequence and Expression Data: Lessons Learned from Cistrack
9:40-10:20	Colin N. Dewey	Transcriptome analysis methods for RNA-Seq data
10:20-10:40	*Coffe Break*	
10:40-11:20	Susmita Datta	Next Generation Sequencing: Statistical Challenges and Opportunities
11:20-12:10	Bento Soares (Invited lecture)	Large-scale sequencing-based epigenomic analysis of Alu repeats in normal and in tumor cells
12:10-12:20	CAMDA2009 award annoucement	
12:20-14:00	*Lunch / Sponsered talk (13:00 – 14:00) by XtremeData*	
Afternoon Session (Session Chair: Gang Feng)		
14:00-14:40	Peter Larsen	Using Next Generation Sequencing Data for Structural Annotation of L. bicolor Mycorrhizal Transcriptome
14:40-15:20	Kun Huang	Parallel Computing Strategies for Sequence Mapping of NGS Data
15:20-15:40	*Coffe Break*	
15:40-16:20	Doug Cork, Steven Lembark*	Comparative Analysis of Pol II and HIV-1 Sequences Using the W-Curve
16:20-17:00	Fahad Saeed	How to Multiple Align Huge Number of Short Reads
17:00	Close of conference	

Conference Location:

Prentice 3rd Floor, Conference Room L
250 E. Superior Street
Chicago, IL 60611

Contents of Presentations

Human Genome Annotation

Keynote Speaker: Mark Gerstein, Yale University

Abstract

A central problem for 21st century science will be the annotation and understanding of the human genome. My talk will be concerned with topics within this area, in particular annotating pseudogenes (protein fossils), binding sites, CNVs, and novel transcribed regions in the genome. Much of this work has been carried out in the framework of the ENCODE and modENCODE projects.

In particular, I will discuss how we identify regulatory regions and novel, non-genic transcribed regions in the genome based on processing of tiling array and next-generation sequencing experiments. I will further discuss how we cluster together groups of binding sites and novel transcribed regions.

Next, I will discuss a comprehensive pseudogene identification pipeline and storage database we have built. This has enabled us to identify >10K pseudogenes in the human and mouse genomes and analyze their distribution with respect to age, protein family, and chromosomal location. I will try to inter-relate our studies on pseudogenes with those on transcribed regions. At the end I will bring these together, trying to assess the transcriptional activity of pseudogenes.

Throughout I will try to introduce some of the computational algorithms and approaches that are required for genome annotation -- e.g., the construction of annotation pipelines, developing algorithms for optimal tiling, and refining approaches for scoring microarrays.

References

http://bioinfo.mbb.yale.edu
http://pseudogene.org
http://tiling.gersteinlab.org

1. S Balasubramanian, D Zheng, YJ Liu, G Fang, A Frankish, N Carriero, R Robilotto, P Cayting, M Gerstein (2009), “Comparative analysis of processed ribosomal protein pseudogenes in four mammalian genomes.” Genome Biol 10: R2.

2. D Zheng, A Frankish, R Baertsch, P Kapranov, A Reymond, SW Choo, Y Lu, F Denoeud, SE Antonarakis, M Snyder, Y Ruan, CL Wei, TR Gingeras, R Guigo, J Harrow, MB Gerstein (2007), “Pseudogenes in the ENCODE regions: consensus annotation, analysis of transcription, and evolution.”, Genome Res 17: 839-51.

3. ZD Zhang, A Paccanaro, Y Fu, S Weissman, Z Weng, J Chang, M Snyder, MB Gerstein (2007), "Statistical analysis of the genomic distribution and correlation of regulatory elements in the ENCODE regions." Genome Res 17: 787-97.

4. MB Gerstein, C Bruce, JS Rozowsky, D Zheng, J Du, JO Korbel, O Emanuelsson, ZD Zhang, S Weissman, M Snyder (2007), "What is a gene, post-ENCODE? History and updated definition." Genome Res 17: 669-81.

5. J Rozowsky, G Euskirchen, RK Auerbach, ZD Zhang, T Gibson, R Bjornson, N Carriero, M Snyder, MB Gerstein (2009), "PeakSeq enables systematic scoring of ChIP-seq experiments relative to controls.". Nat Biotechnol

6. LY Wang, A Abyzov, JO Korbel, M Snyder, M Gerstein (2009), "MSB: A mean-shift-based approach for the analysis of structural variation in the genome." Genome Res 19: 106-17.

7. HY Lam, E Khurana, G Fang, P Cayting, N Carriero, KH Cheung, MB Gerstein (2009), "Pseudofam: the pseudogene families database.", Nucleic Acids Res 37: D738-43.

8. PM Kim, HY Lam, AE Urban, JO Korbel, J Affourtit, F Grubert, X Chen, S Weissman, M Snyder, MB Gerstein (2008), "Analysis of copy number variants and segmental duplications in the human genome: Evidence for a change in the process of formation in recent evolutionary history." Genome Res 18: 1865-74.

Computational Challenges in the Analysis of Short Read DNA Sequences

Keynote Speaker: Martin Morgan, Fred Hutchinson Cancer Research Center

Abstract

Short read DNA sequence data poses significant challenges for computational analysis. Here we survey and assess these challenges, providing creative solutions and possible directions for development. It is useful to distinguish between large-scale public data such as the TCGA, 1000 genomes and ENCODE projects, and data generated with more modest resources. The size of primary data is a major computational hurdle. However, many analyses are most interesting after data has been reduced (e.g., by alignment to reference sequences) to manageable size. The computational challenges then involve formulation and design of appropriate statistical questions, domain-specific (e.g., ChIP-seq) analyses, integrative approaches that combine sequence and other data sources, and sequence-based annotation. These themes are illustrated with reference to several examples from our group.

PeakRegressor identifies composite sequence motifs responsible for STAT1 binding sites and their potential rSNPs

Jean-François Pessiot[1], Hirokazu Chiba[1], Hiroto Hyakkoku[2,1], Takeaki Taniguchi[3], Wataru Fujibuchi[1*]

[1]Computational Biology Research Center, Advanced Industrial Science and Technology (AIST),

[2]Waseda University, [3]Mitsubishi Research Institute, Inc.

Abstract

How to identify true transcription factor binding sites on the basis of sequence motif information (e.g., motif pattern, location, combination, etc.) is an important question in bioinformatics. We present "PeakRegressor", a system that identifies binding motifs by combining DNA-sequence data and ChIP-Seq data. PeakRegressor uses L1-norm log linear regression in order to predict peak values from binding motif candidates. Our approach successfully predicts the peak values of STAT1 and Pol II with correlation coefficients as high as 0.65 and 0.66, respectively. Using PeakRegressor, we are able to identify composite motifs for STAT1, as well as potential regulatory SNPs (rSNPs) involved in the regulation of transcription levels of neighboring genes.

1 Introduction

The experimental identification of *cis*-regulatory sites based on transcription factor binding motifs (TFBMs) is a difficult and time-consuming task. In this regard, *in silico* analysis of TFBMs has recently attracted attention as a promising tool for discovering true *cis*-regulatory sites. Previous works attempt to find TFBMs to model the mechanisms underlying the control of gene expression levels[2, 4]. They assume that the gene expression levels are determined by the presence of certain motifs in the upstream regions of the genes. Based on this assumption, they find TFBM candidates which show a strong correlation with changes in the gene expression levels.[5] Instead of modeling the expression levels, another solution is to model the binding affinities between a protein and its target genes based on the thermodynamics theory. However, the binding affinities are difficult to measure and related works use transcription factor occupancy to

*Corresponding Author - w.fujibuchi@aist.go.jp

approximate binding affinity[6, 7].

In this article, we present PeakRegressor, a new tool for the identification of functional TFBMs from ChIP-Seq data. As far as we know, this is the first attempt to perform peak signal regression based on candidate motif models. Our contribution is twofold. First, in contrast with previous approaches, we use the peak scores (provided by[9]) as a surrogate for the binding affinities. We argue that they provide more accurate approximations and therefore lead to better identification of functional TFBMs. Second, our approach identifies not only primary TFBM candidates but also secondary motifs that may often synergistically strengthen or weaken the binding. The rest of this paper is organized as follows. We describe PeakRegressor in section 2. In section 3, we illustrate the performance of our approach on two ChIP-Seq datasets and discuss its ability to identify the binding motifs of STAT1 and Pol II.

2 PeakRegressor System to Find Functional TFBMs

PeakRegressor is a system to find TFBMs that are statistically important for transcription factor binding signals, by taking ChIP-Seq data as input, and outputs a list of TFBM candidates. The workflow is summarized in Figure 1.

Step 1 First, we define the peak sequences as the 200-bp genomic regions centered around the peaks. Then, we sort the peak sequences according to their ascending scores. We group the peak sequences into clusters such that each cluster contains 200 peaks of consecutive scores. Then, we apply MEME[1] to each peak sequence cluster. For each sequence cluster, MEME is parameterized in ZOOPS mode to find 10 motifs of lengths $8 - -20$.

This strategy has two advantages. First, it allows us to identify motifs that may be associated with a given binding affinity level. If a cluster contains only low (resp. high) binding affinity peaks, the corresponding sequences may contain weak (resp. strong) binding motifs, i.e., motifs that are specific to low (resp. high) binding affinity. Second, it reduces computational time by parallelizing MEME computations.

Step 2 In order to predict the binding affinity of the peaks, we need to represent each peak as a vector in the motif space. Let seq^i be the DNA sequence of peak i. Let $\text{seq}^i_{j,\ell}$ be the ℓ-length sub-sequence of seq^i, starting from position j. Let S^d be the PSSM of motif d. Let ℓ_i be the length of seq^i and ℓ_d be the length of motif d. We represent peak i as

[1] http://meme.sdsc.edu/

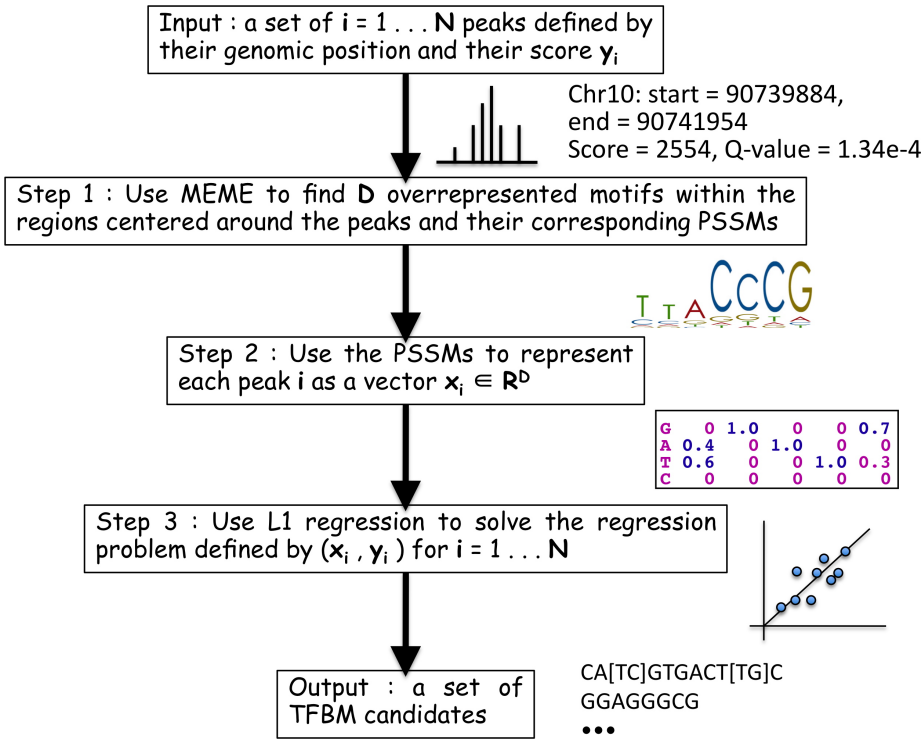


Figure 1: **Schematic view of the workflow of PeakRegressor.** PeakRegressor takes ChIP-Seq data as input and outputs a list of TFBM candidates and their weights that give the best regression accuracies.

vector $x_i \in R^D$, such that

$$x_{id} = \max_{j=1\ldots\ell_i-\ell_d+1} f(\text{seq}^i_{j,\ell_d}, S^d) - \max(S^d)$$

for $d = 1 \ldots D$. The quantity $f(\text{seq}^i_{j,\ell_d}, S^d)$ is a sum of log-odd scores, representing how well motif d matches sub-sequence seq^i_{j,ℓ_d}. Hence, the first term of the sum, x_{id}, corresponds to the best match when we slide motif d along sequence seq^i. The term $\max(S^d)$ is the maximum score achievable by any sequence matching with the motif d. Therefore we always have $x_{id} \leq 0$, with $x_{id} = 0$ for the best possible match.

Step 3 Quantities y_i to be fitted are the log values of the peak enrichment scores, as given by PeakSeq[9]. We can now solve the regression problem defined by (x_i, y_i) pairs for $i = 1 \ldots N$. Linear regression is a simple and popular approach, but is prone to overfitting. Hence, we choose to regularize the model with L1-norm, i.e., we want to minimize the sum of squared errors and the L1-norm of the regression coefficient vector:

$$\min_{b \in R^D} \beta \|b\| + \sum_{i=1}^{N} (b^T x_i - y_i)^2,$$

where $\beta > 0$ is a user-defined regularization coefficient. The L1-norm regression is able to select a small number of features that best explain the fitted quantity[10]. In our case, the features correspond to DNA motifs and hence, the result of this step is a set of motifs that best explain the binding signal values from ChIP-Seq dataset. We use Lasso, a popular algorithm for solving L1-norm regression. Lasso is available as part of the LARS package for R[2].

3 Results and Discussion

3.1 Input datasets

We use the ChIP-Seq data provided by[9]. For STAT1, we use 200-bp windows around the peak centers to define the peak sequences. For Pol II, the peak centers are not available and thus, we use the peak start and peak end coordinates to define the peaks. When the length of the resulting sequence is less than 200 bp, we enlarge it in both directions in order to reach 200 bp length. When the length is more than 4000 bp, we trim it in both directions in order to reach 4000 bp length. As a result, all the Pol II peak sequence lengths lie between 200 and 4000 bp.

For the regression analysis, we have to set the regularization parameter β. First, we define $\beta = 2^i$ for $i \in [-25, 25]$. Then for each value of β, we perform a 30 folds cross-validation. In each fold, we split the dataset into a training set and a test set, with a $90\% - 10\%$ ratio. The optimal value for β is the one which corresponds to the lowest prediction error on the test set. All the following results are averaged over the 30 folds cross-validation.

3.2 L1-norm log linear regression

We considered three settings before applying PeakRegressor. In the first setting, we considered all the peaks for regression. In the second setting, we excluded the peaks which showed no overlap with a promoter region (as defined by UCSC dataset[3]). In the third setting, we excluded the peaks which showed high Q-values ($> 10^{-3}$), as provided by [9]. Table 1 shows the averaged correlation coefficients between peak values and their predicted values in the test dataset. We can see that filtering peaks with their Q-values enhances the correlation coefficient for both STAT1 and Pol II. However, when filtering with promoter proximity, we observe than the correlation coefficient improves for Pol II but decreases for STAT1.

In Figure 2, we plot the STAT1 peak scores with two filtering methods such as Q-value $< 10^{-3}$ and promoter proximity in the test dataset against

[2]http://www-stat.stanford.edu/ hastie/Papers/LARS/

[3]http://hgdownload.cse.ucsc.edu/goldenPath/hg18/bigZips/upstream1000.fa.gz

Filtering method	*#Peaks (STAT1/Pol II)*	*STAT1*	*Pol II*
None	36,998 / 24,739	0.50	0.44
Promoter proximity	3,907 / 9,094	0.41	0.53
Q-value $< 10^{-3}$	16,639 / 17,580	0.65	0.66

Table 1: Influence of the peak filtering methods on the correlation coefficients between peak values and their predicted values in the test dataset. The correlation coefficients are averaged in 30-fold cross-validation.

their predictions by PeakRegressor. The correlation coefficient is as high as 0.65 between the peak and predicted values for the Q-value filtering, whilst it is as low as 0.41 for promoter proximity filtering. Interestingly, however, the data points that are selected by promoter proximity exist only in a biased region, leading to worse prediction.

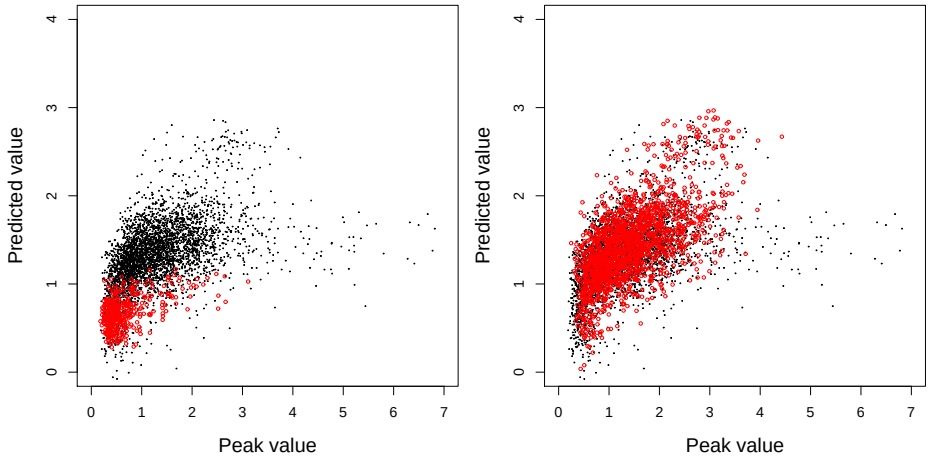


Figure 2: **The STAT1 regression results in test data with two filtering methods (shown by circle): promoter proximity (left) and Q-value (right).** The correlation coefficients between peak values and their predicted values are 0.45 and 0.65 for promoter proximity and Q-value filtering, respectively.

In Tables 2 and 3, we show the top 10 motifs for STAT1 and Pol II identified by PeakRegressor, respectively. The motifs are sorted according to the absolute values of their averaged regression coefficients. A motif with a positive (resp. negative) coefficient is thought to have a strengthening (resp. weakening) effect on the binding. In the case of STAT1, it is clear that our approach correctly identifies the classical GAS motif TTC[TC]N[GA]GAA as the main binding pattern[8]. Meanwhile, the Pol II binding motifs also contain Downstream Promoter Element [AG]G[AT][CT][GAC] and Initiator Site [TC][TC]AN[TA][TC][TC][3].

As the most important feature of PeakRegressor, it can give us a list of

STAT1	*Normalized coef.*
CA[TC]GTGACT[TG]C	1.
[TG]G[GTA][GC][AG]T**TT[CA]C[AGC][GA]GAA**[AC][TG]G[GA][GC]	0.96
TTC[CT][TG][GA]GAAAT[GC][CA][CA][CAT][AT][TCG][CG][CT]	0.72
[CT][TC]CA[GT]**TTCCAGGAA**[AT]T[CG][CAT]C[CT]	0.65
GGAGGGCG	-0.57
GGACGCCG	-0.56
A[CT]**TTC[TC][TG]GGAA**	0.56
TT[CA]C[TAG][GA]GAA[GA]T	0.55
A[TA]**TTCC[CT][GA]GAA**[AC]T[CG][AC]	0.48
TT[CA][TC][GA]GGAA[AG]	0.47

Table 2: List of putative STAT1 binding motifs. The classical GAS motifs are shown in boldface.

Pol II	*Normalized coef.*
T[AG]**A[GC][TAG]CA**[GCT]A[AC]AA	1.
A[GA]AA[AC][CA]AA[AC]AAA	0.78
C[ACT]**[GT][CG][CT][TA]C**C[AGT]CC[TA]	0.76
C[CT][CG][AT]GGCTGG[AG]G	0.68
TTTCTGC[CT][CT]TT[GT]	0.67
T[TA]T[TC]**[CA]CAGACT**[AT]	0.63
GGAGGGAGGC[AG]G	0.62
AC[AC][CA][AC][AT][AG]AGAAA	0.61
TTTG**T[CT][TA]T[TG][AC][AT]**T	0.54
AAA[AT][GC]AAA[AT]A[GA]A	0.54

Table 3: List of putative Pol II binding motifs. The Downstream Promoter Element and Initiator site motifs are shown in boldface.

putative composite motifs. Basically, it is difficult to evaluate whether a composite motif consists of the same motif or multiple (different) motifs. In order to identify the composite motifs, we proceed as follows. First, we consider the best set of motifs according to PeakRegressor (i.e. the set which corresponds to the best prediction accuracy). Among these, we select 136 motifs which have a normalized coefficient higher than 0.1. We use these motifs to represent each peak sequence as a binary vector, indicating whether a motif is present or not in the peak sequence. Then we cluster the resulting peak vectors using the K-means algorithm. Thus each cluster contains peak vectors which show similar motif patterns, i.e. sequences containing potential composite motifs.

Here we show an example of a composite motif that is responsible for STAT1 binding signals:

TCACA[TG]G[ACG] + [TC]TT[CA]C[CA][AG][GC][AC]A.

3.3 Candidate motifs and their potential rSNPs

Single or composite motifs found in the PeakRegressor system may reflect actual transcription factor binding sites. If a single nucleotide polymorphism (SNP) occurs within the sites, regulatory control of nighboring gene transcription will be perturbed, thus leading to genetic diseases in some cases[1]. Therefore, true binding sites may have SNPs less frequently than the non-binding sites. As an important verification, we check the number of known SNPs to be found within the STAT1 positions presented by PeakRegressor by using dbSNP database[4]. We find that 0.39% (138 for 35,156 bp) of mapped positions with 7 GAS-like motifs in Table 2 on the whole genome contains SNPs, while as much as 0.54% (18,097 for 3,344,439 bp) of all positions contains SNPs on the whole genome sequences. The statistical difference between the above two ratios (0.39 % vs. 0.54 %) is highly significant such as $p < 7.8 \times 10^{-5}$ by Fisher's exact test. These sites are possible candidates of rSNPs because the slight change within the motif may affect the change of gene expression level and might cause diseases.

References

[1] A. Ameur, A. Rada-Iglesias, J. Komorowski, and C. Wadelius. Identification of candidate regulatory snps by combination of transcription-factor-binding site prediction, snp genotyping and haplochip. *Nucleic acids research*, 37(12):e85+, July 2009.

[2] Harmen J. Bussemaker, Hao Li, and Eric D. Siggia. Regulatory element detection using correlation with expression. In *RECOMB '01: Proceedings of the fifth annual international conference on Computational biology*, page 86, New York, NY, USA, 2001. ACM.

[4] http://www.ncbi.nlm.nih.gov/SNP/

[3] J. E. Butler and J. T. Kadonaga. The rna polymerase ii core promoter: a key component in the regulation of gene expression. *Genes Dev*, 16(20):2583–2592, October 2002.

[4] Erin M. Conlon, X. Shirley Liu, Jason D. Lieb, and Jun S. Liu. Integrating regulatory motif discovery and genome-wide expression analysis. *PNAS*, 2003.

[5] Debopriya Das, Matteo Pellegrini, and Joe W. Gray. A primer on regression methods for decoding cis-regulatory logic. *PLoS Comput Biol*, 5(1):e1000269, 01 2009.

[6] Barrett C. Foat, Alexandre V. Morozov, and Harmen J. Bussemaker. Statistical mechanical modeling of genome-wide transcription factor occupancy data by matrixreduce. *Bioinformatics*, 22(14):e141–e149, 2006.

[7] Feng Gao, Barrett C. Foat, and Harmen J. Bussemaker. Defining transcriptional networks through integrative modeling of mrna expression and transcription factor binding data. *BMC Bioinformatics*, 2004.

[8] Gordon Robertson, Martin Hirst, Matthew Bainbridge, Misha Bilenky, Yongjun Zhao, Thomas Zeng, Ghia Euskirchen, Bridget Bernier, Richard Varhol, Allen Delaney, Nina Thiessen, Obi L. Griffith, Ann He, Marco Marra, Michael Snyder, and Steven Jones. Genome-wide profiles of stat1 dna association using chromatin immunoprecipitation and massively parallel sequencing. *Nat Meth*, 4(8):651–657, August 2007.

[9] Joel Rozowsky, Ghia Euskirchen, Raymond K. Auerbach, Zhengdong D. Zhang, Theodore Gibson, Robert Bjornson, Nicholas Carriero, Michael Snyder, and Mark B. Gerstein. Peakseq enables systematic scoring of chip-seq experiments relative to controls. *Nat Biotech*, 27(1):66–75, January 2009.

[10] Robert Tibshirani. Regression shrinkage and selection via the lasso. *J. Roy. Statist. Soc. Ser. B*, 58(1):267–288, 1996.

Critical Assessment of Massive Data Analysis (CAMDA 2009)

September 10, 2009

A Statistical Framework for the Analysis of ChIP-Seq Data

Pei Fen Kuan
Department of Statistics,
University of Wisconsin, Madison, WI 53706.

Guangjin Pan
Genome Center of Wisconsin, Madison, WI 53706.

James A. Thomson
Morgridge Institute for Research, Madison, WI 53707.
School of Medicine and Public Health,
University of Wisconsin, Madison, WI 53706.

Ron Stewart
Morgridge Institute for Research, Madison, WI 53707.

Sündüz Keleş
Department of Statistics,
Department of Biostatistics and Medical Informatics,
University of Wisconsin, Madison, WI 53706.

A Statistical Framework for the Analysis of ChIP-Seq Data

Pei Fen Kuan, Guangjin Pan, James A. Thomson, Ron Stewart and Sündüz Keleş

1 Introduction

Studying protein-DNA interactions is central to understanding gene regulation in molecular biology. Significant progress has been made in profiling transcription factor binding sites and histone modifications using chromatin immunoprecipitation (ChIP) techniques with high throughput microarrays (Buck and Lieb, 2004; Cawley et al., 2004). More recently, a new technology has been developed to directly sequence ChIP samples (ChIP-Seq) and offers whole-genome coverage at a lower cost. Most of the published work in ChIP-Seq are conducted via the Solexa/Illumina platform (Mikkelsen et al., 2007; Barski et al., 2007; Johnson et al., 2007). This high-throughput sequencing technology works by sequencing one/both ends of each fragment ($\sim 25-70$ bps) in the ChIP sample and generates millions of short reads/tags. These tags are then mapped to reference genome, followed by summarizing the total tag counts in each small genomic bin and analysis to detect enriched regions. Since then, a number of algorithms have been developed to detect enriched regions in ChIP-Seq data (Zhang et al., 2008a; Ji et al., 2008; Rozowsky et al., 2009). Enriched regions are detected with either a one sample analysis of the sequenced ChIP DNA sample or a two sample analysis by comparing sequenced ChIP sample to reference genomic control (Input DNA) sample. Although the sequencing-based technology offers promising results for surveying large genomes at higher resolutions, it is not free of sequencing and other source of biases (Dohm et al., 2008; Vega et al., 2009; Rozowsky et al., 2009). Despite this, most of the existing tools do not consider such biases. Furthermore, two recent publications (Teytelman et al., 2009; Auerbach et al., 2009) observed that high throughput sequencing of Input DNA reveals open chromatin regions and regions of other biological interest and challenged the use of Input control for detecting enriched regions. This motivates revisiting the one sample and two sample analysis.

In this paper, we study sources of bias in the underlying data generating process of ChIP-Seq technology by utilizing sequenced Naked DNA (non-cross-linked, deproteinized DNA) and develop a model that captures the background signal in the ChIP-Seq data. We then propose mixture models for both one and two sample analyses of ChIP-Seq data and apply these to analyze STAT1 dataset. Our modeling framework incorporates the variability in both the mappability and GC-content of regions on the genome and sequencing depths of the samples. We show that our model fits very well on real data and provides a fast model-based approach for ChIP-Seq data analysis.

2 A non-homogeneous zero inflated negative binomial regression model for the background distribution

Standard pre-processing and analysis of ChIP-Seq data involve retaining only tags that align uniquely to the genome. This induces apparent bias in the subset of tags used for the analysis. This factor is usually ignored in modeling the background/non-enriched distribution generating ChIP-Seq data. The mappability bias is apparent as illustrated in Figure 1 of Rozowsky et al. (2009). These authors provided an efficient code to score the number of uniquely mappable nucleotides within a genomic window (mappability score) and introduced the PeakSeq algorithm for analyzing ChIP-Seq data. To the best of our knowledge, PeakSeq is the only software that incorporates mappability bias, albeit in an ad hoc manner, by performing a local permutation in a pre-specified genomic window. Within each genomic window, all the nucleotides are assumed to have the same mappability score. However, the size of the genomic window needs to be calibrated in this local permutation scheme. A small genomic window would result in insufficient tags for

permutation, while a large genomic window would downplay the effect of mappability bias and decrease the resolution.

In addition to mappability bias, the observed tag counts have been shown to be correlated with GC content (Dohm et al., 2008). In particular, regions with higher GC content exhibit increasing number of tags (Dohm et al., 2008). Since both the mappability and GC bias are characteristics of underlying genomic DNA sequence, observed tag counts from naked DNA (non-cross-linked, deproteinized) sample provides a natural platform to study such biases. We utilized naked DNA high-throughput sequencing data of HeLa S3 cells from Gene Expression Omnibus under accession number GSE14022 and the naked DNA sample from human embryonic stem (hESCs) cells from the Thomson Lab, UW-Madison, for developing a model of observed tag counts that captures the mappability and GC bias.

Let Y_j denote the total number of overlapping extended tag counts in bin j and M_j be the average mappability score, where $0 \leq M_j \leq 1$. Let GC_j denote the average GC content in bin j which is calculated in a similar manner as M_j to account for tag extensions. In Figure 1, we plot the average bin level tag counts against M_j and GC_j for both the HeLa S3 and hESCs naked DNA samples. This plot indicates that the tag counts are increasing in both M_j and GC_j. This provides evidence for mappability and GC biases in the observed tag counts of the sequenced naked DNA sample.

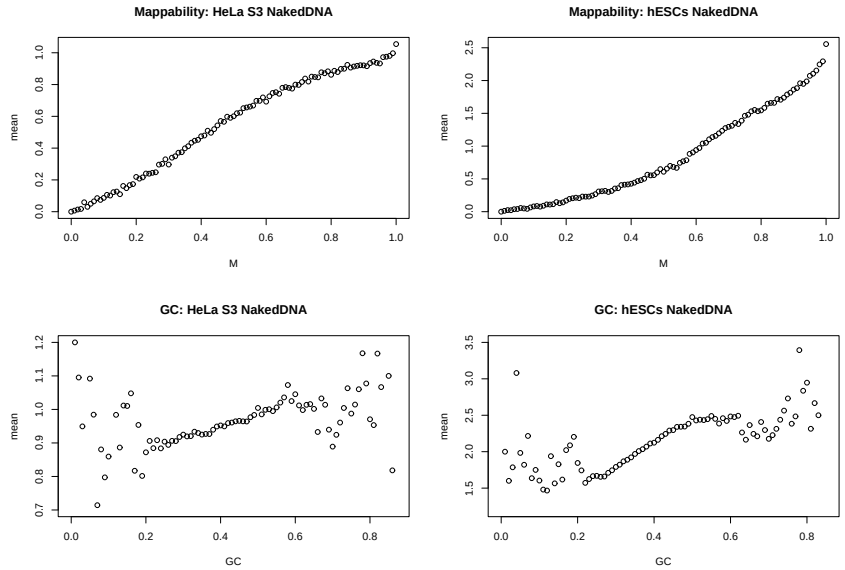


Figure 1: *Mappability and GC bias in sequenced naked DNA samples.* Top row plots mean tag counts against the mappability score M_j. Bottom row plots mean tag counts against the GC content GC_j.

Since bins with zero mappability are never sequenced, this gives rise to excess zeroes in the observed data. We introduce a Bernoulli random variable B_j that takes value 1 if bin j is sequenced. Consider the following general formulation for modeling the background (non-enriched) distribution:

$$\begin{aligned} Y_j|\mu_j &\sim N_j I(B_j = 1), \\ B_j|p_j &\sim Ber(p_j), \\ p_j &\sim Beta(M_j, v), \\ N_j|\mu_j &\sim g(\mu_j), \end{aligned}$$

where N_j measures tag counts arising from non-specific sequencing biases. We choose a beta-binomial family for B_j instead of the more common logit link function so that $P(B_j = 0|M_j = 0) = 1$, i.e., a bin is never sequenced if it has zero mappability score which is consistent with the pre-processing step that only retains uniquely aligned tags.

To ascertain if inclusion of M_j and GC_j improves the model fit, we first ignore the Bernoulli indicator for simplicity and consider fitting a generalized linear model with Poisson family, i.e., $Y_j \sim Po(\mu_j)$ for (1)

$\mu_j = \exp(\beta_0)$ (None), (2) $\mu_j = \exp(\beta_0 + \beta_1 M_j)$ (M_j only), (3) $\mu_j = \exp(\beta_0 + \beta_1 GC_j)$ (GC_j only), (4) $\mu_j = \exp(\beta_0 + \beta_1 M_j + \beta_2 GC_j)$ (M_j and GC_j). We compare the different μ_j formulations based on BIC scores. In Table 1, the BIC scores for the model which includes both M_j and GC_j are the lowest for both naked DNA sample. Therefore, we choose $\mu_j = \exp(\beta_0 + \beta_1 M_j + \beta_2 GC_j)$.

BIC	None	M_j only	GC_j only	M_j and GC_j
HeLa S3 nakedDNA	5694641	5069068	5365597	5046438
hESCs nakedDNA	9454619	7674979	8539502	7598011

Table 1: *Model selection based on BIC scores.* Each cell reports the BIC score under different μ_j formulations.

Next we consider two candidate models for $N_j \sim g(\mu_j)$: (1) $g(\mu_j) \sim Po(\mu_j)$ and (2) $g(\mu_j) \sim NegBin(a, a/\mu_j)$. This gives rise to a zero-inflated Poisson regression (ZIPreg) under (1) and a zero-inflated Negative Binomial regression (ZINBreg) under (2). Figure 2 compares simulated data from the fitted ZIPreg and ZINBreg against the actual data for both HeLa S3 and hESCs naked DNA samples. In both samples, ZIPreg is unable to capture high tag counts as shown by the lighter tail compared to the distribution of the actual data. On the other hand, ZINBreg provides a better fit to the actual data and is able to trace the over-dispersion in the distribution of the actual data as displayed in Figure 2.

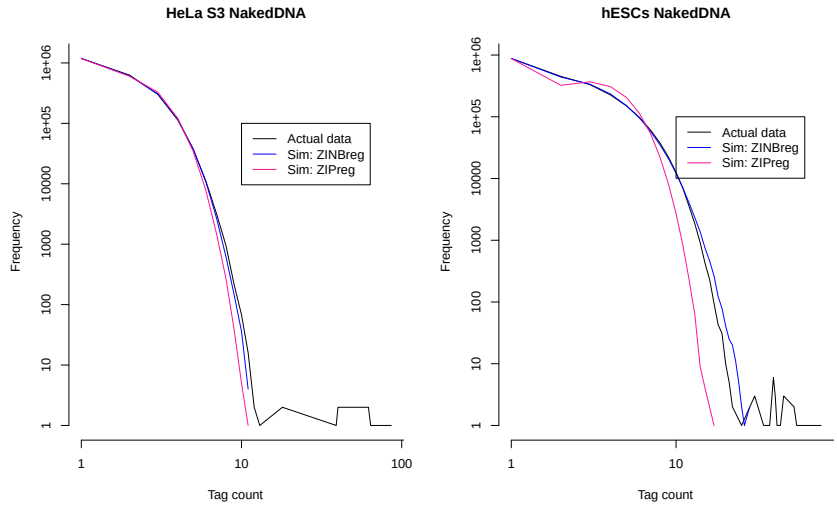


Figure 2: *Goodness of fit for naked DNA samples.* Each panel compares the simulated data against the actual data.

3 One sample problem

Observed counts in ChIP-Seq data can be considered as coming from two populations of genomic regions, namely, protein bound and unbound regions. Exploratory analysis in Section 2 motivates a background model for tags from unbound regions. Next we outline this model and also propose a model for enriched tag counts. Let Y_j denote the observed tag counts for bin j, and Z_j be the unobserved random variable specifying if bin j comes from enriched ($Z_j = 1$) or non-enriched ($Z_j = 0$) distribution. We define $Y_j \sim N_j I(B_j = 1)$ given $Z_j = 0$. Here $N_j \sim NegBin(a, a/\mu_j)$ measures non-specific sequencing which is related to both M_j and GC_j, while B_j indicates if bin j is sequenced and it depends on M_j which gives rise to a ZINBreg, as described in Section 2.

We let $Y_j = N_j + S_j$ where S_j represent the true signal due to enrichment, and model $S_j \sim NegBin(b, c)$. Therefore, the observed tag counts can be written as a mixture model $P(Y_j = y) = \pi_0 P(Y_j = y|Z_j = 0) + (1 - \pi_0) P(Y_j = y|Z_j = 1)$, where $\pi_0 = P(Z_j = 0)$, $Y_j = y|Z_j = 0 \sim$ZINB and $Y_j = y|Z_j = 1 \sim N_j + S_j$.

Although there is no closed form for $Y_j|Z_j = 1$ (convolution of two negative binomial distributions), we have an efficient and robust procedure for estimating all the unknown parameters $(v, \beta_0, \beta_1, \beta_2, a, b, c, \pi_0)$ in the model. To identify a set of bins which are enriched while controlling false discovery rate (FDR) at level α, we use a *direct posterior probability approach* of Newton et al. (2004).

We illustrate our proposed model on the ChIP-Seq data measuring STAT1 binding in interferon-γ-stimulated HeLa S3 cells from Rozowsky et al. (2009) for Chromosome 13 and 21. Figure 3 compares the simulated data from the mixture model (red lines) against the actual STAT1 data for Chromosome 13 and 21, respectively. As evident in this figure, our proposed mixture model provides a good fit to the actual data. These model based fits are comparable to simulation based approach for ChIP-Seq data studied in Zhang et al. (2008b).

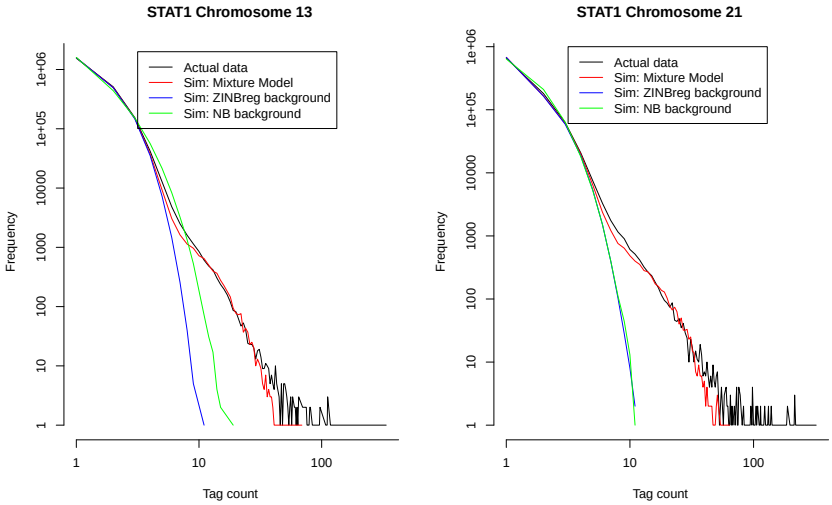


Figure 3: *Goodness of fit for STAT1 ChIP-Seq sample.* Red lines correspond to simulated data from the mixture model. Blue lines correspond to simulated data from the ZINBreg null model of no enrichment. Green lines correspond to simulated data from negative binomial (Ji et al., 2008) null model of no enrichment.

At FDR of 0.05, we identify 1328 and 776 enriched regions for Chromosome 13 and 21, respectively, based on one sample analysis. As in Rozowsky et al. (2009), we compare the identified regions against the ChIP-chip target sites validated independently by qPCR (Euskirchen et al., 2007). 15 out of 21 regions validated positive by qPCR are common to our peak set. Similarly, PeakSeq also detects the same 15/21 qPCR validated regions. The 6 regions missed by both our proposed mixture model and PeakSeq have low tag counts (average of 1.5 tag per 50 bps), high mappability scores ($\sim$ 1) and moderate GC content ($\sim$ 0.4) in the STAT1 ChIP-Seq data. On the other hand, both our method and PeakSeq only detect 1 out of 21 regions that are validated negative by qPCR. Four contiguous bins in this region have high tag counts (average of 22.25 tag per 50 bps) in the STAT1 ChIP-Seq data. We also compare our results to CisGenome (Ji et al., 2008) which assumes independent and identically distributed negative binomial null distribution for all the bins. The right panel of Figure 4 shows that the FDR control for CisGenome is not continuous. That is, the FDR level for declaring all bins with $\geq$6 counts to be enriched is 0.099, whereas the FDR level for declaring all bins with $\geq$7 to be enriched is 0.038 and the FDR for declaring all bins with $\geq$8 counts is 0.014 in Chromosome 13. We obtain 1047 (811) enriched regions under FDR of 0.038 (0.014) for Chromosome 13 using the negative binomial background as in CisGenome. In contrast, the FDR control based on our mixture model is almost continuous as shown in the left panel of Figure 4. In a way, our model provides a mechanism to discriminate bins with same tag counts and thereby facilitates continuous FDR control. We are currently exploring practical implication of this in terms of biological discovery.

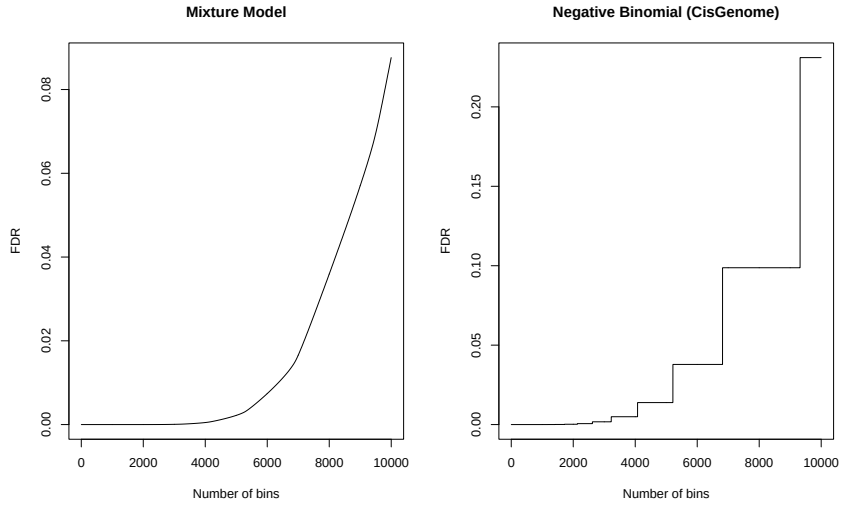


Figure 4: *Comparison of FDR control.* Left panel plots the FDR control against the number of bins declared to be significant under our proposed mixture model. Right panel plots the FDR control under the negative binomial null distribution as in CisGenome (Ji et al., 2008).

4 Two sample problem

Next, we introduce our modeling framework for inferring enriched regions relative to a control experiment in two sample problem. Let (Y_j, X_j) be the observed sample 1 (treatment) and sample 2 (control) tag counts on bin j. Similarly, we define Z_j to be the unobserved random variable specifying the underlying latent state of bin j. Let D_X and D_Y be the sequencing depths of control and treatment experiments, respectively. Most of the current approach in the analysis of two sample ChIP-Seq apply linear scaling to the observed tag counts to normalize for the difference in sequencing depths (Zhang et al., 2008a; Rozowsky et al., 2009), which is undesirable under the assumption of commonly used count distributions. Another popular strategy is to randomly sample D_X counts from Y (assuming $D_Y > D_X$). This is again undesirable, since using only a fraction of the original data results in some information loss.

Let λ_{jX} and λ_{jY} denote the bin specific latent mean tag counts of X_j and Y_j. We assume that X_j and Y_j are random samples from $p_X(.|\lambda_{jX}) = Po(\lambda_{jX}\mu_j D_X)$ and $p_Y(.|\lambda_{jY}) = Po(\lambda_{jY}\mu_j D_Y)$ respectively, where $\mu_j = \exp(\beta_0 + \beta_1 M_j + \beta_2 GC_j)$ and

$$\begin{aligned} \lambda_{jX} \geq \lambda_{jY} \text{ if } Z_j = 0, \\ \lambda_{jX} < \lambda_{jY} \text{ if } Z_j = 1. \end{aligned}$$

As in Newton et al. (2004); Keleş (2007), we assume that the latent mean counts $(\lambda_{jX}, \lambda_{jY})$ to be a random pair from an unknown bivariate distribution f, which is taken to be a mixture over the two hypotheses of interest:

$$f(\lambda_{jX}, \lambda_{jY}) \quad = \quad P(Z_j = 0) f_0(\lambda_{jX}, \lambda_{jY}) + P(Z_j = 1) f_1(\lambda_{jX}, \lambda_{jY})$$

where the densities f_0 and f_1 describe the fluctuations of the means within each hypothesis. The joint distribution of λ_{jX} and λ_{jY} is related to a one-dimensional base distribution $\pi = \mathcal{G}a(a, a)$. Under this model assumption, we derive the marginal distribution of observed tag counts in the treatment sample conditional on the sum of tag counts from both samples.

Therefore, our modeling approach automatically accounts for the difference in sequencing depths and bypass the problem of linear based normalization. Given the marginal distribution of Y_j conditioned on $X_j + Y_j$, inference for identifying enriched regions relative to input control at a particular FDR follows from one sample problem. We are currently exploring practical advantages of this framework for two sample analysis.

5 Conclusions and on going work

We investigated the effect of mappability and GC biases that arise in high-throughput sequencing data. We showed that these effects are significant in naked DNA samples, which represent the background distribution of no enrichment. We proposed a zero inflated negative binomial regression model (ZINBreg) that incorporates both the mappability and GC biases and showed that this model provides an excellent fit as background distribution. We then utilized this background model for one sample analysis.

In one sample analysis, we considered a mixture modeling approach for the observed tag counts, in which the non-enriched distribution is modeled with a ZINBreg and the enriched distribution is modeled with a convolution of two negative binomials. We showed that the proposed mixture model fits the actual STAT1 ChIP-Seq data quite well, and further demonstrated that this model is able to achieve good operating characteristics based on independently validated qPCR results. We are currently applying our proposed modeling approach for two sample analysis (Section 4) to automatically account for difference in sequencing depths in identifying enriched regions in the presence of Input or other type of control and in detecting differential enrichments between two ChIP-Seq samples. Comparison of one sample analysis with two sample analysis that utilizes different controls (input DNA, naked DNA or IgG control) is very likely to yield more insights on ChIP-Seq data analysis. We expect to have further results on this in time for CAMDA 2009.

References

Auerbach, R. K., Euskirchen, G., Rozowsky, J., Lamarre-Vincent, N., Moqtaderi, Z., Lefrançois, P., Struhl, K., Gerstein, M., and Snyder, M. (2009), "Mapping accessible chromatin regions using Sono-Seq," *PNAS*, 106, 14926–14931.

Barski, A., Cuddapah, S., Cui, K., Roh, T., Schones, D., Wang, Z., Wei, G., Chepelev, I., and Zhao, K. (2007), "High-resolution profiling of histone methylations in the human genome," *Cell*, 129, 823–837.

Buck, M. and Lieb, J. (2004), "ChIP-chip: considerations for the design, analysis, and application of genome-wide chromatin immunoprecipitation experiments ," *Genomics*, 84, 349–360.

Cawley, S., Bekiranov, S., Ng, H., Kapranov, P., Sekinger, E., Kampa, D., Piccolboni, A., Sementchenko, V., Cheng, J., Williams, A., Wheeler, R., Wong, B., Drenkow, J., Yamanaka, M., Patel, S., Brubaker, S., Tammana, H., Helt, G., Struhl, K., and Gingeras, T. (2004), "Unbiased mapping of transcription factor binding sites along h uman chromosomes 21 and 22 points to widespread regulation of non-coding RNAs," *Cell*, 116, 499–511.

Dohm, J., Lottaz, C., Borodina, T., and Himmelbauer, H. (2008), "Substantial biases in ultra-short read data sets from high-throughput DNA sequencing," *Nucleic Acids Research*, 36, e105.

Euskirchen, G., Rozowsky, J., Wei, C., Lee, W., Zhang, Z., Hartman, S., Emanuelsson, O., Stolc, V., Weissman, S., Gerstein, M., Ruan, Y., and Snyder, M. (2007), "Mapping of transcription factor binding regions in mammalian cells by ChIP: Comparison of array- and sequencing-based technologies," *Genome Research*, 17, 898–909.

Ji, H., Jiang, H., Ma, W., Johnson, D., Myers, R., and Wong, W. (2008), "An integrated software system for analyzing ChIP-chip and ChIP-seq data," *Nature Biotechnology*, 26, 1293–1300.

Johnson, D., Mortazavi, A., Myers, R., and Wold, B. (2007), "Genome-wide mapping of in Vivo protein-DNA interactions," *Science*, 316, 1749–1502.

Keleş, S. (2007), "Mixture modeling for genome-wide localization of transcription factors," *Biometrics*, 63, 10–21.

Mikkelsen, T., Ku, M., Jaffe, D., Issac, B., Lieberman, E., Giannoukos, G., Alvarez, P., Brockman, W., Kim, T., Koche, R. P., Lee, W., Mendenhall, E., O'Donovan, A., Presser, A., Russ, C., Xie, X., Meissner,

A., Wernig, M., Jaenisch, R., Nusbaum, C., Lander, E., and Bernstein, B. (2007), "Genome-wide maps of chromatin state in pluripotent and lineage-committed cells," *Nature*, 448, 653–560.

Newton, M., Noueiry, A., Sarkar, D., and Ahlquist, P. (2004), "Detecting differential gene expression with a semiparametric hierarchical mixture model," *Biostatistics*, 5, 155–176.

Rozowsky, J., Euskirchen, G., Auerbach, R., Zhang, D., Gibson, T., Bjornson, R., Carriero, N., Snyder, M., and Gerstein, M. (2009), "PeakSeq enables systematic scoring of ChIP-Seq experiments relative to controls," *Nature Biotechnology*, 27, 66–75.

Teytelman, L., Özaydin, B., Zill, O., Lefrançois, P., Synder, M., Rine, J., and Eisen, M. B. (2009), "Impact of Chromatin Structures on DNA Processing for Genomic Analysis," *PLoS One*, 4, e6700.

Vega, V., Cheung, E., Palanizamy, N., and Sung, W. (2009), "Inherent signals in sequencing-based chromatin-immunoprecipitation control libraries," *PloS One*, 4, e5241.

Zhang, Y., Liu, T., Meyer, C., Eeckhoute, J., Johnson, D., Bernstein, B., Nussbaum, C., Myers, R., Brown, M., Li, W., and Liu, X. (2008a), "Model-based Analysis of ChIP-Seq (MACS)," *Genome Biology*, 9, R137.

Zhang, Z., Rozowsky, J., Snyder, M., Chang, J., and Gerstein, M. (2008b), "Modeling ChIP sequencing in silico with applications," *PLoS Computational Biology*, 4, e1000158.

Comparative Analysis of ChIP-seq Data using Mixture Model (Extended Abstract)

Cenny Taslim[1,2,*], Tim Huang[1], Shili Lin[2] and Kun Huang[3,4]

[1]Department of Human Genetics, [2]Department of Statistics, [3]Department of Biomedical Informatics, [4]OSUCCC Biomedical Informatics Shared Resources

The Ohio State University, Columbus, OH 43210.

Introduction

Antibody-based Chromatin Immunoprecipitation assay followed by massive sequencing technology (ChIP-seq) has enable scientist to study protein-DNA binding in shorter time with less error and cheaper cost than ChIP-chip experiment. However, the large amount of data being produced and errors in procedure such as tags amplification, base-calling, image processing, sequence alignment pose new challenges in analyzing this high-throughput data. In order to process and separate biological signal from noise, computational and statistical approaches are required. One of them is data normalization which is very critical when comparing results across multiple samples. We introduced a nonlinear normalization algorithm and a mixture modeling method for comparing ChIP-seq data from multiple samples and characterizing genes based on their RNA polymerase II (Pol II) binding patterns (Taslim *et al.*, 2009). Here, we are going to apply the same nonlinear normalization on the contest data sets comparing Pol II ChIP-seq with input DNA then use model-based classification and *fdr* (false discovery rate) to identify genes that are associated with enriched binding sites.

Methods

Preprocessing and determining putative binding sites

ChIP-seq data for Pol II in unstimulated HeLa S3 (an immortalized cervical cancer) cell line are compared against matching sequenced input DNA control data sets (Rozowsky *et al.*, 2008). Pol II HeLa data sets has three replicates. Each replicates has two, five, and four lanes, respectively. The matching input DNA has one replicate and thirteen lanes. For each replicate in one data set, we add up all the lanes. The different number of lanes in each experiment will be handled in the normalization process. Then, if the sample has more than one replicate, we calculate the average sequence counts for all replicates.

The data sets we used have the alignment information produced by ELAND (Cox, unpublished software). We take only the sequence reads that are mapped uniquely to the genome (i.e. U0, U1, U2 type of match). In ChIP-seq protocol, a tag is sequenced by reading both ends of the ChIP fragment. Hence, the real binding sites are unknown. However, since Pol II are known to bind throughout promoter, upstream, and downstream regions of the activated gene, we feel it is unnecessary to do any shifting in our analysis.

Normalization

Let x_{ij} be the Pol II binding quantity for bin i (i = 1, ..., n), where n is the total number of bins in a chromosome and j = 1, 2 refers to control (input DNA) and treatment (Pol II Hela) respectively. For each chromosome, we divide the region into bins of size 1000 nt (nucleotide) and report the number of sequences in each bin. Thus, x_{ij} is the total number of sequence reads that are mapped between location (i-1) × 1000 and i × 1000 + 1 in sample j. If a particular sample has replicates, x_{ij} is the average of number of sequence reads that are mapped in all replicates. 1k-nt bin size is chosen to balance between number of data points and resolution.

The purpose of normalization is to enable comparison between multiple experiments. Due to various errors throughout the processing the high-throughput data, the results of ChIP-seq data in different experiments will have bias and error. For example, experiments which have more number of lanes will produce more sequence reads compare to the ones with less lanes. Thus, without normalization, differential binding sites will be discovered because the effect of lanes not due to biological differences. Here, we use a three-step normalization process. It is based on locally weighted polynomial least square regression (Cleveland, 1988). This nonlinear method does not assume any relationship in the data.

In the first step, we perform sequencing depth normalization, making the total number of sequences to be equal in both control and sample data.

Equation 1

$$\bar{x}_{i2} = x_{i2}\frac{\sum_{i=1}^{n} x_{i1}}{\sum_{i=1}^{n} x_{i2}} \qquad \forall i = 1, \ldots, n;$$

Second, the mean of the difference counts are estimated and then subtracted from the observed difference to normalize the data with respect to the mean.

Equation 2

$$D_{seq.mean.norm} = (\bar{x}_{i2} - \bar{x}_{i1}) - loess\left[(\bar{x}_{i2} - \bar{x}_{i1}) \sim \left(\frac{\bar{x}_{i2} + \bar{x}_{i1}}{2}\right)\right]$$

Where loess[.] are the fitted values obtained by regressing the observed counts of difference on the mean.

Lastly, the normalized data calculated in the second step are divided by the estimated mean of variance obtained by regressing the absolute of the normalized data on the observed mean counts.

Equation 3

$$D_{seq.mean.var.norm} = \frac{D_{seq.mean.norm}}{loess\left[D_{seq.mean.norm} \sim \left(\frac{\bar{x}_{i2} + \bar{x}_{i1}}{2}\right)\right]}$$

Finite Mixture Model for Differential Genes Selection

In order to identify genes that are associated with differential binding quantity in HeLa Pol II vs. Input DNA, we fit a finite mixture model and model-based classification using the idea of false discovery rate. We assume that the data comes from three groups, i.e. no-change, positive- and negative-differential.

The no-change group is assumed to come from a mixture of *K*-component Normal distributions, where *K* is estimated from the data. The positive- and negative-differential groups are assumed to follow an Exponential and the mirror of Exponential distribution respectively. In order to identify genes associated with differential binding quantity, the data $d_i \in D_{seq.mean.norm}$ are grouped based on RefSeq gene database. Let W equals to the total number of genes in the database, then for each gene region *w*, we have $G_w = \sum d_i, \ \forall i \ \in R$ where R is the sets of fragments within a gene region $w \in \{1, \ldots, W\}$. An empirical distribution is fitted based on G_w.

Equation 4

$$f(g, \Psi) = \sum_{k=1}^{K} [\gamma_k \phi(g, \mu_k, \sigma_k^2)] + \pi_1 E_1[-g \times I(g < -\xi_1)] + \pi_2 E_2[-g \times I(g > \xi_2)]$$

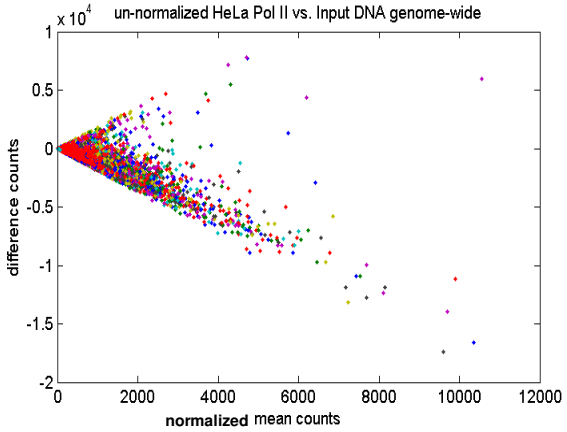
Where $f(g, \Psi)$ is the unknown function of the observed data G_w and the vector of unknown parameters (Ψ) ; $\phi(.)$ denotes the Normal density function with mean μ_k **and variance** σ_k^2 ; I(.) is the indicator function which equals to 1 if condition specified in () is satisfied and 0 otherwise; $\xi_1, \xi_2 > 0$ are the location parameter which in practice may be estimated by $\hat{\xi}_1 = |\max(g_w < 0)|$ and $\hat{\xi}_2 = |\min(g_w > 0)|$. By maximizing the likelihood function using Expectation-maximization (EM) algorithm and selecting *K* using AIC (*Akaike, 1973*), a set of optimal parameters ($\Psi^* $) is obtained. The local false discovery rate is calculated as follows:

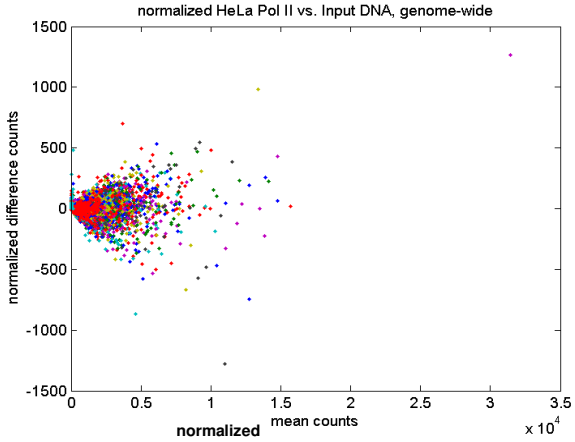
Equation 5

$$fdr(g_w) = \frac{\sum_{k=1}^{K} [\gamma_k \phi(g, \mu_k, \sigma_k^2)]}{f(g, \Psi)}$$

Results

We apply the three-step nonlinear normalization and statistical modeling methods described above to the study comparing the Pol II binding quantity in HeLa cell line and matching input DNA.





(a) (b)

Figure 1 Plots to show the effects of the three-step normalization on the genome-wide data. Each point is a gene. Colors refer to different chromosome (**a**) raw data with clear bias toward the negative direction. (**b**) data normalized with respect to sequencing depth, mean and variance.

Since Pol II has three replicates, we calculate the averages of all the replicates and compare it with input DNA. As demonstrated in Figure 1a, the raw data before any normalization is biased toward negative difference counts. This bias is due to the fact that there are more lanes of data in input DNA than in Pol II HeLa cells. Other processing error can also contribute to the bias toward larger mean counts. Without doing any normalization, longer genes (which will have larger mean counts) will be called as differential in many more times than shorter genes. Furthermore, more differential genes will be associated with less binding quantity. These systematic effects are what we are trying to correct by the three-step normalization. As shown in Figure 1b, the normalized data are no longer overwhelmed by bias toward the negative counts and longer genes. By normalizing the data with respect to depth, mean and variance, we are able to spread the points evenly around zero and reduce systematic error. Next, we fit the normalized difference with the mixture model using EM algorithm. Due to time constraints, the EM algorithm was re-initialized 125 times to prevent it from getting stuck in a local optimum. Each time the EM step was terminated either after 2000 iterations or when the improvement is not greater than 10^{-16}. Figure 2 shows the mixture of two exponential and three normal components which are found to best represent the data. Applying model-based classification, we find 294 genes are associated with differential binding sites using $fdr < 0.1$. 178 of these genes are associated with increased binding sites which indicate genes with significant amount of Pol II binding, while 116 of them are associated with decreased binding sites indicating binding sites for other proteins.

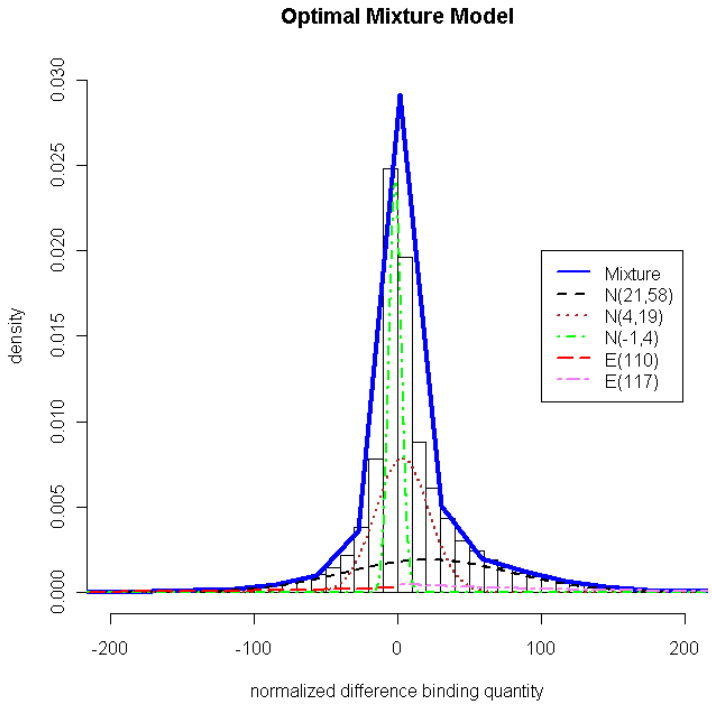


Figure 2. The fit of the best mixture model on the normalized HeLa Pol II vs. input DNA data. The bars are the observation data. The optimal mixture model and its individual components are plotted. Blue (solid) line represents the best mixture model (mixture of two exponential and three normal components) imposed on the histogram of the normalized difference of binding quantity. Black (dashed), brown (dotted) and green (dot-dash) lines represent Normal components with (μ_1=-21, σ_1=58), (μ_2=4,

σ_2=19), (μ_3=*-1*, σ_3=4), respectively. Red (long dash) and magenta (two-dash) represent Exponential components with β_1 = 110and β_2 =117 respectively.

References

Akaike,H. (1973) Information theory and an extension of the maximum likelihood principle. In *International Symposium on Information Theory*, 2nd, Tsahkadsor, Armenian SSR, pp. 267–281.

Cleveland,W.S. (1988) Locally-weighted regression:An approach to regression analysis by local fitting. *J. Am. Stat. Assoc.*, 85, 596–610.

Khalili,A. *et al.* (2009) A robust unified approach to analyzing methylation and gene expression data. *Comput. Stat. Data Anal.*, 53, 1701 – 1710.

Rozowsky,J. *et al.* (2009) Peakseq enables systematic scoring of ChIP-seq experiments relative to controls. *Nat. Biotechnol.*, **27**, 66–75.

Taslim, C. *et al.* (2009) Comparative study on ChIP-seq data: normalization and binding pattern characterization", *Bioinformatics*, 25, 18, pp. 2334-2340

Scoring of ChIP-seq experiments by modeling large-scale correlated tests

Pingzhao Hu[1*], Zhi Wei[2*], Zhuozhi Wang[1], Andrew D. Paterson[1,3], Joseph Beyene[1,3], Stephen W Scherer[1,4]

[1]The Centre for Applied Genomics, The Hospital for Sick Children, Toronto, Ontario, Canada
[2]Department of Computer Science, New Jersey Institute of Technology, Newark, NJ, USA
[3]Dalla Lana School of Public Health, University of Toronto, Health Sciences Building 155 College St, Toronto, ON, M5T 3M7, Canada
[4]Department of Molecular and Medical Genetics, University of Toronto, Toronto, Ontario, Canada.

*Equally contributed to this work
Correspondence should be addressed to P.H. (phu@sickkids.ca)

Abstract

Chromatin immunopercipitation followed by direct sequencing (ChIP-Seq) plays key roles in profiling DNA-protein interactions and determine transcription factor binding sites. One of major challenges in scoring the ChIP-Seq experiments is to control False Discovery Rate (FDR). The standard FDR procedures, due to ignore dependence among tests, suffer from loss of efficiency. Here we exploit to use the dependency information of adjacent regions to improve the rankings of enriched transcription factor binding sites.

1. Introduction

Because of the recent advancements in new high-throughput sequencing technologies, ChIP-seq has become a popular tool for genome-wide mapping of in vivo protein DNA association. In analyzing ChIP-seq data, it is typical to test hundreds of thousands of segments simultaneously. For example, Rozowsky et al. (2009) evaluated more than 120,000 potential segments for identifying transcription factor binding sites in ChIP-seq Data. Therefore, it is necessary to control the false discovery rate (FDR, Benjamini and Hochberg, 1995). The FDR controlling procedures have been successfully applied in many large-scale studies such as microarray experiments, genome-wide association studies, ChIP-seq experiments, among others (Tusher *et al*., 2001; Storey and Tibshirani, 2003, Sabatti *et al*., 2009; Rozowsky et al. 2009). Despite the increasing popularity, most commonly used FDR procedures are based on thresholding the ranked *p*-values (Benjamini and Hochberg, 1995; Storey, 2002), where the dependence among tests is ignored. The "do nothing" approaches may suffer from severe loss of efficiency. For example, Sabatti et al. (2003) found that the Benjamini and Hochberg (BH) procedure suffers from increased power loss with increased dependency among markers in a genome scan. The works of Nyholt et al. (2004) and Conneely et al. (2007) showed that

by exploiting the dependency structure, more precise FDR control can be achieved and hence the statistical power can be improved. Qiu *et al*., 2005; Efron, 2007 explored the correlation effects of SNPs at adjacent genomic loci on FDR analyses.

The development of a multiple testing procedure essentially involves two steps: ranking the hypotheses and choosing a cutoff along the rankings. The ranking step is more fundamental. Different from Nyholt et al. (2004) and Conneely et al. (2007) that utilize the dependency to choose the cutoff, Wei et al. (2009) proposed to utilize the dependency to create more efficient rankings. The proposed procedure uniformly improves all *p*-value based procedures by re-ranking the importance of all SNPs. They explored to use a hidden Markov Model (HMM) to model the dependency of adjacent SNPs. Therefore, when deciding the significance level of a SNP, the neighboring SNPs are taken into account. They called the new multiple testing procedure as pooled local index of significance (PLIS). The goal of this analysis is to compare how that conventional "do nothing" procedures, that is the BH multiple testing proceeding, can be greatly improved by PLIS procedure with exploiting the HMM dependency in ChIP-seq experiments.

2 Statistical Methods

The HMM is an effective model to characterize the dependency among neighboring segments. This strategy has been widely used in copy number variation analysis (Colella et al., 2007; Wang et al., 2007). Here we use this approach to model enrichment of mapped tags in adjacent segments on chromosomes. In an HMM, each segment has two hidden states: enriched or not enriched, and the states of all segments along a chromosome form a Markov chain. The observed mapped tag data are generated conditionally on the hidden states via an observation model.

2.1 Modeling of Enrichment of Mapped Tags Using A Hidden Markov Model

Suppose there are m_k no overlapping segments of length $L_{segment}$ (typically 1 Mb) on chromosome $k, k = 1,\cdots,K$. The total number of segments is $m = \sum_{k=1}^{K} m_k$. In order to determine whether a given target segment *r* is enriched in the number of mapped tags from the ChIP-seq sample compare to input-DNA control, it is typical to calculate a p-value from a suitable statistical test, such as the cumulative distribution function for the binomial distribution as done by Rozowsky et al. (2009). z-values related to the p-values can be obtained using appropriate transformations for further analysis.

Let $\theta_k = \{\theta_{k1},\cdots,\theta_{km_k}\}$ be the underlying states of the segment sequence on chromosome k from the 5' end to the 3' end, where $\theta_{ki} = 1$ indicates that segment i from chromosome k is enriched in the number of mapped tags from the ChIP-seq sample compare to input-DNA control and $\theta_{ki} = 0$ otherwise. We assume that

θ_k is distributed as a stationary Markov chain (1)

With transition probability $a_{kij} = P(\theta_{ks} = j | \theta_{k,s-1} = i)$, and that

θ_i and θ_j are independent for $i \neq j$, (2)

i.e., segments on different chromosomes are independent. In an HMM, the observed data are assumed to be conditionally independent given the hidden states:

$$p(z_k|\theta_k, F_k) = \prod_{i=1}^{m_k} p(z_{ki}|\theta_{ki}, F_k), \quad (3)$$

For $k = 1, \cdots, K$. Let $Z_{ki}|\theta_{ki} \sim (1-\theta_{ki})F_{k0} + \theta_{ki}F_{k1}$. Denote by $A_k = (a_{kij})$ the transition matrix, $\pi_k = (\pi_{k0}, \pi_{k1})$ the stationary distribution, $F_k = \{F_{k0}, F_{k1}\}$ the observation distribution, and $\Psi_k = (A_k, \pi_k, F_k)$ the collection of all HMM parameters. Let $z_k = (z_{k1}, \cdots, z_{kmk})$ be the observed z-values on chromosome k and $z = (z_1, \cdots, z_K)$. We assume that for a non-enriched segment, the z-values distribution is standard normal $F_{k0} = N(0,1)$, and for an enriched segment, the z-values distribution is a normal mixture $F_{k1} = \sum_{l=1}^{L_k} N(\mu_{kl}, \sigma_{kl}^2)$. The normal mixture model can approximate a large collection of distributions and has been widely used (Pan, 2003). When the number of components in the normal mixture L_k is known, the maximum likelihood estimate (MLE) of the HMM parameters can be obtained using the EM algorithm (Sun and Cai, 2009; Wei et al. 2009). When L_k is unknown, Bayesian information criterion (BIC) can be used to select an appropriate L_k.

2.2 Optimal FDR analysis of scoring ChIP-seq experiment results

Following Sun and Cai (2009), we define a local index of significance (LIS), that is, $LIS_i^k = P_\Psi(\theta_i = 0|z)$, which is the probability that a segment is a null (not enriched) given the observed data and where i is the ith segment on chromosome k and $\hat{\Psi}$ be an estimate of the HMM parameters. Denote by $LIS_{(1)}, \cdots, LIS_{(m_k)}$ the ordered LIS values and $H_{(1)}, \cdots, H_{(m_k)}$ the corresponding hypotheses. Therefore, we can define the LIS multiple testing procedure as follows:

Let $l^k = \max\left\{i : \frac{1}{i}\sum_{j=1}^{i} LIS_{(j)} \leq \alpha\right\}$. Then reject all $H_{(i)}$, for $i = 1, \cdots, l$. (4)

Sun and Cai (2009) have shown that under some regularity conditions, the *LIS* procedure is *optima*l in the sense that it controls the FDR at level α.

Although LIS is valid for FDR control, in practice it is desirable to combine the testing results from several chromosomes so that the *global (or genome-wise)* FDR is controlled at the nominal level. A more powerful approach is the pooled LIS procedure (PLIS). Following Wei et al. (2009), the PLIS procedure operates in three steps:

1. Calculate the plug-in LIS statistic $LIS_{ki} = P_{\Psi_k}(\theta_{ki} = 0 | z_k)$ for individual chromosomes.
2. Combine and rank the plug-in LIS statistic from all chromosomes, Denote by $LIS_{(1)}, \cdots, LIS_{(m)}$ the ordered values and $H_{(1)}, \cdots, H_{(m)}$ the corresponding hypotheses.
3. Reject all $H_{(i)}, i = 1, \cdots, l,$ where $l = \max\left\{ i : (1/i)\sum_{j=1}^{i} LIS_{(j)} \le \alpha \right\}$.

Wei et al. (2009) shows that PLIS is valid and asymptotically optimal. It is important to note that PLIS not only gives the optimal rankings of all hypotheses, but also suggests an optimal way of combining testing results from different chromosomes.

3. Results

3.1 Potential binding sites for STAT1

In this analysis, we focus on scoring results of STAT1 ChIP-seq data set generated by Rozowsky et al. (2009). Using PeakSeq procedure, they initially identified 123,321 potential binding sites for STAT1 (http://archive.gersteinlab.org/proj/PeakSeq/Scoring_ChIPSeq/Results/STAT1/STAT1_Targets/Extended/STAT1.txt). These are the potential targets that are found to be enriched in the STAT1 signal density maps compared to a simulated null random background. For each of these regions, Rozowsky et al. (2009) calculated a p-value from the cumulative distribution function for the binomial distribution to evaluate the enrichment of the number of mapped tags from the ChIP-seq sample compared to the normalized input-DNA control.

3.2 Transform p-values to z-values

Based on PLIS multiple testing theory, we need to transform the p-values obtained from a statistical test to z-values. Following McLachlan et al. (2006), we convert the p-values obtained from the cumulative distribution function for the binomial distribution as shown by Rozowsky et al. (2009) into z-values as follows: $z_i = \Phi^{-1}(1 - p_i)$, where p_i is the p-value for measuring the significance of enrichment of mapped tags in a given segment and Φ is the $N(0,1)$ distribution function. With this definition of z_i, departures from the null are indicated by large positive values of z_i. The distributions of z-values on different chromosomes are shown in **Figure 1**. For an enriched segment, we considered the z-values distribution in two cases: one is a single normal distribution and another is two normal mixtures.

3.3 Correcting for multiple testing under dependence

As a baseline, we apply Benjamini and Hochberg (BH) multiple testing procedure (Benjamini and Hochberg, 1995), which does not consider the dependence of mapped

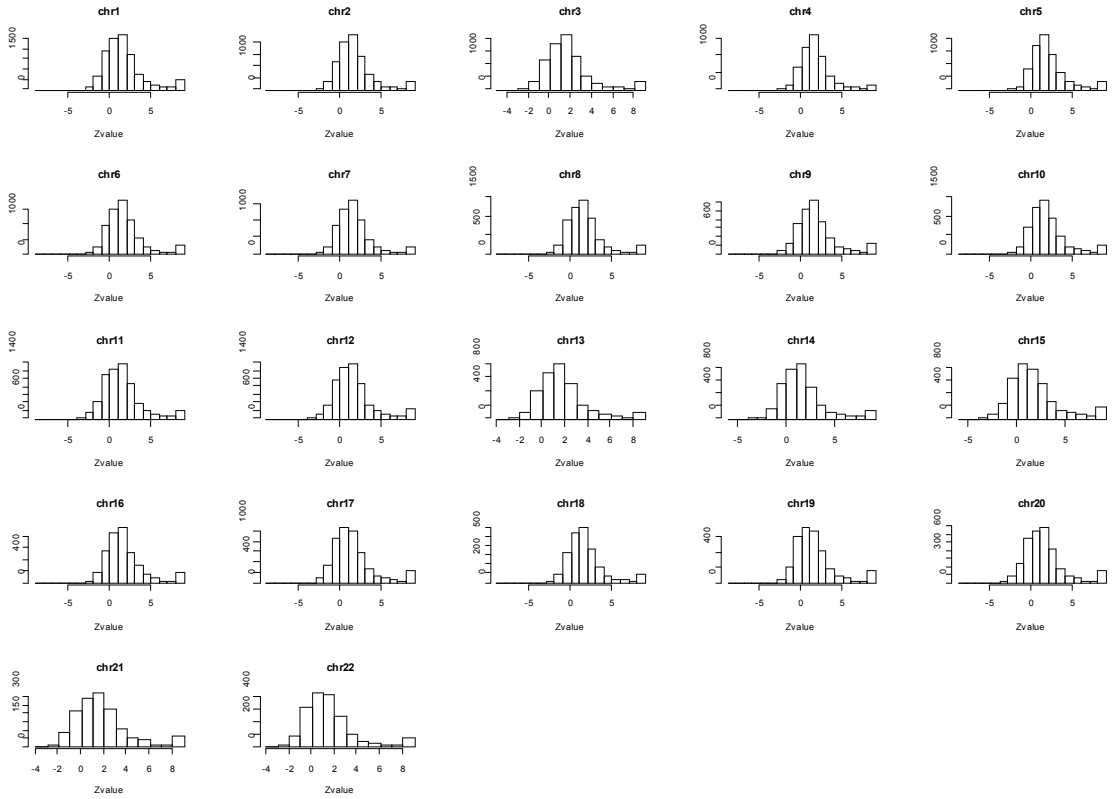


Figure 1. Distribution of z-values on different chromosomes

tags between neighboring regions, to adjust the p-values discussed in Section 3.1. The PLIS multiple testing procedure is applied to modeling the z-values discussed in Section 3.2. **Figure 2** demonstrates how the number of target binding sites varies for a range of different false-discovery rate thresholds for BH and PLIS multiple testing procedures. It is obvious that given the same FDR level, there are more significant segment sites identified by PLIS procedure than standard BH procedure. For example, given a false-discovery rate threshold 0.01, BH procedure identified 23,070 of these potential biding regions as significant region while PLIS procedure detected more than ~33,000 (L=1) of these biding segments as significant region.

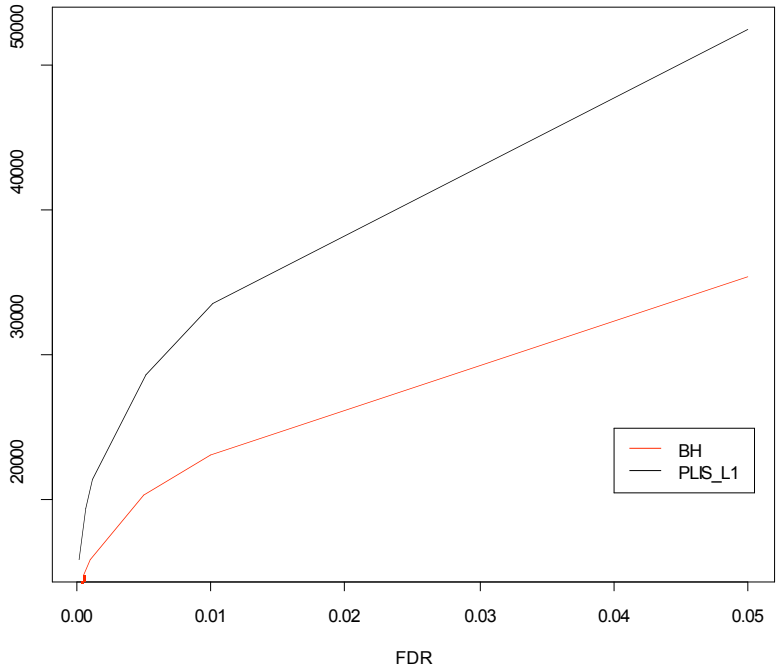


Figure 2 The number of significant binding sites varies for a range of different false-discovery rate thresholds for BH and PLIS multiple testing procedures

We further explore the rankings of these potential bidding sites based on the p-values adjusted by BH and PLIS multiple testing procedures. As shown in **Table 1**, for the sites ranked on the top (say top 2,000), the number of common regions identified by BH and PLIS procedures is not high (~36 - 46% for top 2,000 sites). However, as the number of top regions increases, say top 4,000, the number of common regions identified by two multiple testing procedures are quite high (large than 80%). This implies that multiple testing with dependence procedure has strong effect on the most significant target biding sides.

Number of Top Regions	Number of Common Regions (BH vs PLIS_L1*)	Number of Common Regions (BH vs PLIS_L2**)
1000	113	127
2000	727	927
4000	3368	3251
8000	7820	7608
16000	15207	13626
32000	28733	23854
64000	56584	50259

*L1- a signal normal distribution; ** two normal mixtures

Table 1 The number of common regions based on different number of top regions ranked by BH and PLIS procedures, respectively

We compare the results obtained for STAT1 ChIP-seq against the ChIP-Seq results obtained by Robertson et al. (2007) based on the BH and PLIS_L1 procedures to control FDR. Give a FDR level 0.05, we find that 21,340 and 25,632 STAT1 binding sites are present in the ChIP-Seq results obtained by Robertson et al. (2007), respectively.

4. Conclusion

In this study, we applied the large-scale multiple testing under dependence procedure, that is, pooled local index of significance (PLIS), developed by Wei et al. (2009) to scoring ChIP-seq experiments. The method uses HMM to model the dependence of mapped tags between neighboring segments. Essentially, PLIS is a "separate" analysis because, in the initial analysis stage of the method, the grouping information is exploited to calculate chromosome-wise HMM parameters; PLIS is also a "pooled" strategy because, in the last analysis stages of the approach, the group labels are dropped and the rankings of all hypotheses are determined globally. Similar idea was also explored by Efron (2004). The difference is that Efron suggests using identical FDR levels for all chromosomes, whereas here PLIS suggests using different FDR levels, which are automatically adapted to the features of all groups.

Our analysis show that the PLIS multiple testing procedure, compared to conventional BH multiple testing approach can improve identifying significant segments enriched with mapped tags. We are exploring whether the same conclusions can be made when the PLIS multiple testing framework is applied to the scoring results identified by other peak identification procedures, such as CisGenome (Ji et al., 2008). We are also comparing the results obtained for STAT1 ChIP-seq with other studies in detail.

Acknowledgement

We thank Dr. Sergio Pereira and Zhizhou Hu for their helpful discussions.

References

1. Benjamini, Y., and Hochberg, Y. (1995), Controlling the false discovery rate: a practical and powerful approach to multiple testing. *Journal of the Royal Statistical Society*, Series B, 57, 289–300.
2. Colella, S., Yau, C., Taylor, J., Mirza, G., Butler, H., Clouston, P., Bassett, A.S., Seller, A., Holmes, C.C. and Ragoussis, J. (2007), QuantiSNP: an Objective Bayes Hidden-Markov Model to detect and accurately map copy number variation using SNP genotyping data. *Nucleic Acids Research*, 35, 2013-2025.
3. Conneely, K.N., and Boehnke M. (2007), So Many Correlated Tests, So Little Time Rapid Adjustment of P Values for Multiple Correlated Tests, *Am J Hum Genet*, 81, 1158–1168.
4. Efron, B. (2004), Large-Scale Simultaneous Hypothesis Testing: the Choice of a Null Hypothesis, *Journal of the American Statistical Association,* 99, 96–104.
5. Efron, B. (2007), Correlation and Large-Scale Simultaneous Testing, *Journal of the American Statistical Association,* 102, 93–103.

6. Ji, H., et a. (2008), An integrated software system for analyzing ChIP-chip and ChIP-seq data. *Nature Biotechnology*, 26, 1293-1300.
7. McLachlan, G.J., Bean, R.W., and Ben-Tovim Jones, L. (2006), A simple implementation of a normal mixture approach to differential gene expression in multiclass microarrays, *Bioinformatics*, 22, 1608 – 1615.
8. Nyholt DR. (2004), A simple correction for multiple testing for single-nucleotide polymorphisms in linkage disequilibrium with each other, *Am J Hum Genet*, 74, 765–769.
9. Pan,W., Lin, J., and Le CT. (2003), A mixture model approach to detecting differentially expressed genes with microarray data, *Funct Integr Genomics*, 3, 117–24.
10. Qiu, X., Klebanov, L., and Yakovlev, A. (2005), Correlation Between Gene Expression Levels and Limitations of the Empirical Bayes Methodology for Finding Differentially Expressed Genes, *Statistical Applications in Genetics and Molecular Biology*, 4, Article 34.
11. Robertson, G. et al. (2007), Genome-wide profiles of STAT1 DNA association using chromatin immunoprecipitation and massively parallel sequencing. *Nature Methods*, 4, 651–657.
12. Rozowsky, J. et al. (2009), PeakSeq enables systematic scoring of ChIP-seq experiments relative to controls, *Nature Biotechnology*, 27, 66-75.
13. Sabatti, C., Service, S., and Freimer, N. (2003), False Discovery Rate in Linkage and Association Genome Screens for Complex Disorders, *Genetics*, 164, 829–833.
14. Sabatti, C. (2009), Genomewide association analysis of metabolic phenotypes in a birth cohort from a founder population, *Nature Genetics*, 41, 35–46.
15. Storey, J. (2002), A Direct Approach to False Discovery Rates, *Journal of the Royal Statistical Society ,* Sries B, 64, 479–498.
16. Storey, J., and Tibshirani, R. (2003), Statistical significance for genome-wide studies, *Proceedings of the National Academy of Sciences*, 100, 9440–9445
17. Sun, W., and Cai, T. (2009), Large-scale multiple testing under dependence, *Journal of the Royal Statistical Society*, Series B, 71, 393–424.
18. Tusher, V., Tibshirani, R., and Chu, G. (2001), Significance analysis of microarrays applied to the ionizing radiation response, *Proceedings of National Academy of Science,* USA 98, 5116–5121.
19. Wang K, Li M, Hadley D, Liu R, Glessner J, Grant S, Hakonarson H, Bucan M. (2007), PennCNV: an integrated hidden Markov model designed for high-resolution copy number variation detection in whole-genome SNP genotyping data, *Genome Research*, 17,1665-1674.
20. Wei, Z., Sun W., Wang, K., and Hakonarson, H. (2009), Multiple testing in genome-wide association studies via hidden Markov models, *Bioinformatics*, Advanced published August 4, 2009.

SeqMapReduce: software and web service for accelerating sequence mapping

Yanen Li[1] and Sheng Zhong[2,*]

[1]Department of Computer Science, University of Illinois at Urbana-Champaign, Urbana, Illinois 61801, USA

[2]Department of BioEngineering, University of Illinois at Urbana-Champaign, Urbana, Illinois 61801, USA

ABSTRACT

Next-generation sequencing technologies are increasing our ability to study genome function. A new and rapidly growing family of assays for measuring the genome-wide profiles of mRNAs, small RNAs, transcription-factor binding, chromatin structure and DNA methylation status are now being implemented by applying the massively parallel, ultrahigh-throughput DNA sequencing systems. These rapid growths demand reliable, fast and easy-to-use analysis tools. We present the SeqMapReduce software for parallelizing sequence mapping using the cloud computing technology. The speed is quasi-linear to the number of computing nodes available. It took 4.5 minutes to map 6 million sequence reads to the human genome with 32 computing nodes. A comparison between SeqMapReduce and CloudBurst demonstrated that SeqMapReduce was 57.9 times faster than CloudBurst on average. We also present a user-friendly web server for unsophisticated users. The SeqMapReduce software and web service are available at http://www.seqmapreduce.org.

1 INTRODUCTION

Nearly 80 gigabytes of sequence data come out for each run of a typical high-throughput sequencer available at the current stage, and the technology is improving rapidly. Mapping these sequences onto a genome, e.g., the human genome, takes up to several days for an upscale modern computer. Bioinformatic analysis often requires multiple rounds of mapping, using different parameters, such as the maximum number of mismatches allowed, in order to determine the best mapping result. Such efforts would take weeks to accomplish, and usually cannot be done in a typical biology lab that generates the sequencing data.

A number of algorithms have been developed for efficient sequence mapping, including ELAND (Cox, unpublished), , RMAP (Smith *et al*., 2008), SeqMap (Jiang *et al*., 2008), SOAP (Li *et al*., 2008), ZOOM (Lin *et al*., 2008) and others. ELAND, RMAP, SeqMap and SOAP shared the idea of first identifying the exact matching seeds and then extending to the full sequences. While ELAND, RMAP, and SeqMap index the sequence reads, SOAP indexes the target genome, and thus requiring much larger memory. RMAP can map reads with or without error probability information (quality scores) and supports paired-end reads or bisulfite-treated reads mapping. All these algorithms were designed to run on a single computer, and took dozens of hours to map hundred gigabytes of sequence data.

A key feature for mapping high-throughput sequences is that the same task is repeated for a huge amount of times. This feature makes parallel computing a tempting option for speeding up the computation. We developed a sequence mapping algorithm, SeqMapReduce, using the cloud computing technology. The cloud computing technology was chosen because users can easily purchase inexpensive computing time from cloud computing solution providers such as Google, Amazon, Microsoft, and IBM, and therefore an ordinary user can use SeqMapReduce with her laptop. We also provide a user-friendly web server for those who do not even want to install any software at www.seqmapreduce.org .

SeqMapReduce utilized the Apache Hadoop software framework. Hadoop supports data intensive distributed applications under a free license. It enables applications to work with thousands of nodes and petabytes of data. Hadoop is an open-source implementation of the MapReduce (Dean et al., 2004) programming model. The basic idea of MapReduce is to break the computational tasks into a Map phase that generates intermediate key/value pairs and a Reduce phase that merges the intermediate values associated with the same intermediate key.

CloudBurst (Schatz, 2009) is another program for mapping sequences using a MapReduce strategy. The major differences between SeqMapReduce and CloudBurst are as follows. First, SeqMapReduce provides a web application, which eliminates the hassles of software installation or even hardware upgrade. Second, SeqMapReduce implemented a in memory seed-and-extension and late emission strategy, which tremendously reduced the amount of intermediate results stored on the cluster and transmitted between the computing functions, resulting in 26 to 97 folds of speed increases on the datasets being tested.

*To whom correspondence should be addressed.

2 METHODS

SeqMapReduce is implemented on the MapReduce programming model. Under the MapReduce framework, SeqMapReduce implemented a map function (mapper) and a reduce function (reducer) [Figure 1]. The workflow of MapReduce is as follows.

Pre-processing: formatting the genome

The target genome is segmented into segments of the same length. Each segment is recorded with a tag/seq pair. The tag records the chromosomal location of the segment, which also serves as an identifier of the segment. The seq is the actual DNA sequence. After pre-processing, all segments are written into a file which will be automatically partitioned and loaded to all computing nodes of a Hadoop cluster. Once a genome is formatted, it can be re-used for all mapping tasks on this cluster without formatting again.

Map phase: mapping sequences using seed-and-extension

In the map phase, the formatted genome segments are split into roughly equal size subsets and each subset is sent to a mapper function. In each Mapper, a hash table is built for the sequence reads and scan for seed match on the segments of the reference genome. Every sequence read is stored in the hash table as a key/value pair. We utilized the pigenhole principle to find qualified seed matches. A read is divided into *n* seeds, where n depends on the maximum number of mismatched allowed. If this number equals 2, then *n* equal to 4. In this case, a qualified seed match is found if at least 2 out of 4 parts of the reads are exactly matched with the genome segment. Once a qualified seed match is found, the read and the genome segment will be scanned for extended matches. If the extended matches satisfy a pre-specified sensitivity threshold, this match result will be transferred to the reducers using a key/tmp_res pair. Here the key is the same key as the read, and the tmp_res records the temporary results including the read and the genome segment . The seed-and-extension approach greatly reduced the amount of intermediate results stored and transmitted in a Hadoop cluster. Because indexing the temporary results is heavily I/O-bound, reducing the amount of I/O boosted the performance.

Reduce phase: aggregating intermediate results to output

The Reducer receives all the key/tem_res pairs emitted from the Mappers, and organizes the final results as key/final_res pairs. The key is the identifier of the read. The final_res records all matching genome segments. The Reducer outputs the final key/final_res pairs ordered by the values of keys.

3 RESULTS

3.1 The web server.

We implemented SeqMapReduce as a web application (www.seqmapreduce.org). Users can access this web application using any web browsers. The computing engine of this web application is the Illinois Cloud Computing Testbed (CCT) cluster with 108 computing nodes. Each node is equipped with eight 64 bit 2.6 GHz CPUs, 16 GB RAM, and 2 TB storage.

We evaluated performance of SeqMapReduce with different numbers of computing nodes and using several datasets of different sizes.

First, we mapped two sets of human Chip-Seq reads from the CAMDA 2009 Contest Datasets (Rozowsky *et al*., 2009) (4.5 million reads and 6.2 million reads) to the human genome (3.5 Gbp). Two mismatches were allowed for each sequence match. The running time decreased quasi-linearly as the number of computing nodes increased [Figure 2]. With all the 32 nodes, the SeqMapReduced took 276 seconds to map 6.2 million reads to the human genome . The hash table design within each mapper enabled the runtime to be sub-linear to the number of sequence reads . For example, the average runtime of mapping 6.2 million reads was 1.03 fold of that of mapping 4.5 million reads, although there was 1.4 fold difference in the data sizes. We totally tested 16 CAMDA data sets. The detailed mapped results can be found at www.seqmapreduce.org/wiki/index.php/CAMDA2009_results .

We also compared the running time of SeqMapReduce and CloudBust with a human Illumina/Solexa dataset from the 1000 Genomes Project (accession SRR001113). We carried out the test with 24 computing nodes, using only one CPU on each node. Four sets of sequence reads were used; each set contained 1, 2, 4, and 8 million reads. Up to 2 mismatches were allowed. SeqMapReduce exhibited an average of 58.9 fold of speed acceleration in these tests (Table 1). Importantly, the runtime ratio of the two algorithms increased as the size of the input dataset increased, in a consistent manner. Considering a typical experiment may generate hundreds of millions of reads, we estimate the speed difference between these two algorithms to be in the range of thousand to ten thousand folds in a real data analysis.

Table 1. Runtime comparison of SeqMapReduce and CloudBurst

|Software|	1 M Reads	2 M Reads	4M Reads	8 M Reads
CloudBurst	5137 s	10529 s	22118 s	42639 s
SeqMapReduce	195 s	248 s	322 s	437 s
Runtime Ratio	26.4	42.5	68.8	97.6

1, 2, 4, and 8 million Solexa reads were mapped to the human genome using 24 computing nodes.

3.2 Running SeqMapReduce on the Amazon EC2 system.

Amazon Elastic Compute Cloud (EC2) provides an easy-to-use and cost effective resource for cloud computing. Users can purchase the computing time as needed. We tested SeqMapReduce on two mouse Chip-Seq datasets (2 millions and 6 millions reads respectively). EC2 provides two options of computing time, i.e., the "Large Standard Instances" and the "High-CPU Instances", with the former one being less expensive. We used "Large Standard Instances"for our tests (Table 2). The runtimes on EC2 were longer than those on the CCT cluster. This is likely due to the better hardware of the CCT cluster. Again, SeqMapReduce achieved quasi-linear speed increment as the number of computing nodes increased (Table 2). For example, SeqMapReduce gained 8.67 fold of speed increment on 32 computing nodes as compared to 4 nodes. It cost us $99.01 to finish these tests. The quasi-linear speed acceleration of SeqMapReduce enables about N fold of speed gain with *N* computing nodes. This property reduces the computing time from days to hours or minutes with several dozen nodes.

Table 2. Runtime of SeqMapReduce on Amazon EC2

|Reads|	4 nodes	8 nodes	16 nodes	32 nodes
2 Million	10056 s	4309 s	2242 s	1160 s
6 Million	17132 s	7719 s	3835 s	1976 s

2 and 6 millions of mouse Chip-Seq reads were mapped to the mouse genome (2.6 Gbp) using Amazon EC2. Up to 2 mismatches were allowed.

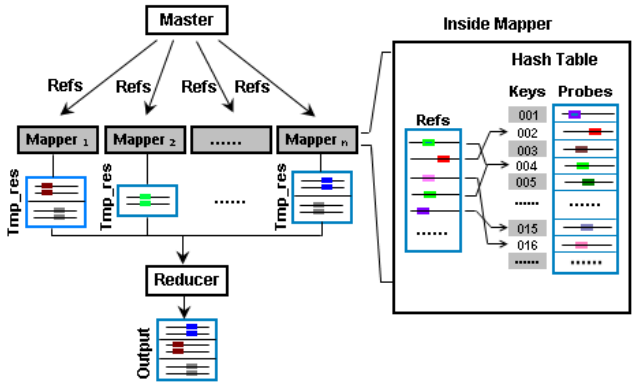


Fig. 1. The SeqMapReduce Program Framework.

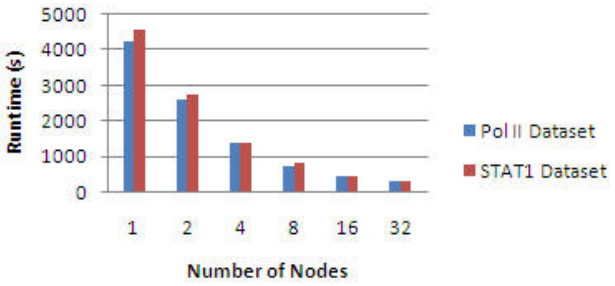


Fig. 2. Running time of SeqMapReduce on the CAMDA 2009 datasets in the CCT cluster. Two data sets were tested, one is from Pol II ChIP-seq FC201WVA_20080307_s_5 with 4.5 million reads, the other is from IFNg stimulated STAT1 ChIP-seq FC302MA_20080507_s_1 with 6.2 million reads.

ACKNOWLEDGEMENTS

We thank Michael Schatz for providing the CloudBurst software for testing. This work is supported by NSF DBI 08-45823 (SZ).

REFERENCES

ELAND: Efficient Local Alignment of Nucleotide Data.

Li H, Ruan J, Durbin R (2008) Mapping short DNA sequencing reads and calling variants using mapping quality scores. Genome Research 18, 1851-1858.

Smith A.D. et al. (2008) Using quality scores and longer reads improves accuracy of Solexa read mapping. BMC Bioinformatics, 9:128.

Li R et al. (2008) SOAP: short oligonucleotide alignment program. Bioinformatics 24 (5), 713-714.

Lin H et al. (2008) ZOOM! Zillions of oligos mapped. Bioinformatics 24 (21), 2431-2437.

Michael C. Schatz. (2009) CloudBurst: highly sensitive read mapping with MapReduce. Bioinformatics 25 (11), 1363¨C1369. 2009

H. Jiang and W. H. Wong, (2008) Seqmap : mapping massive amount of oligonucleotides to the genome," Bioinformatics, pp. btn429+.

J. Dean and S. Ghemawat. (2004) MapReduce: Simplified data processing on large clusters. In OSDI ¡¯04.

J. Rozowsky et al. (2009) PeakSeq enables systematic scoring of ChIP-seq experiments relative to controls. Nature Biotechnology 27, 66 – 75.

APPENDIX

1 COMPARATIVE ANALYSIS OF SEQMAPREDUCE AND CLOUDBURST

How to reduce the amount of data communicating between mappers and reducers is a ctritical issue of the MapReduce framework. SeqMapReduce builds a Hash Table of all Reads in every Mapper, scans against the splitted Genome and emmits qualified aligned results to the Reducers, which simply collect and sort the final output. Within each Mapper, SeqMapReuce utilizes the Pigeonhole Principle to filter out large portion of unqualified alignments. For example, if at most 2 mismatches are allowed, SeqMapReduce divides the Read into 4 parts, only the Genome sequences that exactly match 2 out of 4 parts needed to be extended in the Mapper. A large amount of random matches are filtered out, and the data load in the Mapper-Reducer transmission is significantly reduced.

On the other hand, CloudBurst emmits short keys both from reads set and the reference Genome, sorts the keys by the Hash functionality provided by the Hadoop system itself, and does extension in the Reducers on reads and reference sequences with shared key. The CloudBurst algorithm is not fit for short sequence mapping in the MapReduce framework due to the following flaws. The major limitation of CloudBurst is that it generates too many exact matches by chance. For example, with 7 bp seed length (read length=36bp, the number of allowed mis-matches k=4), the expected number of occuerance of a seed by chance in the Human Genome only (2.87 Gb) ≈ 175,000. Even if the maximum mis-matches k=2, this number ≈ 171 (seed length=12). Since CloudBurst emmits seeds both from reference Genome as well as Reads, if there are 7 M Reads, the number of random occurance from Reads is 427 (k=4) and 0.4 (k=2). Therefore, about 175,427 and 171.4 more un-necessary (*seed*, *MerInfo*) pairs are emitted per seed, which should have been filtered in the Mappers. Due to the big number of seeds, a huge amount of intermediate (*seed*, *MerInfo*) pairs are tranfered from Mappers to Reducers, slowing down the system significantly due to the heavy I/O loads. Secondly, in the Reduce phrase, because all potential aligned reads and ference sequences are shared by the same key, it needs to do alignments for all pairs of read-reference within the Reducer. The large number of random occurance of the seed as mentioned above makes this join computation formidable. Thirdly, For every seed, the positions in the reference or read, left flanking and right flanking sequences need to carried with the seed and transmitted to the Reducers; this considerable amount of data could be reduced sinificantly if indexing and processing in the memory in a Mapper before qualified results are emmited to the Reducers.

Table S1. Features of SeqMapReduce and CloudBurst

Program	Algorithm	Web Service	Mapper Size	Data Transfer[1]	Disk Space Consumed	Hash Table Design	Genome Reusaable[2]	Indels	Paired End Read	Read Quality Information[3]
SeqMapReduc	Pigeonhole Principle	Yes	Big	Small	Small	Built in the Mapper	Yes	Allowed	No	No
CloudBurst	Seed and Extend	No	Small	Big	Large	Hadoop System Hash Table	Yes	Allowed	No	No

1. The data transferred between Mappers and Reducers.
2. Genome only needs to be formatted once and put to the Hadoop Distributed File System; and it is reuserable for subsequent tasks.
3. Unlike RMAP, CloudBurst doesn't use the read quality information. SeqMapReduce doesn't use quality information in this version.

2 THE SEQMAPREDUCE WEB SERVICE

A easy-to-use SeqMapReduce Web Service is also open to the public. The screenshot of the service (www.seqmapreduc.org) is shown in Fig. S1.

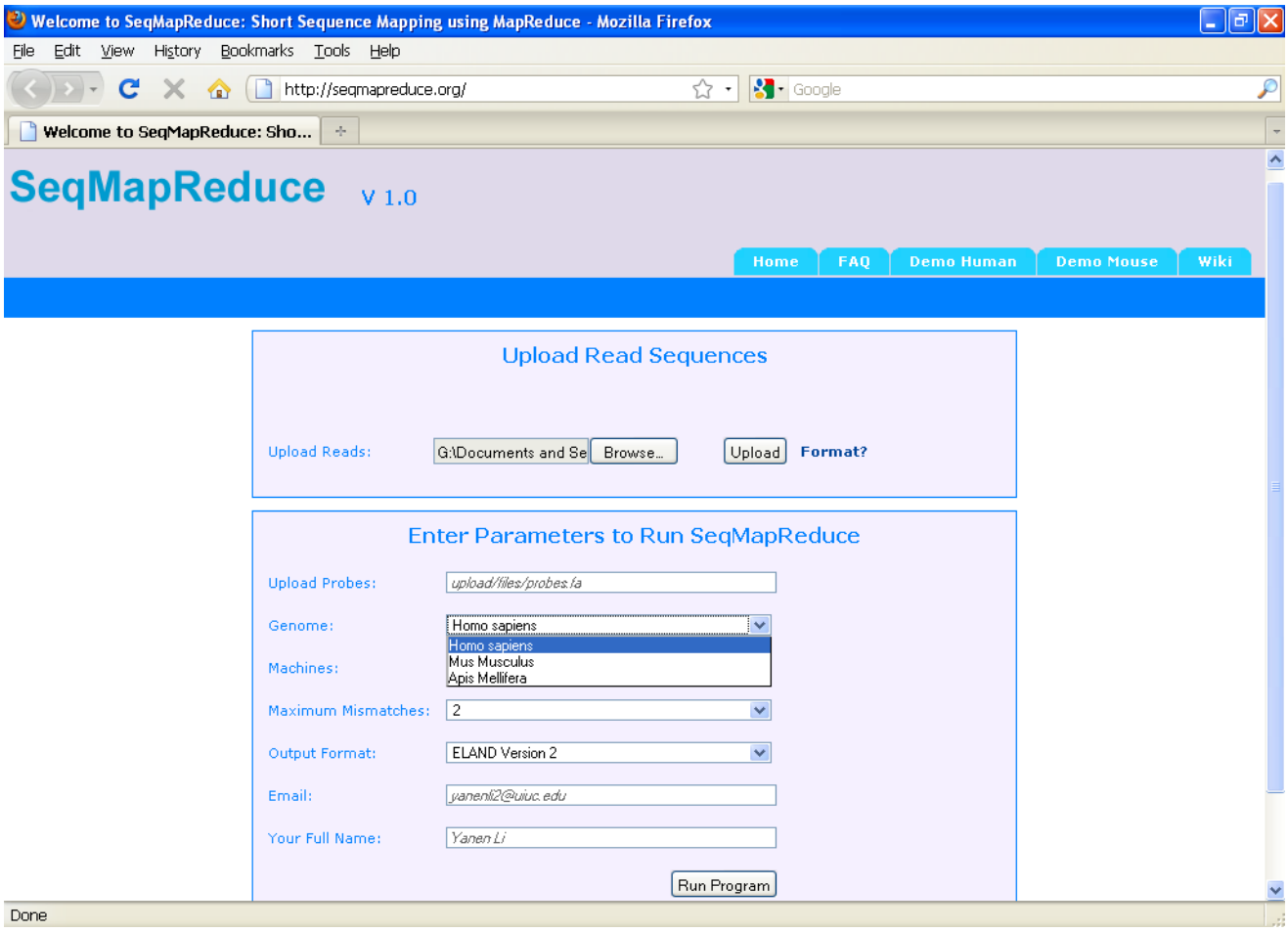


Fig. S1. Screenshot of the SeqMapReduce Web Service. Read Sequences can be uploaded to the server with the zip format. Several Genomes are supported in the SeqMapReduce Web Service. ELAND and other output formats are supported. Results of several millions of Reads usually can be returned to the users email in a few minutes.

STAT1 regulates microRNA transcription in interferon γ – stimulated HeLa cells

Guohua Wang [1], Yadong Wang [1,6], Denan Zhang [1], Lang Li [2,3], Yunlong Liu [2,3,4,5,6]

[1] School of Computer Science and Technology, Harbin Institute of Technology, Harbin, Heilongjiang 150001, PR China,
[2] Division of Biostatistics Department of Medicine, [3] Center for Computational Biology and Bioinformatics, [4] Center for Medical Genomics, [5]Department of Medical and Molecular Genetics, Indiana University School of Medicine, Indianapolis, IN 46202, USA
[6] Correspondence should be addressed to: Y.W. ydwang@hit.edu.cn and Y.L. yunliu@iupui.edu

Extended Abstract

MicroRNAs are small non-coding RNAs known to regulate the target transcripts by promoting mRNA degradation and suppressing translation [1,2]. To date, several hundred precursor microRNAs (pre-microRNAs) and mature microRNAs have been annotated in several mammalian genomes [3]. Despite of such development, the genomic landscape of most primary microRNAs (pri-microRNAs), however, has not been fully annotated. Without knowing the transcription start site (TSS) and promoter regions of their primary forms (pri-microRNA), it is difficult to identify the transcription factors and their binding sites that regulate the microRNA transcription, and therefore hinders the understanding of microRNA-mediated regulatory network.

Previously, using RNA polymerase II (Pol II) ChIP-seq data, we developed a computational approach to indentify promoter region and TSS of pri-microRNAs, based on the observation that Pol II binding sites are enriched around the promoter regions of the expressed genes [4]. This model used the biological knowledge that most microRNAs (not all of them) are transcribed by RNA Polymerase II [5]. The overall procedure is to first model Pol II binding pattern near the TSS of well annotated protein-coding genes that are highly expressed, and then to search similar patterns in the upstream region of annotated mature or pre-microRNA.

In this study, we indentified promoter regions of intergenic microRNAs in HeLa cells using the Pol II ChIP-seq data provided by the CAMDA 2009 challenging dataset [6]. Highly expressed genes were selected based upon microarray experiment using Affymetrix platform (GEO number: GSE3051 [7]). Following the similar strategy as we did previously [4], we focused only on that genes not overlapping or close to other genes (length greater than 10,000-bp and with no other genes present within 10,000-bp of its TSS). This results in 4,120 expressed genes and 2,682 unexpressed genes in HeLa cells, based on the absent and present calls in the Affymetrix MAS5 algorithm. To evaluate the predictive power of the provided Pol II ChIP-seq data and our model to identify active promoters in HeLa cells, we randomly selected 1/4 of expressed genes to train our model. The remaining genes, both expressed and non-expressed, were used as test sets. The area under the curve (AUC) in the recursive operative

curve (ROC) reached 0.86 in differentiating all the expressed genes in the test set and unexpressed genes (Figure 1), suggesting excellent predictive power of our strategy. We further divided the expressed gene into three categories based on their expression levels. The result (Figure 1) clear demonstrates that the prediction accuracy of our model is higher for the genes that are highly expressed.

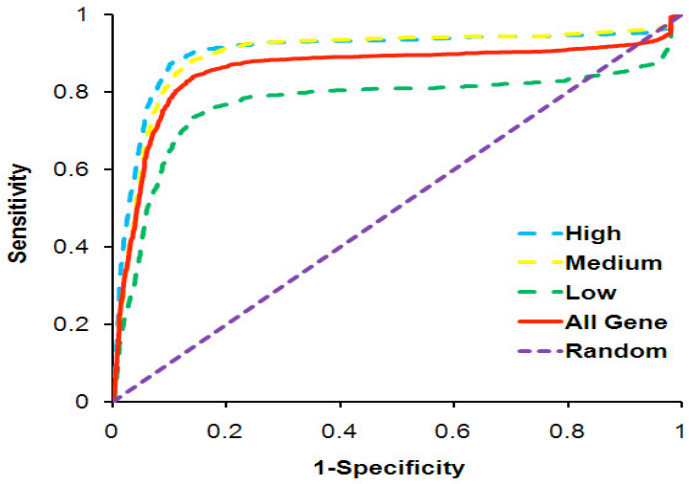


Figure 1. ROC curve for TSS prediction of protein coding genes. The expressed genes were separated into three categories, high (light blue), low (green), and medium expressed gene (yellow). The three categories expressed genes and non-expressed genes are considered positive and negative sets, respectively. One fourth of the genes are used as training data, while the remaining as test set. The ROC curve was generated using ROCR library in R project (http://www.r-project.org).

We obtained annotations of 685 human mature or pre-microRNAs from the miRBase microRNA sequence database (version 11.0, [3]). Among them, 419 microRNAs that locate between protein-coding genes (or intergenic microRNAs) were used for promoter identification. Using the model parameters estimated based on Pol II binding patterns around the transcription start sites of protein coding genes, we indentified 83 active microRNA promoters in HeLa cell (with false discovery rate ≤ 0.2). The median value of the length of regulatory region was 1,476-bp, with longest and shortest widths of 4,989-bp and 397-bp, respectively (Figure 2A). The distances between the identified TSS and their corresponding mature or pre-microRNA also differ in a great deal, ranging from 200 to 10,000-bp, with median distance around 3600-bp (Figure 2B).

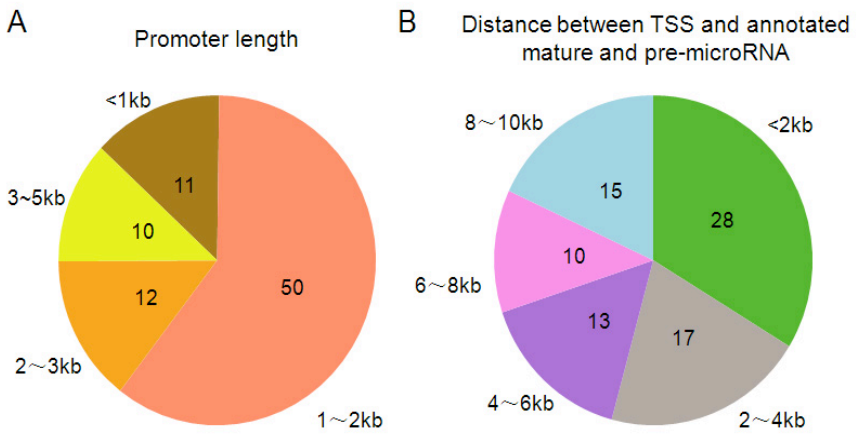


Figure 2. Statistics of predicted microRNA promoters. Pie diagram shows the numbers of microRNAs with different ranges of (A) promoter lengths and (B) distances between their predicted transcription start sites and annotated mature and pre-microRNAs.

We further examined the sequence features of identified promoter regions, including their conservation levels across evolution and their relationship with annotated CpG islands. We observed high GC content within or around the predicted regulatory regions. Among the 83 predicted microRNA promoters, 66 promoters (79.5%) were found to either contain or overlap with annotated CpG island [8]. In addition, the identified promoter region and transcription start site also demonstrated higher conservation (PhastCons scores based on 17 species, including mammalian, amphibian, bird, and fish [8]) comparing to randomly selected regions (red dash line in Figure 3).

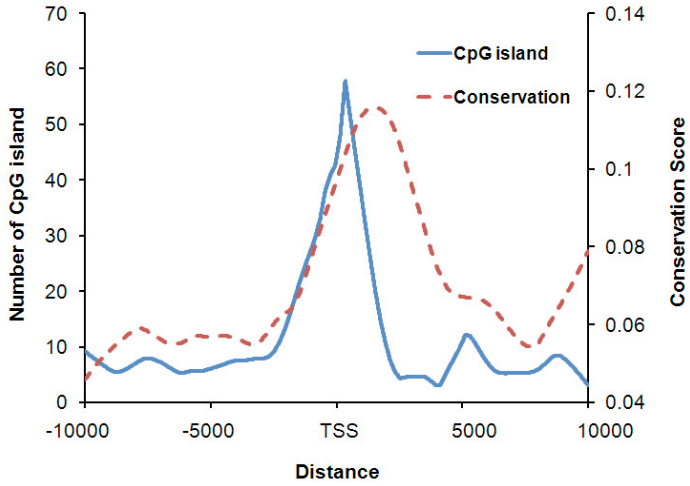


Figure 3. Sequence features around predicted microRNA promoter. CpG islands and conservation score were retrieved from the UCSC genome browser, where CpG islands were defined as genomic regions of the length greater than 200 bp, with a minimal GC content of 50%, and the ratio of observed /expected CpG greater than 0.6. The conservation scores were calculated based on a phylogenetic hidden Markov model that measures the evolutionary conservation in 17 vertebrates [8].

We searched the STAT1 binding sites identified by Gerstein's group [6] in 83 predicted microRNA promoters. Among these, promoter regions of 41 microRNAs (49.4%, Supplement table S1) contain or overlap with STAT1 enriched regions. These represent the microRNAs that are potentially regulated by STAT1 in HeLa cells in response to interferon γ stimulation. Most promoters contain one binding sites, while the promoters of hsa-mir-21 and hsa-mir-92b have two STAT1 target sites, and a microRNA cluster, hsa-mir-193b and hsa-mir-365-1, has three target sites. We further compared the density of STAT1 target sites related to the distance from transcription start sites, for both protein coding genes, whose

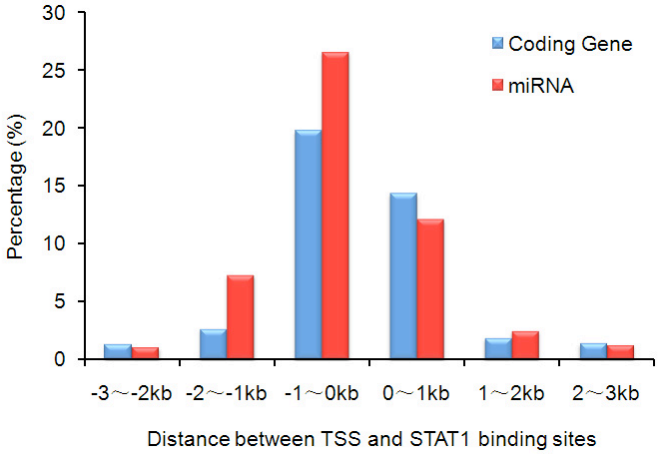


Figure 4. Percentage of genes containing STAT1 binding sites within every 1KB region surrounding transcription start sites. The calculation is based on 36,998 STAT1 binding sites identified in the PeakSeq algorithm with FDR ≤ 0.05 [6] and their relative locations with 4,120 expressed coding genes and 83 predicted microRNAs.

annotations are well established, and microRNAs. We counted the number of STAT1 binding sites in every 1,000-bp interval from 3,000-bp upstream to 3,000-bp downstream related to TSS of 4,120 expressed coding genes and for the 83 microRNAs predicted to be actively transcribed. The percentage of genes contains STAT1 targets in each 1,000-bp interval were calculated (Figure 4). We observed significant enrichment of STAT1 binding sites within -1,000 bp to +1,000 bp of the transcription start site, for both protein coding genes (34%) and microRNAs (38%).

High throughput DNA sequencing offers new opportunities for genomic research. In the current study, we identified the promoter regions of 83 microRNAs using Pol II ChIP-seq data in HeLa cell. The identified regions correlate with annotated CpG islands, and are highly conserved across multiple species. In the predicted microRNA promoters, many include STAT1 binding sites while HeLa cells were stimulated by interferon γ; these microRNAs were potentially responding to interferon stimulation through activation of STAT1.

Acknowledgement

This work is supported by the U.S. National Institutes of Health grants AA017941 (Y.L.), CA113001 (L.L. and Y.L.), China 863 High-Tech Program 2007AA02Z302 (Y.L.), and China Natural Science Foundation 60901075 (G.W.).

Reference

1. Ambros V (2001) microRNAs: tiny regulators with great potential. Cell 107: 823-826.
2. Kim VN (2005) MicroRNA biogenesis: coordinated cropping and dicing. Nat Rev Mol Cell Biol 6: 376-385.
3. Griffiths-Jones S, Grocock RJ, van Dongen S, Bateman A, Enright AJ (2006) miRBase: microRNA sequences, targets and gene nomenclature. Nucleic Acids Res 34: D140-144.
4. Wang G, L. L, Shen C, Wang Y, Huang Y, et al. (submitted) RNA Polymerase II binding patterns reveal genomic regions involved in microRNA gene regulation.
5. Bartel DP (2004) MicroRNAs: genomics, biogenesis, mechanism, and function. Cell 116: 281-297.
6. Rozowsky J, Euskirchen G, Auerbach RK, Zhang ZD, Gibson T, et al. (2009) PeakSeq enables systematic scoring of ChIP-seq experiments relative to controls. Nat Biotechnol 27: 66-75.
7. Mense SM, Sengupta A, Zhou M, Lan C, Bentsman G, et al. (2006) Gene expression profiling reveals the profound upregulation of hypoxia-responsive genes in primary human astrocytes. Physiol Genomics 25: 435-449.
8. Karolchik D, Kuhn RM, Baertsch R, Barber GP, Clawson H, et al. (2008) The UCSC Genome Browser Database: 2008 update. Nucleic Acids Res 36: D773-779.

Supplementary table 1. List of 83 microRNAs and their predicted transcription start sites and promoter regions.

microRNA	Chromosome	Strand	microRNA position	Predicted TSS	Predicted promoter region	CpG island	STAT1 binding sites
hsa-mir-1259	chr20	+	47330254-47330364	47328454	47327958-47329130	1	1
hsa-mir-21	chr17	+	55273409-55273480	55270009	55268897-55273231	0	2
hsa-mir-24-2	chr19	-	13808101-13808173	13814573	13811422-13815167	1	1
hsa-mir-23a	chr19	-	13808401-13808473	13814673	13811722-13815033	1	1
hsa-mir-27a	chr19	-	13808254-13808331	13814731	13811756-13814979	1	1
hsa-mir-92b	chr1	+	153431592-153431687	153430192	153429022-153431598	2	2
hsa-mir-1304	chr11	-	93106488-93106578	93114378	93113614-93114931	1	1
hsa-let-7i	chr12	+	61283733-61283816	61283333	61281958-61283985	1	1
hsa-mir-760	chr1	+	94084976-94085055	94084576	94083843-94086408	2	1
hsa-mir-940	chr16	+	2261749-2261842	2257949	2257515-2258728	1	1
hsa-mir-320a	chr8	-	22158420-22158501	22158701	22157984-22159040	1	0
hsa-mir-219-1	chr6	+	33283590-33283699	33280390	33280273-33280811	1	0
hsa-mir-1289-1	chr20	-	33505190-33505333	33506533	33505758-33507220	1	1
hsa-mir-196b	chr7	-	27175624-27175707	27176107	27175306-27176348	1	0
hsa-mir-632	chr17	+	27701241-27701334	27701041	27700771-27701814	1	0
hsa-mir-196a-2	chr12	+	52671789-52671898	52664189	52662019-52667008	1	1
hsa-mir-141	chr12	+	6943521-6943615	6941121	6940221-6941928	0	0
hsa-mir-200c	chr12	+	6943123-6943190	6941123	6940217-6941937	0	0
hsa-mir-639	chr19	+	14501355-14501452	14501155	14501094-14501895	1	0
hsa-let-7a-1	chr9	+	95978060-95978139	95968260	95967698-95969675	1	1
hsa-let-7f-1	chr9	+	95978450-95978536	95968450	95967669-95969581	1	1
hsa-mir-20a	chr13	+	90801320-90801390	90797920	90797290-90799576	1	1
hsa-mir-19b-1	chr13	+	90801447-90801533	90797847	90797338-90799565	1	1
hsa-mir-92a-1	chr13	+	90801569-90801646	90797969	90797241-90799589	1	1
hsa-mir-18a	chr13	+	90801006-90801076	90797806	90797321-90799595	1	1
hsa-mir-19a	chr13	+	90801146-90801227	90797946	90797253-90799584	1	1
hsa-mir-17	chr13	+	90800860-90800943	90797860	90797325-90799550	1	1
hsa-mir-484	chr16	+	15644652-15644730	15644452	15643650-15645182	0	1
hsa-mir-374a	chrX	-	73423846-73423917	73428917	73428175-73429954	1	1
hsa-mir-545	chrX	-	73423664-73423769	73428969	73428120-73429962	1	1
hsa-mir-564	chr3	+	44878384-44878477	44878184	44877826-44878917	1	0
hsa-mir-220c	chr19	-	53755341-53755423	53763823	53763323-53764214	0	1
hsa-mir-1282	chr15	-	41873149-41873249	41880249	41879515-41880834	1	0
hsa-mir-1281	chr22	+	39818463-39818516	39817263	39816769-39818930	1	1
hsa-mir-607	chr10	-	98578416-98578511	98582111	98581328-98583103	1	1
hsa-mir-659	chr22	-	36573631-36573727	36575327	36575046-36575917	1	0
hsa-mir-658	chr22	-	36570225-36570324	36575324	36575041-36575930	1	0
hsa-mir-450b	chrX	-	133501881-133501958	133511358	133510499-133511725	1	0
hsa-mir-1285-2	chr2	-	70333554-70333641	70338641	70338362-70339227	1	1
hsa-mir-503	chrX	-	133508024-133508094	133511294	133510543-133511729	1	0
hsa-mir-542	chrX	-	133503037-133503133	133511333	133510533-133511729	1	0
hsa-mir-450a-2	chrX	-	133502204-133502303	133511303	133510505-133511735	1	0
hsa-mir-424	chrX	-	133508310-133508407	133511407	133510454-133511752	1	0
hsa-mir-450a-1	chrX	-	133502037-133502127	133511327	133510508-133511706	1	0
hsa-mir-96	chr7	-	129201768-129201845	129207645	129206528-129208221	1	1
hsa-mir-183	chr7	-	129201981-129202090	129207690	129206551-129208188	1	1
hsa-mir-550-2	chr7	+	32739118-32739214	32734318	32733922-32735381	1	1
hsa-mir-146b	chr10	+	104186259-104186331	104182259	104181579-104182655	1	0
hsa-mir-101-1	chr1	-	65296705-65296779	65306379	65304635-65307079	2	0
hsa-mir-202	chr10	-	134911006-134911115	134921115	134920762-134921221	0	0
hsa-mir-182	chr7	-	129197459-129197568	129207568	129206461-129208322	1	1
hsa-mir-662	chr16	+	760184-760278	750584	750191-751986	0	0
hsa-mir-200b	chr1	+	1092347-1092441	1082747	1082554-1083767	1	0
hsa-mir-193a	chr17	+	26911128-26911215	26910328	26909297-26911052	1	1
hsa-mir-210	chr11	-	558089-558198	566598	565841-566918	1	1
hsa-mir-505	chrX	-	138833973-138834056	138842856	138842113-138844047	1	1
hsa-mir-1303	chr5	+	154045529-154045614	154040729	154040224-154042138	0	1
hsa-mir-886	chr5	-	135444076-135444196	135444396	135443930-135444899	1	1
hsa-mir-150	chr19	-	54695854-54695937	54696137	54694991-54696467	0	0
hsa-mir-200a	chr1	+	1093106-1093195	1083106	1082373-1083797	1	0
hsa-mir-34a	chr1	-	9134314-9134423	9141423	9141119-9141516	0	0
hsa-mir-365-1	chr16	+	14310643-14310729	14303643	14301804-14305383	1	3

hsa-mir-301b	chr22	+	20337270-20337347	20336470	20335816-20337176	1	0
hsa-mir-193b	chr16	+	14305325-14305407	14303525	14302053-14305321	1	3
hsa-mir-132	chr17	-	1899952-1900052	1900452	1899182-1900821	1	0
hsa-mir-212	chr17	-	1900315-1900424	1900624	1899191-1900820	1	0
hsa-mir-320b-1	chr1	+	117015894-117015972	117011694	117011245-117012545	1	0
hsa-mir-130b	chr22	+	20337593-20337674	20336393	20335901-20337204	1	0
hsa-mir-613	chr12	+	12808850-12808944	12803050	12802708-12804273	0	1
hsa-mir-1302-2	chr15	-	100318185-100318322	100319322	100319002-100319427	1	0
hsa-mir-548h-2	chr16	-	11307798-11307885	11314485	11313572-11314726	0	1
hsa-mir-181d	chr19	+	13846689-13846825	13844689	13844317-13845428	1	0
hsa-mir-181c	chr19	+	13846513-13846622	13844713	13844291-13845408	1	0
hsa-mir-125a	chr19	+	56888319-56888404	56883319	56882480-56885676	0	0
hsa-mir-1826	chr16	+	33873009-33873093	33870209	33869987-33871815	1	0
hsa-mir-345	chr14	+	99843949-99844046	99841549	99840687-99843457	1	1
hsa-mir-99b	chr19	+	56887677-56887746	56883277	56882520-56885670	0	0
hsa-let-7e	chr19	+	56887851-56887929	56883251	56882540-56885677	0	0
hsa-mir-1302-3	chr2	-	114057006-114057143	114058343	114057544-114059007	1	0
hsa-mir-135b	chr1	-	203684053-203684149	203685749	203684904-203685968	0	0
hsa-mir-1253	chr17	-	2598122-2598226	2605426	2605062-2605910	1	0
hsa-mir-1247	chr14	-	101096377-101096512	101097712	101096659-101097939	1	0
hsa-mir-1224	chr3	+	185441887-185441971	185439487	185439267-185440572	0	0

Cloud-based Services for Large Scale Analysis of Sequence and Expression Data: Lessons Learned from Cistrack

David Hanley[1], Yunhong Gu[1], Xiangjun Liu[1,2], Nick Bild[3,4], Nicolas Negre[3,4], Parantu K Shah[3,4], Feng Tian[1,2], Jia Chen[1], Michal Sabala[1], Kevin P White[3,4], Robert L Grossman[1,3]*

[1]National Center for Data Mining, University of Illinois at Chicago, MC 249, 851 South Morgan Street, Chicago, IL 60607-7045
[2]School of Medicine, Tsinghua University, Beijing, China 100084
[3]Institute for Genomics & Systems Biology, The University of Chicago, Cummings Life Sciences Center 431A, 920 East 58th Street, Chicago, IL 60637
[4]Department of Human Genetics and Department of Ecology and Evolution, Cummings Life Sciences Center 5th Floor, 920 East 58th Street, Chicago, IL 60637

September 10, 2009

Introduction

Next generation sequencing, microarrays and other functional genomics technologies are changing the way in which biological and biomedical research is carried out by providing genome-wide data on various cellular phenomena. However, archiving, managing, and analyzing the large datasets produced can be challenging using the current generation of database-based technologies. In this article, we described the design and implementation of the Cistrack system, which integrates cloud-based computing platforms with databases. The Cistrack bioinformatics system (https://www.cistrack.org) is currently used to support large scale genomics project such as modENCODE (www.modencode.org). It provides a simple, lightweight and integrated solution for archiving the raw data, processing it, and sharing the results with the community.

Cistrack consists of the following four major components:

1. A database for managing genomics, expression and related data.
2. Data analysis pipelines for large-scale studies of *cis*-regulatory elements.
3. A storage cloud for archiving data and a compute cloud to support the pipeline analysis of data as well as data reanalysis.

4. A web portal containing Web 2.0 widgets for browsing, downloading and analyzing Cistrack data.

See Figure 1. Components 1, 2 and 3 are relevant to this paper and are described in more detail below.

Until recently, management of biological data has been done primarily using databases. There are three challenges with this approach today. First, today's high throughput sequencing machines, such as the Illumina Genome Analyzers, are producing 1 TB or more of data per run and this is simply too much data to be managed with today's relational databases. Sequencing datasets will grow even larger in the future. Second, there are so many different databases today that integrating data across them is more and more of a challenge. Third, as the size of data grows larger and as the number of datasets that need to be integrated increases, databases are not always the best platform for managing data to support computation, especially high performance computing.

Cistrack differs from related projects, such as CisRED (Robertson et. al., 2006) and CEAS (Ji et. al., 2006), in that Cistrack provides complete management and analysis capabilities for massive, raw genome-wide data.

Cloud Computing Platforms

Cistrack not only uses databases to manage data, but also integrates cloud computing services. Cistrack's unique integration with cloud computing services has helped to address each of the problems described above that databases have when working with very large biological datasets.

Although there is not standard definition of a cloud today, a good working definition for the purposes here is to define a cloud as a computing platform consisting of racks of commodity computers that provide on-demand resources and services over a network, usually the Internet, with the scale and the reliability of a data center (Grossman 2009).

There are at least two different, but related, types of clouds: the first type of clouds provide computing instances on demand, while the second type of clouds provide computing capacity on demand. Amazon's EC2 services (aws.amazon.com) are an example of the first category. The Eucalyptus system (Nurmi *et. al.*, 2009) is an open source system that provides on-demand computing instances and shares the same APIs as Amazon's EC2 cloud.

The Google File System (GFS) is a distributed file system that is designed to support extremely large datasets, datasets so large that they cannot be easily managed by databases. Google's MapReduce is an application that supports a simple parallel programming framework over data managed by GFS. GFS/MapReduce is an example of the second type of cloud (Dean *et. al.*, 2004). Hadoop is an open source implementation of GFS/MapReduce (hadoop.apache.org). Sector is another open source cloud that is based upon a scalable distributed file system (called the Sector Distributed File System) (Gu, *et. al.*, 2009). Sector is integrated with an application called Sphere that supports another simple parallel programming framework – arbitrary User Defined Functions (UDFs) that can be invoked over the data managed by the SDFS. Cistrack manages data in part using the SDFS; the next release of Cistrack will also use Sphere UDFs to support high performance data pipelines.

Cistrack Components

CistrackDB. The first component of Cistrack is a database called CistrackDB. Not all Cistrack data is stored in CistrackDB. Some data is left as files with the metadata managed by the database. This is a standard architecture used by many systems. Unlike most other systems though, these files may be managed by a standard file system or by a cloud storage service. The collection of archived files is called the Cistrack Archive.

Cistrack includes utilities for uploading data, including collections of files, and for creating and annotating experiments. The annotation page allows users to group the files together into experimental units and to annotate the units with metadata. Upon successful annotation of the metadata, the raw data files are copied into the CisTrack Archive, and relevant information is

moved from a staging space to the CistrackDB.

The tables in CistrackDB can be divided into 5 groups. The first group contains 7 tables for storing gene structure information. The second group has 8 tables to support antibody design, preparation and validation. The experiment group contains 24 tables and supports the design, storage and analysis of array and Solexa experiments. The fourth group contains 5 tables for information on various annotations of the data. The sixth group has 6 tables to manage users, organizations, and data access controls.

Pipelines. Cistrack contains a number of *cis*-regulatory data analysis pipelines for large-scale studies of *cis*-regulatory elements. A user visits the CisTrack website to design a new experiment by designating test and control samples, selects the appropriate pipeline, and enters any required parameters. Currently MAT (Johnson, *et al*., 2006) and HMMSeg (Day *et. al.* 2007) for Affymetrix, CisGenome (Ji *et. al*. 2008), HMMSeq and MA2C (Song *et. al*. 2007) for Agilent, Bustard & Gerald (from Illumina) for extraction of Solexa raw images to call bases, and MACS (Zhang *et. al*., 2008) for Solexa eland files are supported. CisTrack generates the final analysis results, including .wig and .bed files, which can be viewed by third party tools such as the Affymetrix Integrated Genome Browser.

Storage and compute cloud services. The Cistrack Archive uses the Sector Distributed File System to manage large data files. Sector has been used to manage hundreds of TB of data and scales by adding additional commodity computers to the system. Cloud storage systems with similar designs have scaled to thousands of nodes and PB of data (Dean *et. al*, 2004). The current version of Sector has been tested on over 100 nodes and over 100 TB of data. Sector is currently being tested on a 250 node testbed containing approximately 1 PB of raw disk.

The Solexa pipeline in Cistrack is currently being ported to Sphere so that cloud compute services can be used to improve the performance of the current pipeline, which uses an 8 way SMP system for parallelism. The October release of Cistrack is expected to include this capability.

Sphere is also used for the reanalysis of data managed by Cistrack. By reanalysis we mean the automated processing of specified datasets using specified pipelines. This is often desirable if an improved version of a program used by the pipeline is available. Reanalysis is not frequent today due to the complexity of capturing all the parameters required when automating analysis and to the computational resources required when reanalyzing large numbers of large datasets. Cistrack solves the first problem by storing the relevant parameters in CistrackDB and the second problem by using Sector/Sphere.

Status

CisTrack currently contains 337 experiments, 859 array/sequencing experimental units and 2198 data files. CisTrackDB currently contains more than 3 TB of genome-wide ChIP-chip and ChIP-seq data from *Drosophila*, mouse and human. There are over 50 TB of files in the Cistrack Archive. Cistrack has just completed adding support for the Solexa pipeline and the amount of data is expected to increase by approximately 8 TB per week. A decision has not yet been made about the length of time that raw Solexa data files will be kept.

References

Day, N.; Hemmaplardh, A.; Thurman, R.E.; Stamatoyannopoulos, J.A.; Noble, W.S. (2007) Unsupervised segmentation of continuous genomic data. *Bioinformatics*, 23:1424-1426.

Dean, J. and Ghemawat, S. (2004) MapReduce: simplified data processing on large clusters. In *Proceedings of the 6th Conference on Symposium on Opearting Systems Design & Implementation - Volume 6* (San Francisco, CA, December 06 - 08, 2004). Operating Systems Design and Implementation. USENIX Association, Berkeley, CA, 10-10.

Grossman, Robert L. (2009) The Case for Cloud Computing, IT Professional, volume 11, number 2, pages 23-27.

Gu, Y.; Grossman, R. L. (2009) Sector and Sphere: Towards Simplified Storage and Processing of Large Scale Distributed Data, Philosophical Transactions of the Royal Society A, Volume 367, Number 1897, pages 2429-2445.

Ji, H.; Jiang, H.; Ma, W.; Johnson, D.S.; Myers, R.M.; Wong, W.H. (2008) An integrated system CisGenome for analyzing ChIP-chip and ChIP-seq data. *Nat Biotechnol.*, 26: 1293–1300.

Ji, X.; Li, W.; Song, J.; Wei, L.; Liu, X.S. (2006) CEAS: cis-regulatory element annotation system. *Nucleic Acids Research*, 34(Web Server issue):W551-W554

Johnson, W.E.; Li, W.; Meyer, C.A.; Gottardo, R.; Carroll, J.S.; Brown, M.; Liu, X.S. (2006) Model-based analysis of tiling-arrays for ChIP-chip. *Proc Natl Acad Sci USA*, 103, 12457-12462.

Nurmi, D.; Wolski, R.; Grzegorczyk, C.; Obertelli, G.; Soman, S.; Youseff, L.; Zagorodnov, D. (2009) Eucalyptus: an open-source cloud computing infrastructure. *J. Phys.: Conf. Ser.* 180:012051

Robertson, G.; Bilenky, M.; Lin, K.; He, A.; Yuen, W.; Dagpinar, M.; Varhol, R.; Teague, K.; Griffith, O.L.; Zhang, X.; Pan, Y.; Hassel, M.; Sleumer, M.C.; Pan, W.; Pleasance, E.D.; Chuang, M.; Hao, H.; Li, Y.Y.; Robertson, N.; Fjell, C.; Li, B.; Montgomery, S.B.; Astakhova, T.; Zhou1, J.; Sander1, J.; Siddiqui, A.S.; Jones, S.J.M. (2006) cisRED: a database system for genome-scale computational discovery of regulatory elements . *Nucleic Acids Research* ,34(Database Issue):D68-D73

Song, J.S.; Johnson, W.E; Zhu, X.; Zhang, X.; Li, W.; Manrai, A.K.; Liu, J.S.; Chen, R.; Liu, X.S. (2007) Model-based analysis of two-color arrays (MA2C). *Genome Biology*, 8:R178

Tian, F.; Shah, P.K.; Liu, X.; Negre, N.; Chen, J.; Karpenko, O.; White, K.P.; Grossman, R.L. (2009) Flynet: a genomic resource for Drosophila melanogaster transcriptional regulatory networks. *Bioinformatics*. 2009 Aug 5.

Zhang, Y.; Liu, T.; Meyer, C.A.; Eeckhoute, J.; Johnson, D.S.; Bernstein, B.E.; Nussbaum, C.; Myers, R.M.; Brown, M.; Li, W.; Liu, X.S. (2008) *Genome Biol*. 9:R137

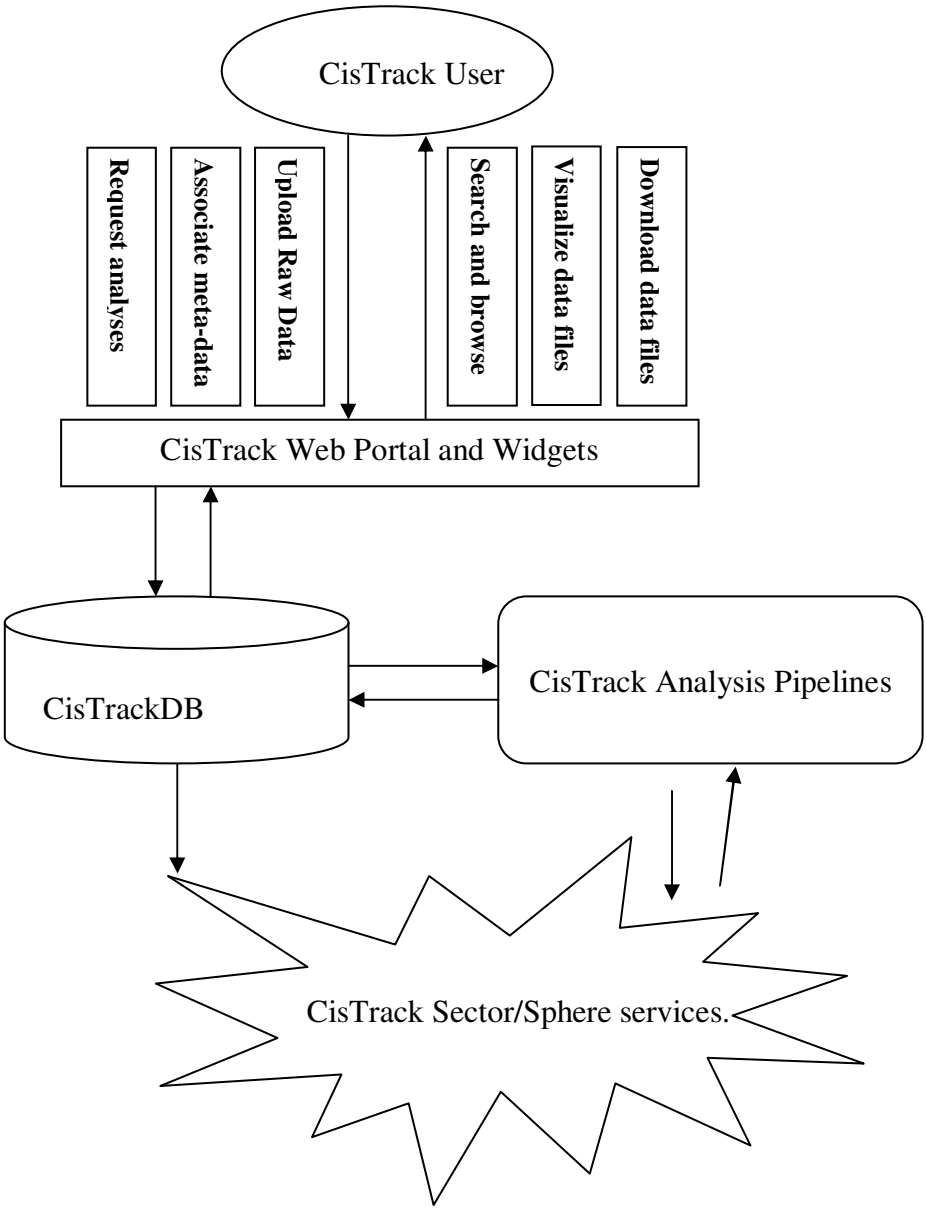


Figure 1 :CisTrack architecture

Transcriptome analysis methods for RNA-Seq data

Colin N. Dewey [1,2,*] and Bo Li [2]
[1]Department of Biostatistics and Medical Informatics, University of Wisconsin-Madison
[2]Department of Computer Sciences, University of Wisconsin-Madison,
{cdewey,bli}@cs.wisc.edu

September 10, 2009

Abstract

Next generation sequencing is enabling new high-precision methods for measuring gene and isoform expression levels. In particular, a technique called RNA-Seq, which produces tens of millions of short reads from across a transcriptome, is rapidly gaining popularity. RNA-Seq allows for a number of inferences about a transcriptome, including expression estimates and novel gene and splice site discovery. In this review, we survey the computational methods that have been introduced for analyzing RNA-Seq data, with a focus on those methods that infer expression levels or gene models. We conclude with a summary of the future challenges in this area.

Introduction

RNA-Seq is for transcriptomes what whole-genome shotgun sequencing is for genomes. While in genome sequencing we are primarily interested in the sequence of an entire genome, in RNA-Seq we are concerned with both the sequences of transcripts and their copy number in a transcriptome. The RNA-Seq protocol consists of three main steps: (1) conversion of mRNA into cDNA fragments, (2) sequencing of the fragments with a high-throughput sequencer, and (3) computational analysis of the sequencer reads for transcriptome characterization [32].

There are a number of inferences that one can make about a transcriptome given RNA-Seq data. First, given a reference genome or transcript set, one can estimate expression levels of genes and even individual isoforms for alternatively-spliced genes. When a reference transcript set is incomplete but a genome sequence is available, RNA-Seq reads can be used for discovering novel genes, exons, and splice junctions. When neither a genome sequence nor a transcript set is available, RNA-Seq can theoretically be used to determine a transcriptome using sequence assembly techniques. Finally, it can be used to determine gene variants, such as SNPs and indels [3].

In this review, we will describe a number of computational methods that have been recently developed for the analysis of RNA-Seq data. We begin with the basics of how RNA-Seq reads can be used to estimate relative expression levels. We then survey methods that have been introduced for this task. In addition to methods that estimate expression levels, we will also describe techniques used for the detection of novel genes, exons, and exon junctions. To conclude, we summarize the computational challenges that remain for analyzing RNA-Seq data sets.

Transcriptome measurements

The estimation of expression levels from RNA-Seq data relies on the assumption that the number of reads derived from an isoform is a function of its expression level. In the ideal case, reads are uniformly distributed across the transcriptome and thus

*to whom correspondence should be addressed

the probability of a read coming from isoform i is proportional to ν_i, where ν_i is the *fraction of nucleotides* in the transcriptome made up by isoform i. Here, for simplicity, we are assuming that reads can start at any position along a transcript, which can be the case when all mRNAs have poly(A) tails. Given these assumptions, a maximum likelihood estimator for ν_i is c_i/N, where c_i is the number of reads derived from isoform i and N is the total number of reads.

For comparisons of expression between isoforms, usually one wants to know the *fraction of transcripts* made up by each isoform. Letting τ_i denote the fraction of transcripts made up by isoform i, we can easily compute this value from ν_i using the relation

$$\tau_i = \frac{\nu_i}{\ell_i} \left(\sum_j \frac{\nu_j}{\ell_j} \right)^{-1}, \quad (1)$$

where ℓ_i is the length, in nucleotides, of isoform i. Thus, given counts of reads derived from each isoform and their lengths, one can estimate expression in terms of either ν or τ.

One of the initial studies using RNA-Seq introduced the notion of *reads per kilobase per million mapped reads* (RPKM), for specifying expression levels [22]. The measured level of isoform i, in RPKM, is defined as $10^9 \times c_i/(N_m \ell_i)$, where N_m is the total number of mappable reads. From Equation 1, we see that the RPKM value for an isoform is proportional to the maximum likelihood estimator for τ. Unfortunately, the normalization factor for converting RPKM to a fraction of transcripts depends on the lengths and relative expression levels of all isoforms. Thus, to make expression levels comparable across experiments, we suggest simply using the fraction of transcripts measure (τ) for specifying expression levels.

Expression estimation methods

The major computational steps for expression analysis from RNA-Seq data are (1) base-calling from raw sequencer output (e.g., fluorescence intensities), (2) mapping of reads to reference sequences, and (3) estimating expression levels from the read mappings. A number of methods have been developed for more accurate base-calling from next generation sequencing technologies, in addition to those provided by the manufacturers [9, 12, 14, 24, 33]. These methods are typically highly technology-dependent, and thus we will not describe them in more detail in this review. Mapping reads involves the rapid alignment of short read sequences against large reference genomes or known transcript sets, allowing for a small number of mismatches or indels. Many alignment tools have been developed for this task (e.g., [16, 19, 18, 25, 27]), which are reviewed in [30]. In this section we review available methods for computing gene or isoform expression levels, given alignments of the reads to a reference. The primary differences between current expression estimation methods are in how they handle reads that map to multiple locations (*multireads*) and whether they estimate expression for individual isoforms, or only for entire gene loci.

The simplest method for estimating expression levels from RNA-Seq data is to keep only those reads that map uniquely to a single gene or location along the genome. Reads that do not map (given a maximum number of mismatches or indels), or that map to multiple genes are discarded. The expression level (in terms of fraction nucleotides) for gene i is then calculated as $\nu_i^{unique} = c_i^{unique}/c^{unique}$, where c_i^{unique} is the number of reads uniquely mapping to gene i and c^{unique} is the total number of uniquely mapping reads. To get expression measured in terms of fraction of transcripts, these values can be converted via Equation 1, where some effective length of the gene must be calculated (e.g., the length of the longest isoform, or the sum of the lengths of the exonic intervals belonging to the gene). This method for calculating gene expression has been used by a number of the initial RNA-Seq studies [23, 20].

The shortcoming of using only uniquely-mapping reads for expression estimation is that it is inaccurate for genes containing relatively-repetitive regions, which give rise to multireads. Gene repetitiveness may be due to either low complexity segments or recent gene duplication. In addition, when estimates of expression for individual isoforms are desired, few reads map to a single isoform, as alternate splice forms often share a significant amount of sequence.

The remaining methods that we describe attempt to correct for these complications.

A more sophisticated method that uses only uniquely-mapping reads attempts to correct for gene repetitiveness by computing the "mappability" of each exon [21]. Ignoring end effects, the mappability of an exon sequence, σ, is $\frac{1}{|\sigma|-L+1}\sum_{i=1}^{|\sigma|-L+1} unique(\sigma_i^{i+L-1})$, where L is the read length and $unique(\sigma_i^j)$ is 1 if the substring of σ from position i to j is unique in the genome, and 0 otherwise. In words, the mappability is essentially the fraction of reads potentially derived from an exon that map uniquely. Given pre-computed mappability values, the read count for an exon is adjusted by dividing by the exon's mappability.

A second class of methods that takes into account gene repetitiveness does are called "rescue" schemes. These methods do not discard multireads and instead attempt to allocate fractions of them to the positions from which they may have been derived. One such method was introduced in [22] and implemented in the ERANGE software package. ERANGE divides the count of a multiread amongst the genes to which it maps in proportion to the transcript fractions, τ^{uni} that it first estimates from uniquely-mapping reads. The count for each gene is calculated as:

$$c_i^{erange} = \sum_{n \,:\, i \in \pi_n} \frac{\tau_i^{uni}}{\sum_{j \in \pi_n} \tau_j^{uni}}$$

where n is an index over reads and π_n is the set of gene indices to which read n maps. When calculating gene expression levels, ERANGE takes the effective length of a gene to be the total length of the union of all exonic intervals belonging to the gene.

A second rescue scheme was initially introduced for CAGE data [10] and later used for RNA-Seq data [5, 11, 4]. Instead of taking into account all unique reads that map to a gene, this scheme allocates a multiread according to the count of unique reads mapping within a fixed-length window around each possible mapping for the multiread. For RNA-Seq a typical window size is 200bp [11]. A memory and time-efficient implementation of this method has been developed, which is shown to compare favorably with ERANGE [11]. An advantage of the method over ERANGE is that it does not rely on having accurate gene models. However, by only using the counts of uniquely-mapping reads within a fixed-width window, it is not using as much information as it could when accurate gene models are available.

We have recently introduced a maximum likelihood (ML) method for gene expression estimation that can be interpreted as a statistically rigorous formulation of the rescue schemes [17]. This method uses a generative probabilistic model for RNA-Seq reads and uses an Expectation-Maximization (EM) algorithm to find its ML parameters, which correspond to isoform expression levels. With this formulation, it can be seen that the rescue method used by ERANGE is roughly equivalent to one iteration of the EM algorithm. The primary advantages of this method are that it gives more accurate gene expression estimates and can produce estimates for individual isoforms. Its disadvantages include its reliance on accurate gene and isoform models and longer computations.

The method of [13] also estimates individual isoform expression levels with a statistical model. Unlike the multinomial model of [17], this method approximates the RNA-Seq read generation process by a set of independent Poisson processes, one process per exonic interval. Maximum likelihood isoform expression levels are estimated via coordinate-wise hill climbing and confidence intervals are established via importance sampling. Like [17], this method also requires an accurate set of gene models.

Novel transcript detection methods

In addition to providing information about expression levels, RNA-Seq data can be used to detect novel genes, exons, and splice junctions. A common strategy for novel transcript detection has emerged from a number of groups [4, 20, 22, 28]. First, reads are mapped against known exons and splice junctions. Any reads that do not map to known transcript sequences are aligned against the entire genome. Genomic "islands" of aligned reads in unannotated regions are candidates for novel exons or genes. Reads

that remain unmappable are possibly novel splice junctions. To identify the locations of these junctions, the unmapped reads can be optionally clustered/assembled and then aligned to the genome using an "intron-aware" alignment tool (e.g., [15]).

Two methods have been recently introduced for more efficiently mapping reads that span novel splice junctions and do not require a set of known gene annotations [7, 29]. The first, QPALMA [7], uses the `vmatch` aligner [1] to map reads against a reference genome. Halves of unmappable reads are then realigned to the genome to find seeds for potential splice junctions. A modified, splice-site-aware, Smith–Waterman algorithm is run on 2kb windows around seed alignments to identify the most likely splice junction identified by a read. Parameters for the Smith–Waterman alignment are determined using a large margin algorithm [26].

A second method, TopHat [29], first identifies "islands" of mapped reads (aligned using the Bowtie aligner [16]) along the genome as putative exons. Unmappable reads are then stored in a k-mer indexed table. All possible canonical donor and acceptor sites between nearby islands are identified and the sequences spanning these possible junctions are searched against the unmappable read table. Junctions that match a significant number of reads are then reported. The authors of TopHat tout its speed in comparison to QPALMA, as TopHat can process an entire mammalian RNA-Seq data set in less than a day on a single workstation.

Future challenges

While current computational methods for RNA-Seq are enabling researchers to probe transcriptomes in more detail than ever before, there remain a number of challenges that need to be addressed until the technology can reach its full potential. First, biases in RNA-Seq data sets need to be fully explored and corrected for by inference methods. Factors that lead to the violation of the assumption that reads are uniformly distributed across a transcriptome are of particular concern, as this assumption is basis for most expression estimation methods. For example, depending on the exact protocol used, reads may be biased towards the 5' or 3' ends of transcripts [32]. Reads may also be biased towards certain transcriptome segments due to base composition (e.g., GC content) [8]. Biases such as these will need to be characterized and taken into account while estimating expression.

A second challenge will be to determine all possible alternative-splicing events undergone by each gene in a genome and estimate the frequencies of these events and the isoforms that result. Although we have methods for estimating isoform expression levels given known gene models [13, 17] and for identifying novel exons and splice junctions [29, 7], these methods will need to be combined to fully characterize alternative splicing within a transcriptome. In addition, given that the number of possible isoforms for a gene may be exponential in the number of its exons, it is not clear how much can be inferred from current RNA-Seq data set sizes.

A third, and likely more difficult, challenge will be the use of RNA-Seq on species for which we do not have a reference genome or transcript set. Most current methods rely on reference sequence sets to do any sort of analysis on RNA-Seq data. Promising work in the direction of reference-free RNA-Seq has been the transcriptome sequencing of the Glanville fritillary butterfly with 454 reads [31] and de novo assembly of Illumina 36bp human RNA-Seq reads [2]. When a closely-related reference genome is available, a comparative approach to RNA-Seq analysis may also be feasible [6].

References

[1] M. Abouelhoda, S. Kurtz, and E. Ohlebusch. The enhanced suffix array and its applications to genome analysis. *Lecture Notes in Computer Science*, pages 449–463, 2002.

[2] I. Birol, S. D. Jackman, C. Nielsen, J. Q. Qian, R. Varhol, G. Stazyk, R. D. Morin, Y. Zhao, M. Hirst, J. E. Schein, D. E. Horsman, J. M. Connors, R. D. Gascoyne, M. A. Marra, and S. J. Jones. De novo transcriptome assembly

with ABySS. *Bioinformatics*, pages btp367–, 6 2009.

[3] I. Chepelev, G. Wei, Q. Tang, and K. Zhao. Detection of single nucleotide variations in expressed exons of the human genome using rna-seq. *Nucleic Acids Research*, pages gkp507–, 2009.

[4] N. Cloonan, A. R. R. Forrest, G. Kolle, B. B. A. Gardiner, G. J. Faulkner, M. K. Brown, D. F. Taylor, A. L. Steptoe, S. Wani, G. Bethel, A. J. Robertson, A. C. Perkins, S. J. Bruce, C. C. Lee, S. S. Ranade, H. E. Peckham, J. M. Manning, K. J. McKernan, and S. M. Grimmond. Stem cell transcriptome profiling via massive-scale mRNA sequencing. *Nat Meth*, 5(7):613–619, 2008.

[5] N. Cloonan, Q. Xu, G. J. Faulkner, D. F. Taylor, D. T. P. Tang, G. Kolle, and S. M. Grimmond. RNA-MATE: A recursive mapping strategy for high-throughput RNA-sequencing data. *Bioinformatics*, pages btp459–, 2009.

[6] L. J. Collins, P. J. Biggs, C. Voelckel, and S. Joly. An approach to transcriptome analysis of non-model organisms using short-read sequences. *Genome Informatics*, 21:3–14, 2008.

[7] F. De Bona, S. Ossowski, K. Schneeberger, and G. Ratsch. Optimal spliced alignments of short sequence reads. *Bioinformatics*, 24(16):i174–180, 2008.

[8] J. C. Dohm, C. Lottaz, T. Borodina, and H. Himmelbauer. Substantial biases in ultra-short read data sets from high-throughput DNA sequencing. *Nucleic Acids Research*, 36(16):e105, 2008.

[9] Y. Erlich, P. Mitra, et al. Alta-Cyclic: a self-optimizing base caller for next-generation sequencing. *Nature methods*, 5(8):679–682, 2008.

[10] G. J. Faulkner, A. R. R. Forrest, A. M. Chalk, K. Schroder, Y. Hayashizaki, P. Carninci, D. A. Hume, and S. M. Grimmond. A rescue strategy for multimapping short sequence tags refines surveys of transcriptional activity by CAGE. *Genomics*, 91(3):281–288, 2008.

[11] T. Hashimoto, M. J. L. de Hoon, S. M. Grimmond, C. O. Daub, Y. Hayashizaki, and G. J. Faulkner. Probabilistic resolution of multi-mapping reads in massively parallel sequencing data using MuMRescueLite. *Bioinformatics*, pages btp438–, 2009.

[12] R. Irizarry and H. Bravo. Model-based quality assessment and base-calling for second-generation sequencing data. *Johns Hopkins University, Dept. of Biostatistics Working Papers*, 2009.

[13] H. Jiang and W. H. Wong. Statistical inferences for isoform expression in RNA-Seq. *Bioinformatics*, 25(8):1026–1032, 2009.

[14] W.-C. Kao, K. Stevens, and Y. S. Song. Bayescall: A model-based base-calling algorithm for high-throughput short-read sequencing. *Genome Research*, 2009.

[15] W. J. Kent. BLAT–the BLAST-like alignment tool. *Genome Research*, 12(4):656–664, 2002.

[16] B. Langmead, C. Trapnell, M. Pop, and S. Salzberg. Ultrafast and memory-efficient alignment of short DNA sequences to the human genome. *Genome Biology*, 10(3):R25, 2009.

[17] B. Li, V. Ruotti, R. Stewart, J. Thomson, and C. Dewey. RNA-Seq gene expression estimation with read mapping uncertainty. Submitted., 2009.

[18] H. Li, J. Ruan, and R. Durbin. Mapping short DNA sequencing reads and calling variants using mapping quality scores. *Genome Research*, 18(11):1851–1858, 2008.

[19] R. Li, Y. Li, K. Kristiansen, and J. Wang. SOAP: short oligonucleotide alignment program. *Bioinformatics*, 24(5):713–714, 2008.

[20] J. C. Marioni, C. E. Mason, S. M. Mane, M. Stephens, and Y. Gilad. RNA-seq: An assessment of technical reproducibility and comparison with gene expression arrays. *Genome Research*, 18(9):1509–1517, 2008.

[21] R. D. Morin, M. A. Marra, M. Bainbridge, A. Fejes, M. Hirst, M. Krzywinski, T. J. Pugh, H. McDonald, R. Varhol, and S. J. Jones. Profiling the HeLa S3 transcriptome using randomly primed cDNA and massively parallel short-read sequencing. *BioTechniques*, 45(1):81–94, 2008.

[22] A. Mortazavi, B. A. Williams, K. McCue, L. Schaeffer, and B. Wold. Mapping and quantifying mammalian transcriptomes by RNA-Seq. *Nat Meth*, 5(7):621–628, 2008.

[23] U. Nagalakshmi, Z. Wang, K. Waern, C. Shou, D. Raha, M. Gerstein, and M. Snyder. The transcriptional landscape of the yeast genome defined by RNA sequencing. *Science*, 320(5881):1344–1349, 2008.

[24] J. Rougemont, A. Amzallag, C. Iseli, L. Farinelli, I. Xenarios, and F. Naef. Probabilistic base calling of Solexa sequencing data. *BMC Bioinformatics*, 9(1), 2008.

[25] S. Rumble, P. Lacroute, A. Dalca, M. Fiume, A. Sidow, and M. Brudno. SHRiMP: Accurate Mapping of Short Color-space Reads. *PLoS Computational Biology*, 5(5), 2009.

[26] U. Schulze, B. Hepp, C. Ong, and G. Ratsch. PALMA: mRNA to genome alignments using large margin algorithms. *Bioinformatics*, 23(15):1892, 2007.

[27] A. Smith, Z. Xuan, and M. Zhang. Using quality scores and longer reads improves accuracy of Solexa read mapping. *BMC Bioinformatics*, 9(1), 2008.

[28] M. Sultan, M. H. Schulz, H. Richard, A. Magen, A. Klingenhoff, M. Scherf, M. Seifert, T. Borodina, A. Soldatov, D. Parkhomchuk, D. Schmidt, S. O'Keeffe, S. Haas, M. Vingron, H. Lehrach, and M.-L. Yaspo. A global view of gene activity and alternative splicing by deep sequencing of the human transcriptome. *Science*, 321(5891):956–960, 2008.

[29] C. Trapnell, L. Pachter, and S. L. Salzberg. TopHat: discovering splice junctions with RNA-Seq. *Bioinformatics*, 25(9):1105–1111, 2009.

[30] C. Trapnell and S. L. Salzberg. How to map billions of short reads onto genomes. *Nat Biotech*, 27(5):455–457, 2009.

[31] J. C. Vera, C. W. Wheat, H. W. Fescemyer, M. J. Frilander, D. L. Crawford, I. Hanski, and J. H. Marden. Rapid transcriptome characterization for a nonmodel organism using 454 pyrosequencing. *Mol Ecol*, 17(7):1636–1647, 2008.

[32] Z. Wang, M. Gerstein, and M. Snyder. RNA-Seq: a revolutionary tool for transcriptomics. *Nat Rev Genet*, 10(1):57–63, 2009.

[33] N. Whiteford, T. Skelly, C. Curtis, M. E. Ritchie, A. Lohr, A. W. Zaranek, I. Abnizova, and C. Brown. Swift: primary data analysis for the Illumina Solexa sequencing platform. *Bioinformatics*, 25(17):2194–2199, 2009.

NEXT GENERATION SEQUENCING: STATISTICAL CHALLENGES AND OPPORTUNITIES

Somnath Datta[1], Ryan S. Gill[2], Seongho Kim[1], Sutirtha Chakraborty[1], and Susmita Datta[1*]

[1] Department of Bioinformatics and Biostatistics, University of Louisville, Louisville, KY

[2] Department of Mathematics, University of Louisville, Louisville, KY

*E-mail: susmita.datta@louisville.edu

Extended Abstract

Introduction: Next generation (NG) sequencing (Shendure and Hanlee 2008), also known as high throughput screening, is enabling the scientists with inspection capabilities of samples at an unprecedented genomic level. Its usefulness in various genotyping applications such as genome-wide detection of SNPs, methylome profiling, mRNA expression profiling and so on are well recognized. With new technologies come new challenges for the data analysts. In particular, statistical issues related to these novel massive data types are plentiful and so are the opportunities of developing clever and novel statistical techniques for the analyses of such data sets. In this paper we present a systematic review of statistical work related to next generation sequencing. This extended abstract divides the existing research work into a number of subtopics each of which will be reviewed at a greater detail in the full paper.

Data quality and reproducibility: Marioni et al. (2008) observed that next generation sequencing data from Illumina are highly reproducible. In an experiment measuring expression levels and detecting the differentially expressed genes between liver and kidney tissue samples, they found the new technology to be very reliable and overall superior to the microarray technology. Fu et al. (2009) arrived at a similar conclusion by comparing the relative accuracy of transcriptome sequencing (RNA-seq) and microarrays with protein expression data from adult human cerebellum using 2D-LC MS/MS. They found that the next generation sequencing provided more accurate estimation of absolute transcript levels. Wall et al. (2009) used simulation models to compare NG sequencing with traditional capillary-based sequencing. They concluded that NG sequencing offer great benefit in terms of coverage over capillary-based sequencing. However they suggested combining sequencing methodologies such as FLX and Solexa to achieve optimal performance at a modest cost.

A number of authors have reported problems and systematic biases with the sequence reads obtained in next generation sequencing, however. Dohm et al. (2008) found that wrong base calls are often preceded by base G. Base substitution errors were significantly disproportionate with A to C substitution error being 10 times more frequent than the C

to G substitution. Similar artifacts were observed by Irizarry and Bravo (2009) who reported A to T miscall to be the most common error in their calibration study. Both studies reported that the error rates vary with the position on the read. They also question the utility of the quality scores supplied by the manufacturers with a base call. These and other systematic biases may lead to wrong statistical conclusions. Oshlack and Wakefield (2009) demonstrated that when gene expression is calculated using aggregated tag counts for each gene in RNA-seq technology the ability to call differentially expressed genes (or ranking) between samples is strongly associated with the length of the transcript.

Some of the issues discussed above call for better base calling procedures than those provided by the manufacturers. A number of papers in recent months deals with this issue which we review next.

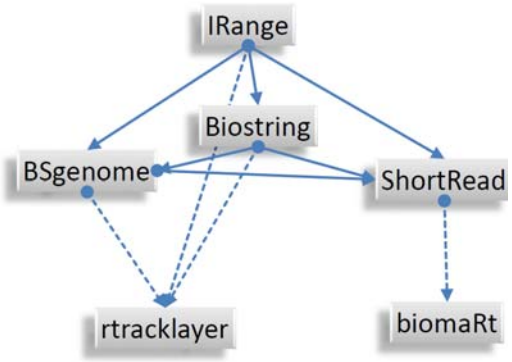
Base calling techniques: Most of the work in this important research area has taken place primarily for the Illumina platform. Rougemont et al. (2008) used probabilistic modeling and model-based clustering to identify and code ambiguous bases and to arrive at decisions to remove uncertain bases towards the ends of the reads. Kao *et al.* (2009) developed *BayesCall* primarily for the Illumina platform. This is based on Bayesian modeling and maximum a posteriori estimation. A more detailed review of their procedure will be provided in the full paper. They found that their procedure significantly improves the accuracy of base calling as compared to Illumina's basecaller. Another attempt to improve the Illumina basecaller led to Ibis (Improved base identification system) by Kircher *et al.* (2009). They use SVM (support vector machine) type classifiers into their basecalling procedure. Very recently, Irizarry and Bravo (2009) came up with their own modeling to quantify the variability in the generation of sequence reads as obtained from the Illumina/Solexa GA platform. Their models, which we review in greater detail in the paper, leads to improved base calling and related quality assessment. Schröder *et al.* (2009) proposed a novel search based algorithm using generalized suffix tree to correct for sequencing errors in NGS. They reported error correction accuracies of over 80% for simulated data and over 88% for real data.

Statistical methods for using sequence reads: Mapping software such as MAQ by Li *et al.* (2008) are useful in assembling short sequence reads to match a reference genome and making a final genotypic call. Bayesian calculations are used to this end. Dalevi *et al.* (2008) considered the problem of matching individual short reads sampled from the collective genome of a microbial community to protein families. Boyle et al. (2008) developed the software package F-Seq that employs kernel smoothing in converting high-throughput sequencing reads into continuous signals along a chromosome whose output can be displayed directly in the UCSC Genome Browser. This type of data summary will be useful to identify specific sequence features, such as transcription factor binding sites (ChIP-seq) or regions of open chromatin (DNase-seq). Zhang *et al.* (2008) developed MACS (Model-based Analysis of ChIP-seq) that utilizes Poisson modeling and to capture local biases in the genome resulting in for more robust predictions of binding sites. Jiang and Wong (2008) also used Poisson modeling in order to estimate the expression levels of various isoforms from NG RNA sequencing.

Applications: We also review a number of applications of the NG sequencing technology in a multitude of areas; each of these papers employ interesting novel statistical methods to use the sequencing data effectively which will be reviewed in further detail in the full paper. In a recent article, Choi *et al.* (2009) used NG ChIP-seq data together with array hybridization data towards enhancing the detection of transcription factor binding sites. This rather interesting analysis uses a hierarchical hidden Markov model to combine individual hidden Markov Models used with each data types. A similar combination of data types were used by Zang *et al.* (2009), who looked for spatial clusters of signals, for identification of ChIP enriched signals for histone modification profiles. Chu *et al.* (2009) applied whole genome sequencing to diagnose the fetal genetic disease using cell-free DNA from maternal plasma samples in the first trimester of pregnancy. Cokus et al. (2008) used NG sequencing to identify novel components of the Arabidopsis for methylation. In a rather potentially high impact application, Quon and Morris (2009) developed a statistical method to identify the primary origin of a cancer sample via next generation sequencing. This utilizes a detail profile of tissues of each primary origin and not a data based classifier.

R and Bioconductor packages: There are already a number of R and Bioconductor packages/tools for analyzing NGS data. The *rtracklayer* (Lawrence et al., 2009) package provides an interface between R and genome browsers. This package includes functions that import/export track data and control/query external genome browser sessions/views. The *chipseq* (Kharchenko et al., 2008) provides useful tools for design and analysis of ChIP-seq experiments and detection of protein-binding positions with high accuracy. These tools include functions that improve tag alignment and correct for background signals. The *Biostrings 2* (Pages, 2009) package allows users to manipulate big strings easily and fast by introducing new implementations and new interfaces into *Biostrings 1*.

Figure. 1: The dependency among the released R/Bioconductor packages. The solid lines represent the direct dependency and the dotted lines the indirect dependency.



The *ShortRead* package (Morgan et al., 2009) provides useful tools for analyzing high-throughput data produced by Solexa, Roche 454, and other sequencing technologies. These tools include input and output, quality assessment, and downstream analysis

functions. The *IRanges* package (Pages et al., 2009) includes functions for representation, manipulation, and analysis of large sequences and subsequences of data as well as tools for attaching information to subsequences and segments. The *BSgenome* package (Pages, 2009) provides infrastructure for accessing, analyzing, creating, or modifying data packages containing full genome sequences of a given organism. The *biomaRt* package (Durinck et al., 2006) allows users to connect to and search BioMart databases and integrates them with software in Bioconductor. This package includes functions that annotate identifiers with genetic information and that allow retrieval of data on genome sequences and single nucleotide polymorphisms. Several of these packages work in consort as shown in Figure 1.

References:

Boyle AP, Guinney J , Crawford, G and Furey, T (2008) F-Seq: a feature density estimator for high- throughput sequence tags. Bioinformatics 24: 2537-2538.

Choi H, Nesvizhskii A, Ghosh D, Qin, Z (2009) Hierarchical hidden Markov model with application to joint analysis of ChIP-chip and ChIP-seq data. Bioinformatics 25: 1715-1721.

Chu T, Bunce K, Hogge W and Peters D (2009) Statistical model for whole genome sequencing and its application to minimally invasive diagnosis of fetal genetic disease. Bioinformatics 25: 1244-1250.

Cokus SJ, Feng S, Zhang X, Chen Z, Merriman B, et al. (2008) Shotgun bisulphite sequencing of the Arabidopsisgenome reveals DNA methylation patterning. Nature 452: 215-219.

Dalevi D, Ivanova NN, Mavromatis K, Hooper SD, Szeto E et al.(2008) Annotation of metagenome short reads using proxygenes. Bioinformatics 24: i7-i13.

Dohm JC , Lottaz C, Borodina T and Himmelbauer H (2008) Substantial biases in ultra-short read data sets from high-throughput DNA sequencing. Nucleic Acids Research 36: e105.

Durinck S, Moreau Y, Kasprzyk A, Davis S, De Moor B et al. (2006). BioMart and Bioconductor: a powerful link between biological databases and microarray data analysis. Bioinformatics 21: 3439-3440.

Fu X, Fu N, Guo S, Yan Z, Xu Y et al. (2009) Estimating accuracy of RNA-Sequencing and microarrays with proteomics. BMC Genomics 10:161.

Irizarry RA and Bravo HC (2009) Model-based quality assessment and base-calling for second-generation sequencing data. Johns Hopkins University, Dept. of Biostatistics Working Papers, Paper 184.

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

Kao W, Stevens C and Song Y (2009) Bayes Call: A model-based basecalling algorithm for high-throughput short-read sequencing. Genome Res., published online August 6, 2009.

Kharchenko PV, Tolstorukov MY and Park PJ (2008). Design and analysis of ChIP-seq experiments for DNA-binding proteins. Nature Biotechnology 26: 1351–1359.

Kircher M, Stenzel U and Kelso J (2009). Improved base calling for the Illumina Genome Analyzer using machine learning strategies. Genome Biol. 10:R83.

Lawrence M, Gentleman R, and Carey V (2009). rtracklayer : an R package for interfacing with genome browsers. Bioinformatics, 25:1841-1842.

Li H, Ruan J and Richard D (2008). Mapping short DNA sequencing reads and calling variants using mapping quality scores. Genome Research 18:1851-1858.

Marioni J, Mason C, Mane S, Stephens M and Gilad Y (2008). RNA-seq: An assessment of technical reproducibility and comparison with gene expression arrays. Genome Research 18:1509-1517.

Morgan M, Anders S, Lawrence M, Aboyoun P, Pages H, and Gentleman R (2009). ShortRead: a Bioconductor package for input, quality assessment, and exploration of high throughput sequence data. Bioinformatics Advance Access published on August 3, 2009.

Oshlack A and Wakefield M (2009). Transcript length bias in RNA-sequencing data confounds systems biology. Biology Direct 4:14.

Pages H (2009). Biostrings. Retrieved from Biostrings:
http://www.bioconductor.org/packages/bioc/html/Biostrings.html

Pages H (2009). BSgenome: Infrastructure for Biostrings-based genome data packages. R package version 1.12.3.

Pages H, Aboyou P and Lawrence M (2009). IRanges: Infrastructure for manipulating intervals on sequences. R packages version 1.2.3.

Quon G and Morris Q (2009). ISOLATE: A computational strategy for identifying the primary origin of cancers using high throughput sequencing. Bioinformatics Advance Access published June 19, 2009.

Rougemont J , Amzallag A, Iseli C, Farinelli L, Xenarios I and Naef F (2008). Probabilistic base calling of Solexa sequencing data, BMC Bioinformatics 9:431.

Schröder J, Schröder H, Puglisi SJ, Sinha R and Schmidt B (2009). SHREC: a short-read error correction method. Bioinformatics 25: 2157–2163.

Shendure, J and Ji, H (2008). Next-generation DNA sequencing. Nature Biotechnology 26: 1135-1145.

Wall PK, Leebens-Mack J, Chanderbali AS, Barakat A, Wolcott E et.al. (2009). Comparison of next generation sequencing technologies for transcriptome characterization. BMC Genomics 10:347.

Zang C, Schones DE, Zeng C, Cui K, Zhao K and Peng W (2009) A clustering approach for identification of enriched domains from histone modification ChIP -Seq data. Bioinformatics 25, 1952-1958.

Zhang Y, Liu T, Meyer CA, Jérôme E , Johnson DS et al. (2008). Model-based Analysis of ChIP-Seq (MACS). Genome Biology 9:R137.

Using Next Generation Sequencing Data for Structural Annotation of *L. bicolor* Mycorrhizal Transcriptome

Peter E. Larsen[1], Geetika Trivedi[2], Avinash Sreedasyam[2] Vincent Lu[1], Gopi K. Podila[2] and Frank R. Collart[1]

[1]Biosciences Division, Argonne National Laboratory, Lemont, IL 60490
[2]Department of Biological Sciences, University of Alabama in Huntsville, Huntsville, AL

With the introduction and rapid adoption of next generation sequencing technologies, the number of completely sequenced genomes can be expected to continue to grow at a tremendous rate. The information derived from the genome assembly however is not directly translatable to biological understanding for an organism. For example, without accurate gene models, complete full genomic sequences can not be fully exploited for informatics or experimental applications. This limitation for accurate gene models can be attenuated by high-throughput, next-generation RNA sequencing of an organism's transcriptome which compliments the genomic sequence by providing the complete set of expressed genes under a specific set of biological conditions.

To evaluate the utility of a transcriptomic approach for improvement of structural annotation, we selected the fungus *L. bicolor*, a mycorrhizal symbiote of the tree *Populus tremuloides*. Fungal-plant symbiosis is a widespread process of major ecological importance and involves a progressive series of complex developmental steps accompanied by radical changes in metabolism and plant/fungal interactions with the environment. Little is known about the function or regulation of the critical proteins associated with the free living or mutualistic forms of this organism. We generated more than 24 million sequence 46 base pair reads with the Illumina "Genome Analyzer" to identify mRNAs associated with the free living, root exudate-treated, mycorrhizal forms

of the fungus. Approximately 50% of the current *L. bicolor* gene models expressed in our data set contain intron/exon boundaries that do not map to the mRNA sequence data. As accurate gene models are necessary for gene function identification and synthesis of gene products, this indicates that additional optimization of these gene models is needed to fully understand the metabolism and regulatory mechanisms of this important mycorrhizal symbiote. While advancement in algorithm design might incrementally improve the ability to identify potential genes from genomic sequence, only biological experimentation can validate those models.

For analysis, a 1,269 gene mycorrhizal transcriptome was generated from previously published microarray analysis of gene differentially expressed during mycorrhizal formation relative to the free living fungus. The approach to utilize a predefined set of gene models eliminated a requirement to assemble all collected reads into contiguous sequences or to search all possible splice-sites within the gene models. The first requirement necessitates a redundancy of effort to regenerate gene models that already exist in a polished, if not perfected, form. The second requirement is prohibitive due to the massive number of possible splice sites within a gene that need to be searched. Our direct approach utilizes the current set of *L. bicolor* models, RNA-seq data to confirm gene models and intron-exon boundaries. The algorithm searches for new intron-exon splice sites only when expression evidence does not support the current gene model. The ultra-fast alignment program "Bowtie 0.9.9" was used to align RNA-seq reads to (1) L. bicolor genomic sequence and to (2) the gene models of the mycorrhizal transcriptome. Errors in the gene models were identified by those regions where there was gene model without alignments from sets (1) or (2) and introns that were not spanned

by alignments from set (2). From the 1,269 gene models examined, 778 models were not found to have errors. Of the 487 gene models that contained errors, an average of 2.04 errors per model was found. Errors in models were corrected by identifying anchor sequences from up- and downstream of the errors, then searching the set of RNA-seq reads for contigs that bridge the anchor sequences. Because the assembled, corrected gene models represent only the expressed gene sequence, it is necessary to re-align the gene sequences to the *L. bicolor* genomic sequence in order to obtain information about the corrected gene structure. A slightly modified semi-global Smith-Waterman algorithm was used for this purpose. In the modified algorithm, low penalties are applied to gaps at the beginning and end of the alignment, high penalties assigned to gaps in the genomics sequence, and moderate penalties to gaps in the gene sequence. Additionally, a minimum of eight consecutive aligned nucleotides in the gene sequence is required to allow and alignment. After annotation by this method, over 80% of mycorrhizal transcriptome gene models were found to have all intron/exon boundaries that map the RNA-seq data. The genes with unresolved errors contain an average of 1.95 errors per gene model.

High-throughput transcriptomic data was found to provide far greater resolution for gene structure definition than a previously collected data set of ESTs. The annotation method used here did not achieve complete, errorless correction of the gene models. However, this is to be expected based on the limitations of the method the biological limitation associated with the small number of biological samples. This method assumes that all gene models are mostly accurate and contain only a few errors each. Although this is likely the case for many if not most of the gene models, a poor quality gene model or error in published genomic sequence will result in an unrecoverable gene. In addition,

a gene must be expressed at appreciable levels in the biological sample to be correctly annotated. If there is insufficient coverage of a gene model in the set of RNA-seq reads, there will be no way to correctly annotate that gene model. This method also only identifies how genes are ultimately transcribed, not how they are translated. Though a stop codon might be more easily identified, the specific start codon used for translation of the gene cannot be explicitly determined by this method. However, corrected gene structures, and even those gene models only incompletely corrected, improve capabilities for the prediction of protein function and are required for in vitro approaches to characterize the function of these proteins. This improved annotation process can be extended to other important gene families and will facilitate the process to identify the molecular mechanisms leading to the development of the mycorrhizal symbiosis and its implications in improving carbon sequestration by poplar. The methods described here are can also be generalized to any species or to accommodate additional biological conditions. With the growing adoption of next generation sequencing techniques, it is likely that this method and other similar transcriptomic analysis methodologies will prove to be indispensable companions to current and future genomic sequencing efforts.

Parallel Computing Strategies for Sequence Mapping of NGS Data (Extended Abstract)

Doruk Bozdag, Terry Camerlengo, Hatice Gulcin Ozer, Joanne Trgovcich, Tea Meulia, Kun Huang, Umit Catalyurek

Department of Biomedical Informatics
The Biomedical Informatics Shared Resource
The Ohio State University

Next-generation high throughput sequencing (NGS) instruments are capable of generating hundreds of millions of short sequences (reads) in a single run. Accurate and efficient mapping of this massive amount of reads to a reference genome is the most time consuming step in many biological application workflows. Numerous short sequence mapping programs have been developed to address this challenging task, each of them offering a different trade-off between speed and accuracy of the mapping results [1-15]. Still, even by using the fastest tool and allowing loss of some accuracy, it takes about a day to map hundreds of millions of reads to a mammalian genome [6, 16]. Therefore, there is a need for parallelization to further speed up the mapping process without compromising the accuracy. In this abstract, we discuss three strategies for parallelizing short sequence mapping: multithreading, cluster computing and cloud computing.

Short sequence mapping algorithms usually employ a two-step strategy. In the first step, either the reference genome or the read sequences are indexed and stored in the memory using a hash table, or a transformed array as in the case of Burrows-Wheeler Transform [6, 7, 9]. For simplicity in discussion, in the rest of this abstract we will assume that the reference genome is indexed in this step rather than the reads. Then, in the second step, reads are mapped to the reference genome by looking up the index structure for the matching locations. The accuracy of mapping depends on several factors such as sequencing and reading errors or existence of SNPs and repetitive regions in DNA. Computational cost grows very quickly as the desired level of accuracy increases. Therefore, in all sequence mapping programs accuracy is compromised in the following ways to limit the computation time:

- Limiting the number of allowed mismatches,
- Ignoring insertions and deletions or limiting their number and length,
- Ignoring base quality score information,
- Limiting the number of reported matching locations,
- Imposing constraints on read length,
- Ignoring information about errors particular to each sequencing technology.

Each mapping program has a unique way of trading-off these factors, and even for the best algorithmic approach, the accuracy can be improved at the cost of increased runtime. In this respect, parallel processing is inevitable to keep the runtime low while achieving higher accuracy. In the rest of this abstract, we will discuss three strategies for parallel short sequence mapping.

Multithreading: Using multiple threads to simultaneously execute independent portions of work on a multi-core computer is a common technique in parallel computing. In the context of short sequence mapping, multithreading is utilized to parallelize the second step of the mapping process by assigning blocks of reads to the threads [1, 2, 4, 6, 7, 9, 11, 14]. As a slightly different approach, in GMAP [15], reading input and writing output are handled by two separate threads whereas the rest of the threads are again assigned blocks of reads for mapping. The most significant drawback of multithreading is the poor scalability beyond a dozen threads due to shared use of memory and thread synchronization. Still, multithreading is relatively easy to use and effective for small-scale parallelization. Up to 3.1 and 3.6 speedup with four threads was reported for the second step of the mapping process using Bowtie [6] and SOCS [11] programs, respectively.

Cluster computing: A larger scale parallelization can be achieved by using a cluster of distributed memory computers. In a recent work, various parallelization techniques for hashing-based short sequence mapping on distributed memory computers have been introduced [16]. These methods are designed to optimize the distribution of genome and read sequence data on a cluster of computers to enhance parallel performance. In addition to read partitioning, genome partitioning is also considered, hence the first step of the mapping process is also parallelized to achieve better speedup in certain cases. Furthermore, partitioning the genome results in reduced memory footprint on each compute node which cannot be achieved via multithreading. This is especially important for mapping programs such as SOAPv1 [17], MapReads [3] and RMAP [13] that build a large hash table in the first step. In Figure 1, comparison of three of the parallelization methods is given for varying number of reads. As demonstrated in this example, efficient distribution of sequence and genome data helps improving the parallel execution time. In [16], execution time cost models for different parallelization techniques are also given to tune the numbers of genome and read blocks depending on the size of the reference genome and the number of reads to deliver optimum performance. Up to 22 speedup was reported while mapping 130 million Solid reads to a human reference genome on a cluster of 64 computers.

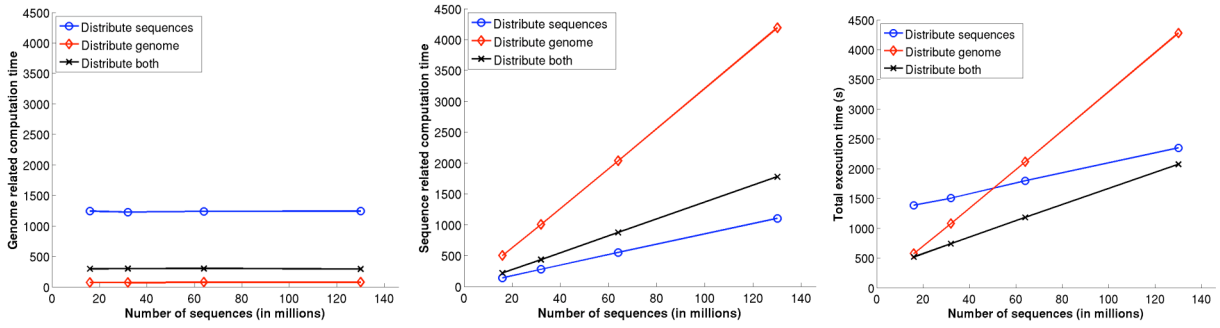


Figure 1. Comparison of three parallelization methods for mapping short sequences to a 800Mbp genome using parallelized MapReads [3] program on a 16-node cluster. **(a)** Time spent on the first step (hashing). **(b)** Time spent on the second step (mapping reads). **(c)** Total execution time.

Cloud computing: An alternative approach for parallel short sequence mapping is cloud computing. This idea is first introduced in CloudBurst [18] where distributed programming framework MapReduce is used to parallelize the RMAP [13] program. Ignoring the time to

transfer data files to the computing environment, the speedup obtained with this approach was between 10 and 12 using 24 nodes for different levels of sensitivity while mapping 7 million Illumina/Solexa reads to the human genome. We have also tested on running the ELAND algorithm using the Amazon Elastic Computing Cloud (EC2) system using 20 nodes. For mapping 7.6 million Illumina read to the human genome when the chromosomes were distributed onto different nodes, it took 28 minutes to finish the mapping (excluding the time for file transferring and selecting unique matched sequences).

To summarize, mapping NGS data to the reference algorithm is a highly parallelizable problem by nature. Currently many algorithms are available and they leave plenty of room for improving the mapping efficiency using parallel computing approaches. In addition, some new computing architectures such as the cloud computing and GPU provide low-cost alternatives to traditional computer clusters.

REFERENCES

1. *NovoAlign*. Available from: http://www.novocraft.com/.
2. *PerM*. Available from: http://code.google.com/p/perm/.
3. *MapReads*. Available from: http://solidsoftwaretools.com/gf/project/mapreads/.
4. *BFAST*. Available from: https://secure.genome.ucla.edu/index.php/BFAST.
5. Campagna, D., et al., *PASS: a program to align short sequences.* Bioinformatics, 2009. **25**(7): p. 967-8.
6. Langmead, B., et al., *Ultrafast and memory-efficient alignment of short DNA sequences to the human genome.* Genome Biol, 2009. **10**(3): p. R25.
7. Li, H. and R. Durbin, *Fast and accurate short read alignment with Burrows-Wheeler transform.* Bioinformatics, 2009. **25**(14): p. 1754-60.
8. Li, H., J. Ruan, and R. Durbin, *Mapping short DNA sequencing reads and calling variants using mapping quality scores.* Genome Res, 2008. **18**(11): p. 1851-8.
9. Li, R., et al., *SOAP2: an improved ultrafast tool for short read alignment.* Bioinformatics, 2009. **25**(15): p. 1966-7.
10. Lin, H., et al., *ZOOM! Zillions of oligos mapped.* Bioinformatics, 2008. **24**(21): p. 2431-7.
11. Ondov, B.D., et al., *Efficient mapping of Applied Biosystems SOLiD sequence data to a reference genome for functional genomic applications.* Bioinformatics, 2008. **24**(23): p. 2776-7.
12. Rumble, S.M., et al., *SHRiMP: accurate mapping of short color-space reads.* PLoS Comput Biol, 2009. **5**(5): p. e1000386.
13. Smith, A.D., Z. Xuan, and M.Q. Zhang, *Using quality scores and longer reads improves accuracy of Solexa read mapping.* BMC Bioinformatics, 2008. **9**: p. 128.
14. Stromberg, M. *Mosaik*. Available from: http://bioinformatics.bc.edu/marthlab/Mosaik.
15. Wu, T.D. and C.K. Watanabe, *GMAP: a genomic mapping and alignment program for mRNA and EST sequences.* Bioinformatics, 2005. **21**(9): p. 1859-75.
16. Bozdag, D., C.C. Barbacioru, and U.V. Catalyurek, *Parallel short sequence mapping for high throughput genome sequencing*, in *International Parallel and Distributed Processing Symposium*. 2009.

17. Li, R., et al., *SOAP: short oligonucleotide alignment program.* Bioinformatics, 2008. **24**(5): p. 713-4.
18. Schatz, M.C., *CloudBurst: highly sensitive read mapping with MapReduce.* Bioinformatics, 2009. **25**(11): p. 1363-9.

Comparative Analysis of Pol II and HIV-1 Sequences Using the W-Curve

Doug Cork and Steven Lembark
US Military HIV Program (MHRP)
HJF
1600 E. Gude Drive
Rockville, MD
20850

address correspondence to:

Col. Jerome Kim
US Military HIV Program (MHRP)
HJF
1600 E. Gude Drive
Rockville, MD
20850

The W-Curve was originally developed as a graphical visualization technique for viewing DNA sequences. Its ability to render features of the DNA also makes it suitable for computational studies. Its main advantage in this area is utilizing a single-pass algorithm for comparing the sequences: avoiding recursion offers advantages for speed and in-process resources. The graphical technique also allows for multiple models of comparison to be used depending on the nucleotide patterns embedded in similar whole genomic sequences.

We are currently using this technique to analyze HIV-1 sequences in some of the U.S. Army's HIV-1 Vaccine Cohort Studies (see http://www.hivresearch.org/ for details). The W-Curve approach allows us to compare large numbers of samples quickly. We are currently tuning the algorithm to accommodate quirks specific to HIV-1 so that it can be used to aid in diagnostic and vaccine efforts. Tracking the molecular evolution of the virus has been greatly hampered by gap associated problems that slow conventional string based alignments of the whole genome. The gaps predominate within the envelope gene of the virus.

This research describes the W-Curve algorithm itself, and how we have adapted it for comparison of similar HIV-1 genomes. A heuristic method has been used to align and tree similar HIV-1 genomes that have similar polar projectedW-Curves. Significant potential exist for utilizing this method in place of conventional string based alignment of HIV-1 genomes, such as ClustalX. It is well known that a gap problem exists in similar HIV-1 Genomes and that this slows down the processing time for alignment analysis of large numbers of sequenced whole HIV-1 Genomes, especially during real time cohort studies.

With W-Curve-based heuristic modifications to the alignment, it may be possible get clinically useful results in a short time --short enough to affect clinical choices for acute treatment.

Herein, we add a description of the generation process and comparison technique of aligning extremes of the curves to effectively phase-shift them past the HIV-1 whole genome gapproblem.

Initially, we examined the complete genomes of:

```
    HIV-U61  (9743 bases)
    HIV-BRU  (9229 bases)
    HIV-Mal  (9229 bases)
    HIV-HXB2 (9719 bases)
    HIV-M61  (9743 bases)
    HIV-MN   (9738 bases)
    HIV-H61  (9743 bases)
```

HIV-Mal is a recombinant of subtype A, D and K. M61 is subtype C. The rest are subtype B.

Initially U61, H61 and M61 clustered closely together using both conventional string based and W-Curve based Euclidean alignment approaches. With further heuristic refinement of gap penalty costs used in the W-Curve graphical alignments, we have attempted to more accurately reflect the conventionally accepted clustering of these HIV-1 subtypes as stated in the previous paragraph. The authors greatly appreciate the help of Sodsai Tovanabutra and Eric Sanders-Buell in the sequence analysis of HIV-1 whole genomes.

Given that the conference sample data includes Pol II genes, we would like to include them, along with HIV -1 Pol genes and other Pol II's we can find -- into the largest analysis we can perform. Our goal would be to analyze the performance of the W-Curve code against other methods for building large trees for performance and quality of the tree.