Table 1. Analytical strategy of alternative precursor messenger RNA (pre-mRNA) splicing altered in malignancy

Alteration	Analytical target	Representative methods
Cis-elements	Consensus splice sites, splice enhancers and silencers	Genomic DNA sequencing, conversion analysis
Trans-elements	Spliceosomes, SR proteins, hnRNP, RNA components	Western blotting, IHC, proteomic analysis
Aberrant splicing		
Aberrant splice variants	Individual transcripts (mRNA)	RT-PCR, cDNA sequencing, transcription assay with minigene constructs
Aberrant expression profile	An entire set of transcripts (mRNA)	Microarray, Northern blotting, RT-PCR

hnRNP, heterogenous nuclear ribonuculeoproteins; IHC, immunohistochemistry; SR proteins, serine/arginine-rich proteins.

Table 2. Public databases

Title	Web address
Catalogue of somatic mutations in cancer	http://www.sanger.ac.uk/genetics/CGP/cosmic/
The IARC TP53 mutation database	http://www-p53.iarc.fr/index.html
The Roche cancer genome database	http://rcgdb.bioinf.uni-sb.de/MutomeWeb/header.html
The p53 web site	http://p53.free.fr/index.html
Mismatch repair variant database	http://www.med.mun.ca/MMRvariants/
MMR gene unclassified variants database	http://www.mmruv.info
Locus specific mutation databases	http://www.hgvs.org/dblist/glsdb.html
Mutation database (The University of Tokyo)	https://reseq.lifesciencedb.jp/resequence/registration_j.html
Finnish disease heritage	http://www.findis.org/
The Singapore human mutation and polymorphism database	http://shmpd.bii.a-star.edu.sg/

IARC, International Agency for Research on Cancer; MMR, mismatch repair.

mutation at codon 461 within exon 12 of the *MLH1* gene, which caused skipping of exon 12 in three unrelated HNPCC families, and this alteration was functionally confirmed *in vitro*.[24] McVety *et al.*[25] reported that disruption of an ESE site also induced exon skipping in HNPCC, and the authors confirmed the exon skipping using *in vitro* minigene assay in COS-1 cells.[25] Pagenstecher *et al.*[26] reported frequent splice alterations in unclassified variants of *MSH2* and *MLH1*. These data indicated that analyses of transcripts should precede functional tests for the characterization of uncharacterized variants. Although several *in silico* algorithms for the prediction of splice variants have been developed, the prediction does not always correlate with results obtained through an *in vitro* and/or *in vivo* assay.[27] Further progress in the development of *in silico* algorithms is needed to apply the prediction in clinics.

Germline mutation in the *APC* gene (MIM #175100) is responsible for FAP, an inherited autosomal dominant disease characterized by several hundred adenomatous polyps in the colon and rectum. In 2004, Aretz *et al.*[28] reported the first systematic evaluation of several single-base substitutions in the *APC* gene at the splice sites or close to splice sites at the transcript level. In this study, one exonic mutation in exon 4 (c.423G>T) and three in exon 14 (c.1956C>T, c.1957A>G and c.1957A>C) led to complete exon skipping due to aberrant splicing, although they had been predicted to result in missense or silent mutations. One possible explanation for this effect may be the disruption of ESE motifs.[28] De Rosa *et al.*[29] disclosed the importance of NMD degradation in alternative splicing of the *APC* gene using cycloheximide, a chemical inhibitor of translation that is also known to inhibit NMD. Furthermore, it was reported that a SNP of the *dihydropyrimidine dehydrogenase* (*DPD*) gene IVS14 + 1G>A at the 5′SS was associated with reduced DPD activity[30] and severe toxicity in colorectal cancer patients treated with 5-fluorouracil.

As summarized above, alterations in *cis*-elements associate with hereditary cancer syndromes through disruption of splicing. However, the analyses describing the alterations in *cis*-elements and their resulting transcripts in sporadic cancer cases are still limited. In addition, the current prediction algorithms for altered splicing need to be improved. Therefore, further studies are necessary for the correct interpretation of alterations in *cis*-elements.

Alteration of *trans*-elements

Alterations of *trans*-elements in gastrointestinal cancer are summarized in Table 4. The current data of deregulated *trans*-elements in tumorgenesis are still limited compared with the alterations in *cis*-elements. Ghigna *et al.*[31] analyzed the expression of SR proteins and hnRNP in colon cancer tissues using a Northern blot analysis. They consequently demonstrated that expression of ASF, SRp40, SRp55 and other elements was more severely reduced in tumors showing a more altered CD44 splicing pattern.[31] Regarding CD44 splice variants in human cancers, there are a number of discrepancies in the published data and thus it appears to be difficult to reconcile all of those results.[32] Reportedly, hnRNP K, an RNA-binding protein that plays a role in RNA editing, alternative splicing and many other processes is upregulated in cancer tissues and involved in tumorigenesis through the modulation of gene expression in response to mitogenic stimuli.[33] It is of note that Klimek-Tomczak *et al.*[33] identified a mutation of *hnRNP K* c.274G>A in tumors and the surrounding mucosa, but mutation of *hnRNP K* was not found in individuals that were tumor free. This observation might suggest that the substitution is involved in the development of colorectal cancer through the deregulation of RNA editing. Matos *et al.*[34] found that RAC1b, an alternative splice variant lacking exon 3b, of the *RAC1* gene was overexpressed in a subset of colorectal tumors and that the expression was required to sustain tumor cell viability. Using an *in vitro* splicing assay with a RAC1 minigene construct in HT29 cells, they additionally showed that SRp20 increased RAC1 expression, and that ASF/SF2 acted as an enhancer of endogenous RAC1b splicing.[35] Consistently, induction of ASF/SF2 by the inhibition of the phosphatidylinositol 3-kinase (PI3K) pathway promoted RAC1b expression, whereas induction of SRp20 by the activation of β-catenin/TCF4 signaling inhibited the expression of RAC1b.[35] Recently, a proteomic approach together with IHC and other analyses revealed that hnRNP A2/B1, hnRNP F and other elements are upregulated in gastric cancer, and that SR-A1, hnRNP A1, hnRNP K and others are upregulated in colorectal cancer. Proteomic approaches have facilitated to identify the deregulation of *trans*-elements in tumorigenesis. The discovery will open a new avenue to study the association between expression of *trans*-elements and splicing alteration. As the sequence technologies are rapidly

doi: 10.1111/j.1349-7006.2010.01797.x

Table 3. Alteration of *cis*-elements in digestive tract malignancy

Cis-element (Gene)	Cancer type	Alteration type	Molecular aspects	Analytical methods	References
CDH1	HDGC	Germline mutation	c.1135 del8ins5 (IVS8+5 del8ins5) of *CDH1* caused three aberrant transcripts	Genomic and cDNA sequencing, IHC	20
FHIT	Gastric cancer (sporadic)	Somatic mutation	Aberrant transcripts of *FHIT* were diverse	LOH analysis, cDNA sequencing, western blotting	21
MYH	Gastric cancer (familial), KATO-III and other cell lines	Germline mutation	c.892–2A>G (IVS10–2A>G) at 3'SS of *MYH* caused a truncated protein	Genomic and cDNA sequencing, transfection assay	22
KLK12	Gastric cancer (sporadic)	Genetic polymorphism	KLK12 protein was absent in individuals with c.457+2C/C in intron 4, but not in those with the T/T or T/C	Genomic and cDNA sequencing, western blotting	23
MMR genes	HNPCC, COS-7 cell line	Germline mutation	The nonsense mutation within exon 12 of *MLH1* caused exon skipping in three unrelated families	*In vitro* translation analysis, *in vitro* transcription assay	24
	HNPCC, COS-1 cell line	Germline mutation	Disruption of an ESE at the 5' end of exon 3 of *MLH1* caused exon skipping	Genomic and cDNA sequencing, *in vitro* transcription assay	25
	HNPCC	Germline mutation	Some of the *MLH1* or *MSH2* single-base substitutions led to exon skipping, but others did not	Genomic and cDNA sequencing, RT-PCR	26
	HNPCC, COS-7 and other cell lines	Germline mutation	Computer predictions do not always correlate with *in vivo* splicing defects of *MLH1* or *MSH2*	*In silico* splicing analysis, *in vitro* transcription assay	27
APC	FAP	Germline mutation	Different single-base substitutions at or close to splice sites of *APC* were systematically evaluated	Genomic and cDNA sequencing, RT-PCR	28
	FAP, Caco-2 cell line	Germline mutation	*In vitro* experiments supported the importance of NMD in alternative splicing of *APC*	RT-PCR, cDNA sequencing, *in vitro* NMD assay with cycloheximide	29
DPD	Colon cancer (sporadic)	Genetic polymorphism	The splice site polymorphism IVS14+1G->A of *DPD* caused a reduction in DPD activity	Genomic sequencing, DPD activity assay	30

DPD, dihydropyrimidine dehydrogenase; FAP, familial adenomatous polyposis; HDGC, hereditary diffuse gastric cancer; HNPCC, hereditary non-polyposis colorectal cancer; IHC, immunohistochemistry; NMD, nonsense-mediated mRNA decay; SNP, single nucleotide polymorphism; 3'SS, 3' splice site.

developing, we hope that involvement of *trans*-elements in tumorigenesis will be resolved in the near future.

Aberrant expression profile of splice variants

The aberrant expression profile of splice variants in digestive tract malignancies has been discussed in various genes; altered expression was reported in genes including *CDH1* (*E-cadherin*), *CD82* (*KAI1*), *WISP1*, *BIRC5* (*survivin*), *CD44*, *FGFR4*, *FHIT*, *MUTYH* (*MYH*), *FGFR2*, *MUC1* and *CDCA1*[6] in gastric cancer, and in genes including *MLH1*, *CCND1* (*cyclin D1*), *APC*, *VEGF* (*VEGFA*), *RAC1*, *MST1R*, *TCF4*, *CTNNB1* and *TP53* (*p53*) in colorectal cancer. Among these genes, the function of the splice variants has been precisely analyzed for survivin (*BIRC5*) and *RAC1* (Fig. 3).

Survivin was initially identified as an anti-apoptotic oncogene, and its expression is enhanced in various tumors.[36] Survivin is classified into the IAP family that includes cIAP1, cIAP2 and XIAP, and the members have been considered to be potential therapeutic targets of several types of malignancies.[37] The elevated expression of survivin might play a role in tumorgenesis through increased tumor cell viability, and may render the anti-apoptotic property for cancer cells to overcome the cytotoxic effects of anticancer drugs.[37] Two alternative splice variants of *survivin* have been reported, Sur-DeltaEx3 (NM_001012270.1) and Sur2B (NM_001012271), by Mahotka *et al.*[38] (Fig. 3a). The authors additionally revealed that Sur2B lacks its anti-apoptotic potential and acts as an antagonist against naturally occurring survivin. In good agreement with the apoptotic role of Sur2B, the variant was downregulated in metastatic lesions of gastric cancer.[39] A clear understanding of the interplay of the pro- and anti-apoptotic functions of the *survivin* splice variants is required before successful anti-survivin therapies can be fully developed.[40]

As described in an earlier paragraph, the small GTPase *RAC1* gene has two splice variants: RAC1WT (NM_018890) and RAC1b (NM_006908). The RAC1b variant lacks alternative exon 3b and is shorter than RAC1WT, a wild-type variant with the in-frame inclusion of exon 3b, by 57 nucleotides in size (Fig. 3b). The splice variant RAC1b is overexpressed in a subset

Table 4. Alteration of *trans*-elements in digestive tract malignancy

Trans-element	Cancer type	Alteration type	Molecular aspects	Analytical methods	Selected references
hnRNP A2/B1, hnRNP F and others	Gastric cancer (sporadic)	Somatic alteration	The trans-elements are upregulated in gastric cancers	IHC, proteomic analysis	
ASF/SF2, SRp40, SRp55, SRp75, hnRNP A1, SRp20 and others	Colon cancer (sporadic)	Somatic alteration	The expressions of ASF/SF2, SRp40 and others were correlated to alternative CD44 splicing	RT-PCR, northern blot analysis	31
hnRNP K	Colon cancer (sporadic)	Somatic alteration	The *hnRNP K* mutation c.274G>A reflected an RNA editing in cancer	Genomic and cDNA sequencing, *in vitro* phospholyration assay	33
SRp20, ASF/SF2	Colon cancer (sporadic), HT29 cell line	Somatic alteration	SRp20 increases inclusion of exon 3b of *RAC1*, whereas ASF/SF2 increases its skipping	*In vitro* transcription assay	35
SR-A1, hnRNP A1, hnRNP K and others	Colon cancer (sporadic)	Somatic alteration	The *trans*-elements are upregulated in colorectal cancers	RT-PCR, IHC, proteomic analysis	

ASF/SF2, alternative splicing factor/splicing factor 2; ESE, exonic splice enhancer; FAP, familial adenomatous polyposis; hnRNP, heterogenous nuclear ribonuclear protein; IHC, immunohistochemistry; SRp, serine/arginine-rich protein.

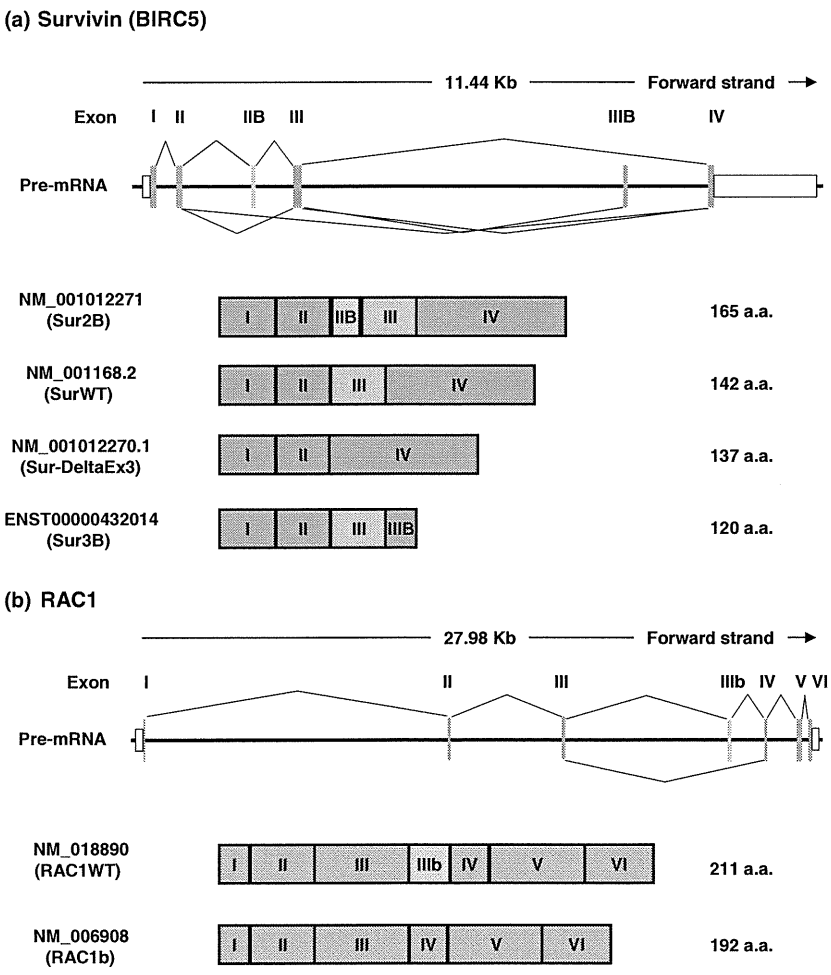

Fig. 3. Splice variants in digestive tract malignancy. (a) Splice variants of the *survivin* gene. (b) Splice variants of the *RAC1* gene. A GenBank accession number or an Emsembl transcript ID number for each variant is indicated on the left. Commonly used symbols for each variants are indicated in parentheses. The green boxes indicate constitutive exons and the blue boxes indicate alternative exons. a.a., amino acid.

of colorectal tumors, and is required to sustain tumor cell viability.[34] Additionally, another report showed that activation of the tumor-specific splice variant RAC1b decreased the adhesion of colorectal cancer cells.[41] On the other hand, RAC1WT but not RAC1b, stimulated RelB-mediated gene transcription in cells.[42]

The association of *VEGF* splice variants with colorectal cancer has been reported by several groups.[43,44] VEGF(xxx) are

doi: 10.1111/j.1349-7006.2010.01797.x

the pro-angiogenic isoforms and VEGF(xxx)b are the anti-angiogenic isoforms, where xxx denotes the amino acid number, and the isoform VEGF(xxx) or VEGF(xxx)b is defined by the alternative splicing of a mutually exclusive exon at the 3′ end, namely exon 8a or exon 8b. SRp55, ASF/SF2 and SRpK are shown to alter the selection of splice sites of the *VEGF* gene in colorectal cancer cells.[43] The elucidation of regulatory mechanisms of the splice variants will be helpful for the development of cancer-specific VEGF inhibitors.[44]

Therapeutic strategies targeting aberrant alternative splicing

There are two strategies to target alternative splicing: (i) target aberrant splice variants or their resulting products; and (ii) target *trans*-elements. Products that are generated from alternative pre-mRNA splicing and are playing a vital role in malignancy are potential therapeutic targets. If such products are identified, it will be possible to treat cancer patients more selectively and with individualized therapies by regulating the altered splice variants of the target gene rather than regulating the entire target gene. Therefore, the identification of functionally relevant cancer-specific splice variants as well as that of cancer-specific deregulation in *trans*-elements are important tasks for the development of future therapies.

Targeting aberrant splice variants may be achieved through conventional small molecules or RNA-based therapeutics including synthetically modified oligonucleotides, RNA interference,[45,46] ribozymes, aptamers and other strategies. Novel strategies targeting splice variants of *survivin* and *VEGF* have also been under investigation.[40,44] Since RNA-based molecular therapeutics target specific nucleotide sequences, it should have a wide range of targets and high selectivity. However, one of the most important issues to be resolved is the development of a drug delivery system suitable for the therapeutics.

Targeting *trans*-elements that act as spliceosomes[17] or splicing modulators is another option. Subunits in activated spliceosomes, hyperphosphorylation of SR proteins and upregulated splicing modulators in tumors are potential targets for cancer treatment. Recently, two natural products have been isolated for this purpose, the pladienolide derivatives[47] and spliceostatin A,[48] which bind to SF3b and inhibit spliceosomal function. Benzothiazole, a Clk1/Sty inhibitor,[49] inhibits ASF/SF2-dependent splicing through the suppression of Clk-mediated phosphorylation. The development of RNA therapeutics targeting the SRpK using the siRNA strategy has been ongoing, because activation of SRpK leads to the hyperphosphorylation of SR proteins, thereby activating splicing.

Conclusions

In the present review, we have summarized the current knowledge of the regulatory mechanisms involved in alternative pre-mRNA splicing and aberrant alternative splicing. In particular, we have focused on aberrant splicing and altered variant expression in gastrointestinal malignancies. As shown in this review, some of the silent/missense/nonsense mutations lead to exon skipping, retention of the intron or introduction of a new or cryptic splice site, although they are generally considered to result in no change, amino acid alteration or termination in amino acid sequence, respectively. Considering that *in silico* computer predictions do not always correlate with *in vitro* and *in vivo* splicing defects, we have to be careful of the interpretation of nucleotide changes in *cis*-elements. We also need to keep in mind that the regulatory mode of alternative pre-mRNA splicing might change in organ-, tissue- or cell-dependent manners. Therefore, accumulation of data on alternative splicing in different normal as well as malignant tissues is of great importance. In addition, information on the changes in *trans*-elements in neoplasms is so far limited. Since changes in splicing cannot be explained by the information of *cis*-elements alone, we have to accumulate knowledge on the changes of *trans*-elements in normal and malignant tissues and their roles in each type of cells. Acquisition of a huge body of human genome and transcript information has started to decipher the splicing codes and unveil the insights of splicing mechanisms.[1,50] The integrated information of the *cis*- and *trans*-elements and that of splice variants will help us develop more accurate prediction algorithms of the aberrant splicing in each type of tumors. As splice variants and altered *trans*-elements specifically in malignant tissues are promising targets for diagnosis and anticancer drugs, we hope that the integration of genome, transcriptome, proteome, functional and clinical data will make rapid progress in the development of new diagnostic and therapeutic strategies.

Disclosure statement

The authors declare no competing interests.

References

1 Blencowe BJ. Alternative splicing: new insights from global analyses. *Cell* 2006; **126**: 37–47.
2 International Human Genome Sequencing Consortium. Finishing the euchromatic sequence of the human genome. *Nature* 2004; **431**: 931–945.
3 Modrek B, Resch A, Grasso C, Lee C. Genome-wide detection of alternative splicing in expressed sequences of human genes. *Nucleic Acids Res* 2001; **29**: 2850–2859.
4 Lee C, Atanelov L, Modrek B, Xing Y. ASAP: the Alternative Splicing Annotation Project. *Nucleic Acids Res* 2003; **31**: 101–105.
5 Black DL. Mechanisms of alternative pre-messenger RNA splicing. *Annu Rev Biochem* 2003; **72**: 291–336.
6 Ohnuma S, Miura K, Horii A *et al.* Cancer-associated splicing variants of the CDCA1 and MSMB genes expressed in cancer cell lines and surgically resected gastric cancer tissues. *Surgery* 2009; **145**: 57–68.
7 Burge CB, Tuschl T, Sharp PA. Splicing of precursors to mRNAs by the spliceosomes. In: Gesteland RF, Cech TR, Atkins JF, eds. *The RNA World II*. New York, NY: Cold Spring Harbor Laboratory Press, 1999; 525–560.
8 Wahl MC, Will CL, Lührmann R. The spliceosome: design principles of a dynamic RNP machine. *Cell* 2009; **136**: 701–718.
9 Chen M, Manley JL. Mechanisms of alternative splicing regulation: insights from molecular and genomics approaches. *Nat Rev Mol Cell Biol* 2009; **10**: 741–754.
10 Krecic AM, Swanson MS. hnRNP complexes: composition, structure, and function. *Curr Opin Cell Biol* 1999; **11**: 363–371.
11 Pajares MJ, Ezponda T, Catena R, Calvo A, Pio R, Montuenga LM. Alternative splicing: an emerging topic in molecular and clinical oncology. *Lancet Oncol* 2007; **8**: 349–357.
12 Zahler AM, Neugebauer KM, Lane WS, Roth MB. Distinct functions of SR proteins in alternative pre-mRNA splicing. *Science* 1993; **260**: 219–222.
13 Cáceres JF, Kornblihtt AR. Alternative splicing: multiple control mechanisms and involvement in human disease. *Trends Genet* 2002; **18**: 186–193.
14 Warzecha CC, Sato TK, Nabet B, Hogenesch JB, Carstens RP. ESRP1 and ESRP2 are epithelial cell-type-specific regulators of FGFR2 splicing. *Mol Cell* 2009; **33**: 591–601.
15 Singh R, Valcárcel J. Building specificity with nonspecific RNA-binding proteins. *Nat Struct Mol Biol* 2005; **12**: 645–653.
16 McGlincy NJ, Smith CW. Alternative splicing resulting in nonsense-mediated mRNA decay: what is the meaning of nonsense? *Trends Biochem Sci* 2008; **33**: 385–393.
17 van Alphen RJ, Wiemer EA, Burger H, Eskens FA. The spliceosome as target for anticancer treatment. *Br J Cancer* 2009; **100**: 228–232.
18 den Dunnen JT, Antonarakis SE. Nomenclature for the description of human sequence variations. *Hum Genet* 2001; **109**: 121–124.
19 Guilford P, Hopkins J, Harraway J *et al.* E-cadherin germline mutations in familial gastric cancer. *Nature* 1998; **392**: 402–405.

20 Oliveira C, de Bruin J, Nabais S *et al.* Intragenic deletion of CDH1 as the inactivating mechanism of the wild-type allele in an HDGC tumour. *Oncogene* 2004; **23**: 2236–2240.
21 Lee SH, Kim WH, Kim HK *et al.* Altered expression of the fragile histidine triad gene in primary gastric adenocarcinomas. *Biochem Biophys Res Commun* 2001; **284**: 850–855.
22 Tao H, Shinmura K, Hanaoka T *et al.* A novel splice-site variant of the base excision repair gene MYH is associated with production of an aberrant mRNA transcript encoding a truncated MYH protein not localized in the nucleus. *Carcinogenesis* 2004; **25**: 1859–1866.
23 Shinmura K, Tao H, Yamada H *et al.* Splice-site genetic polymorphism of the human kallikrein 12 (KLK12) gene correlates with no substantial expression of KLK12 protein having serine protease activity. *Hum Mutat* 2004; **24**: 273–274.
24 Stella A, Wagner A, Shito K *et al.* A nonsense mutation in MLH1 causes exon skipping in three unrelated HNPCC families. *Cancer Res* 2001; **61**: 7020–7024.
25 McVety S, Li L, Gordon PH, Chong G, Foulkes WD. Disruption of an exon splicing enhancer in exon 3 of MLH1 is the cause of HNPCC in a Quebec family. *J Med Genet* 2006; **43**: 153–156.
26 Pagenstecher C, Wehner M, Friedl W *et al.* Aberrant splicing in MLH1 and MSH2 due to exonic and intronic variants. *Hum Genet* 2006; **119**: 9–22.
27 Lastella P, Surdo NC, Resta N, Guanti G, Stella A. In silico and in vivo splicing analysis of MLH1 and MSH2 missense mutations shows exon- and tissue-specific effects. *BMC Genomics* 2006; **7**: 243.
28 Aretz S, Uhlhaas S, Sun Y *et al.* Familial adenomatous polyposis: aberrant splicing due to missense or silent mutations in the APC gene. *Hum Mutat* 2004; **24**: 370–380.
29 De Rosa M, Morelli G, Cesaro E *et al.* Alternative splicing and nonsense-mediated mRNA decay in the regulation of a new adenomatous polyposis coli transcript. *Gene* 2007; **395**: 8–14.
30 van Kuilenburg AB, Muller EW, Haasjes J *et al.* Lethal outcome of a patient with a complete dihydropyrimidine dehydrogenase (DPD) deficiency after administration of 5-fluorouracil: frequency of the common IVS14+1G>A mutation causing DPD deficiency. *Clin Cancer Res* 2001; **7**: 1149–1153.
31 Ghigna C, Moroni M, Porta C, Riva S, Biamonti G. Altered expression of heterogenous nuclear ribonucleoproteins and SR factors in human colon adenocarcinomas. *Cancer Res* 1998; **58**: 5818–5824.
32 Naor D, Nedvetzki S, Golan I, Melnik L, Faitelson Y. CD44 in cancer. *Crit Rev Clin Lab Sci* 2002; **39**: 527–579.
33 Klimek-Tomczak K, Mikula M, Dzwonek A *et al.* Editing of hnRNP K protein mRNA in colorectal adenocarcinoma and surrounding mucosa. *Br J Cancer* 2006; **94**: 586–592.
34 Matos P, Oliveira C, Velho S *et al.* B-Raf(V600E) cooperates with alternative spliced Rac1b to sustain colorectal cancer cell survival. *Gastroenterology* 2008; **135**: 899–906.
35 Gonçalves V, Matos P, Jordan P. Antagonistic SR proteins regulate alternative splicing of tumor-related Rac1b downstream of the PI3-kinase and Wnt pathways. *Hum Mol Genet* 2009; **18**: 3696–3707.
36 Ambrosini G, Adida C, Altieri DC. A novel anti-apoptosis gene, survivin, expressed in cancer and lymphoma. *Nat Med* 1997; **3**: 917–921.
37 Miura K, Karasawa H, Sasaki I. cIAP2 as a therapeutic target in colorectal cancer and other malignancies. *Expert Opin Ther Targets* 2009; **13**: 1333–1345.
38 Mahotka C, Wenzel M, Springer E, Gabbert HE, Gerharz CD. Survivin-deltaEx3 and survivin-2B: two novel splice variants of the apoptosis inhibitor survivin with different antiapoptotic properties. *Cancer Res* 1999; **59**: 6097–6102.
39 Krieg A, Mahotka C, Krieg T *et al.* Expression of different survivin variants in gastric carcinomas: first clues to a role of survivin-2B in tumour progression. *Br J Cancer* 2002; **86**: 737–743.
40 Sampath J, Pelus LM. Alternative splice variants of survivin as potential targets in cancer. *Curr Drug Discov Technol* 2007; **4**: 174–191.
41 Esufali S, Charames GS, Pethe VV, Buongiorno P, Bapat B. Activation of tumor-specific splice variant Rac1b by dishevelled promotes canonical Wnt signaling and decreased adhesion of colorectal cancer cells. *Cancer Res* 2007; **67**: 2469–2479.
42 Matos P, Jordan P. Rac1, but not Rac1B, stimulates RelB-mediated gene transcription in colorectal cancer cells. *J Biol Chem* 2006; **281**: 13724–13732.
43 Qiu Y, Hoareau-Aveilla C, Oltean S, Harper SJ, Bates DO. The anti-angiogenic isoforms of VEGF in health and disease. *Biochem Soc Trans* 2009; **37**: 1207–1213.
44 Harper SJ, Bates DO. VEGF-A splicing: the key to anti-angiogenic therapeutics? *Nat Rev Cancer* 2008; **8**: 880–887.
45 Tuschl T, Zamore PD, Lehmann R, Bartel DP, Sharp PA. Targeted mRNA degradation by double-stranded RNA in vitro. *Genes Dev* 1999; **13**: 3191–3197.
46 Fire A, Xu S, Montgomery MK, Kostas SA, Driver SE, Mello CC. Potent and specific genetic interference by double-stranded RNA in *Caenorhabditis elegans*. *Nature* 1998; **391**: 806–811.
47 Kotake Y, Sagane K, Owa T *et al.* Splicing factor SF3b as a target of the antitumor natural product pladienolide. *Nat Chem Biol* 2007; **3**: 570–575.
48 Kaida D, Motoyoshi H, Tashiro E *et al.* Spliceostatin A targets SF3b and inhibits both splicing and nuclear retention of pre-mRNA. *Nat Chem Biol* 2007; **3**: 576–583.
49 Muraki M, Ohkawara B, Hosoya T *et al.* Manipulation of alternative splicing by a newly developed inhibitor of Clks. *J Biol Chem* 2004; **279**: 24246–24254.
50 Barash Y, Calarco JA, Gao W *et al.* Deciphering the splicing code. *Nature* 2010; **465**: 53–59.

doi: 10.1111/j.1349-7006.2010.01797.x

In Search of True Reads: A Classification Approach to Next Generation Sequencing Data Selection

Edward Wijaya*†, Jean-François Pessiot*, Martin C. Frith*, Wataru Fujibuchi*, Kiyoshi Asai*†, and Paul Horton*‡
*AIST, Computational Biology Research Center, 2-42 Aomi, Koutou-Ku, Tokyo 1350064
†Department of Computational Biology, Graduate School of Frontier Science,
the University of Tokyo, 5-1-5, Kashiwanoha, Kashiwa, Chiba 2778561
‡Email: horton-p@aist.go.jp

***Abstract*—Next generation sequencing (NGS) technology has increasingly become the backbone of transcriptomics analysis, but sequencer error causes biases in the read counts. In this paper we establish a framework for predicting true sequences from NGS data. We formulate this task as a classification problem. We define several features, such as log likelihood ratio of estimated true counts, error probability and observed count of the reads. Using a Support Vector Machine (SVM) classifier, we show that on simulated reads these features can achieve 96.35% classification accuracy in discriminating true sequences. Using this framework we provide a way for users to select sequences with a desired precision and recall for their analysis. The feature generation software and the simulated data set can be obtained from (`http://seq.cbrc.jp/NGSFeatGen`).**

***Keywords*-next generation sequencing; transcriptomics; Illumina; Solexa; expectation maximization; classification**

I. Introduction

Recent advances in DNA sequencing technologies enables transcripts to be measured with unprecedented accuracy and resolution. However these technologies also have a sequencing error rate that creates biases by yielding false sequences [6], [11], and therefore can significantly reduce the quality of the conclusions which can be drawn from the data.

The first batch of sequence count correction methods [1]–[4] were designed for SAGE data [16]. However the software created at that time is not able to handle the large datasets generated by next generation sequencers.

Recently several methods have been developed for correcting errors in DNA reads. One class of methods is aimed at genome sequencing: they assume that the genome is sequenced with high coverage, so that correct sequences tend to be present many times in the data, whereas incorrect sequences tend to be present fewer times [12], [14]. So these methods identify erroneous sequences by their rareness. This approach is not suitable for transcriptome or metagenome sequencing, where correct sequences are often rare. For these sequencing applications, more sophisticated error-correction methods, which consider the entire ensemble of reads when correcting are needed.

At least two such tools have been made for sequence count error correction in next generation sequencing data. The first work, FreClu, by Qu *et. al* [11] involves an iterative procedure to cluster reads and performs the sequencing error test for each cluster to assess the reads membership to the cluster. Finally the estimated true counts are computed from the total frequency of reads inside the cluster. The second is our tool RECOUNT [18]. RECOUNT adopts the method proposed by Beißbarth et al. [2] which models the sequencer error as a multinomial thinning process and applies the Expectation-Maximization (EM) framework [5] to infer the set of true counts which (locally) maximize the likelihood of the observed reads.

Despite the fact that these methods are able to give an estimate of true counts, to our knowledge there has been no quantitative evaluation of their ability to distinguish true from false sequences. In this paper we propose a framework for predicting true sequences from next generation sequencing data. We define true sequences as those that would have been output by an error-less sequencer. We approach this task as a binary classification problem, in which we learn a function mapping a feature set to 0 (sequences with zero real count) or 1 (sequences with non-zero real count). We defined six features for this task: observed count, estimated true count from error correction tool (RECOUNT), log likelihood with entropy penalty, log likelihood ratio, expectation matching score, and self-correctness coefficient (SCC).

Using an SVM classifier on simulated data sets, we show that these six features can achieve 96.35% accuracy. By making use of the predictor's posterior probability, we also suggest a way for users to determine sequences for their downstream analysis with an appropriate tradeoff between precision and recall.

This paper is organized as follows. First we describe our procedure for creating the simulated data set; second, we explain in detail our feature generation step; third, we show our experimental results, discuss the misclassified sequences, and conclude.

II. Data Simulator

In this section we describe our simulator to produce artificial datasets of sequence reads in which we know the

true sequence behind every read. Our simulator assumes a probability distribution of true sequences and of sequencer error, but rather than arbitrarily defining these distributions, we base them on real read datasets in a semi-empirical way described below. For lack of a more precise term, we call this procedure "training" the simulator on the given dataset. Our simulator first selects a pre-simulated dataset and then generates a post-simulated dataset, which simulates the reads which one might obtain in a real sequencing experiment.

A. Pre-simulated dataset

1) Extract 100K reads with high quality from a real experiment by randomly sampling reads with average Solexa quality greater than 33. [1]. Keep the sequences and discard their quality score information.
2) Randomly sample 100K reads of any quality from the same experiment, and retain only their quality data.
3) The final pre-simulated data set is obtained by randomly pairing the obtained sequences and quality score vectors.

B. Post-simulated dataset

We obtain this dataset by mutating bases in each sequence chosen above according to the probabilities stipulated by their assigned quality scores.

The *real* count of a sequence is its count in the *pre-error-simulated* set. Our classification procedure is to generate features from the *post-error-simulated* data set and use them to identify which sequences in that data set have zero or non-zero real count.

Subsequently we will simply denote the *pre-error-simulated* dataset as the *real* dataset and *post-error-simulated* set as the *simulated* dataset.

For the simulations described here, we trained our simulator on two real datasets: *B. vulgaris* genomic clone [6] reads (27bp) and *Drosophila* somatic cell mRNA [10] reads (36bp). Although read lengths generated by Illumina/Solexa recently have increased to 100bp or more, reads with shorter length are still often used in biological experiments.

III. Method

A. Features Generation

1) Observed Count: We define the observed sequence of a read in the usual way, as the sequence obtained by calling the base with highest probability according to the quality scores in each position. The observed count of a sequence s, is the number of reads whose observed sequence is s.

[1]Note that because of this threshold, depending on the type of data set and the quality score, the final number of sampled tags will be $\leq$ 100K.

2) Predicted True Count: We obtained the estimated true count from RECOUNT. Our RECOUNT software has been described elsewhere [18] and it closely follows the method of Beißbarth et al. [2] to infer true counts. For the reader's convenience, we summarize the method here.

Mostly following the notation of Beißbarth et al. [2]; α_{ij} denotes the probability that a true sequence i generates an observed sequence j (when the read is called correctly $i = j$). u denotes the total number of unique sequences, which in principle includes all possible DNA sequences with the right length, but in practice we approximate by only considering sequences observed at least once). For the ith sequence, $1 \leq i \leq u$, n_i denotes its observed count and m_i its true count.

In forming a probability model, we assume the true sequence counts follow a Poisson distribution, namely given a true proportion p_j of a tag j, the true count is m_j with probability:

$$\frac{e^{-p_j\lambda}(p_j\lambda)^{m_j}}{m_j!} \tag{1}$$

for a fixed λ.

We adopt the Expectation Maximization algorithm [2], [5] to calculate the true counts given the observed counts and sequencing error rate estimates. The parameters we want to estimate are λ and the p_j's, under the constraint that the p_j's add to one. The log likelihood function is given by:

$$-\lambda + \sum_{j=1,\cdots,u} \hat{m}_j log(p_j\lambda) \tag{2}$$

The details of the EM algorithm are as follows:

1) E-step: Compute the likelihood and expected count of a sequence j given by:

$$\hat{m}_j = \sum_{i=1,\ldots,u} \left(\frac{\alpha_{ij}p_j}{\sum_{k=1,\ldots,u} \alpha_{ik}p_k} \cdot n_i \right) \tag{3}$$

2) M-step: Maximize the likelihood of the complete data given the expected values and re-calculate new estimates for the parameters: $\hat{\lambda} = \sum_{k=1,\ldots,u} \hat{m}_k$ and $\hat{p}_j = \hat{m}_j/n$, where n is the total read count. Note, the total read count is equivalent to the total sequence count, but different than the total *unique* sequence count.

We iterate these steps until the parameters converge. We initialize the expected values $\hat{m}_j$ with the observed count of read j. The following two features also make use of the log likelihood, formula (2).

3) Entropy Penalty to Log likelihood: In general, the sequences found in a biological sample are expected to cover only a small amount of fraction of the possible sequence space (e.g. the 4^{36} possible length 36 DNA sequences). Based on this prior knowledge, we introduce an extra entropy term to favor sparse solutions, i.e. solutions in which

the number of inferred unique sequences is relatively small. Our modified optimization function is:

$$-\lambda + \sum_{j=1,\cdots,u} \hat{m}_j log(p_j \lambda) + \beta \sum_{j=1,\ldots,u} p_j \log p_j \quad (4)$$

where $\beta \geq 0$ is a user-defined parameter which controls the tradeoff between the likelihood and the sparsity of the solution. When $\beta = 0$, we only focus on maximizing the log likelihood and ignore the sparsity constraint. The higher the value of β is, the more we focus on the model's sparsity.

Since there is no closed form expression for the parameters to maximize equation 4 with respect to the p_j's, we use the conjugate gradient method [9] for the "M step". The m_j's are then updated in the "E step" by equation 1 as in the standard EM procedure.

4) Zero-clamped Log Likelihood Ratio: Above we described the probabilistic model of RECOUNT, where the estimate true count for each observed sequence is determined by expectation maximization. At convergence we also obtain the log likelihood (from equation 2) of the observed data given the inferred true sequence counts. In other words, this is a kind of maximum likelihood estimation.

We conjectured that it would be useful to also compute the likelihood under the constraint that a sequence s of interest is forced to have a true count of zero. Our reasoning is that if clamping that sequence to zero significantly reduces the overall likelihood of the observed data, then it is likely that the sequence actually has a non-zero true count.

More formally, let L be the unconstrained likelihood and $L_{s=0}$ be the likelihood obtained when the true count of sequence s is clamped to zero. The zero-clamped log likelihood ratio for sequence s (or for brevity just *loglikelihood ratio*) is defined as $L - L_{s=0}$.

To compute this we modified RECOUNT to allow p_s for a specified sequence s to be clamped to zero during the calculation. Algorithm 1 illustrates the whole process.

Algorithm 1 Zero-clamped Log Likelihood Ratio

1: Run RECOUNT and compute log likelihood L
2: **for** Each sequence s in library **do**
3: Compute $L_{s=0}$ by running RECOUNT with p_s clamped to zero.
4: Output $L - L_{s=0}$
5: **end for**

Time complexity: Unfortunately, the time complexity of this computation increases quadratically with the number of observed sequences N, which would make the procedure impractical if implemented naïvely. Fortunately, clamping a single probability p_s only affects a small portion of the likelihood computation and the rest can be efficiently reused.

To be more precise, we first define the *plausible misread graph* of an input dataset as an undirected graph whose nodes are the observed sequences and each edge connects two sequences s and r which are similar enough that a true sequence s could plausibly be misread as r (or vice versa). RECOUNT allows the user to stipulate this graph to be defined using either hamming distance or the probability of misreading s as r from the average quality scores of each read called as s. The results discussed here use one hamming distance edges.

In the plausible misread graph, consider C_s, the connected component of a sequence s. We note (without presenting a formal proof) that in the likelihood computation of equation 2, a change in p_s can only affect terms n_r, m_r for sequences r which are in C_s. Using this observation, we can compute $L_{s=0}$ efficiently by reusing the results from the unconstrained likelihood L and only recomputing the terms corresponding to C_s. Using this technique, the running time for generating $L_{s=0}$ for all observed sequence on 100K Solexa/Illumina reads was ~1.5 hours on a 2.9GHz 32 RAM Linux machine. Although we did not pursue this, we note in passing that the computation of the $L_{s=0}$'s is easily parallelizable.

5) Expectation Matching: We coin the term "Expectation Matching" to describe a simple alternative to maximizing an explicit likelihood as in equation 2. In short, expectation matching is an iterative heuristic procedure which takes advantage of the fact that, if we assume we know the true counts, it is easy to compute the expected observed counts.

More formally, the expected counts of the jth sequence are:

$$\text{sequence } j \text{ expected count } = \sum_{i=1,\cdots,u} m_i * \alpha_{ij} \quad (5)$$

where m_i is the assumed true count of the ith sequence and (as in previous sections) u denotes the total number of unique sequences, and α_{ij} denotes the probability that a read of sequence i is called as j. Let M denote an assumed true count vector and $E(M)$ denote the expected observed counts of M.

We define our task as finding an estimated true counts vector $\hat{M}$, such that $E(\hat{M})$ is close to the observed counts N. Algorithm 2 describes the procedure of expectation matching, which uses a form of gradient descent to improve an initial estimate of $\hat{M}$.

For the convergence criteria, we currently use the maximum (absolute value) sequence count change.

An advantage of this approach is its simplicity and speed. Also it conserves the total sequence count between N and M, which is reasonable since we only intend to model the miscalling of reads – not their loss or gain. Note that this property is not shared with the particular Expectation Maximization based approach described in previous sections. However this approach lacks an explicit probabilistic model

Algorithm 2 Expectation Matching

Require: γ holds learning rate
Require: η holds decay rate for γ
Require: N holds observed sequence counts
Let $M = N$
Let done = *false*
while !done **do**
  Let $\Delta = N - E(M)$
  Let $C = \Delta * \gamma$
  Let $\gamma = \gamma * \eta$
  Let $M = M + C$
  if change C small enough **then**
    Let done = *true*
  end if
end while

and suffers from the fact that it sometimes infers negative counts for some sequences.

This approach is in the same spirit as the work by Colinge, et.al, [4], except that they use a sophisticated (and more time consuming) Lanczos numerical algorithm to compute $\hat{M}$.

6) Self-Correctness Coefficient (SCC): If an observed sequence s is false, it must be the case that every read which was called as s was in fact a misread. More formally, let Pc_{ij} denote the probability that the ith such read was called correctly in position j; a quantity indictaed by the quality scores of read i. Let l denote the read (and therefore sequence length) of the input data. The probability that all M_s reads called as sequence s, were miscalled is:

$$\mathrm{SCC}_s = 1 - \prod_{i=1..M_s} (1 - \prod_{j=1..l} Pc_{ij}) \qquad (6)$$

We call this the "Self-Correctness Coefficient", because it is a very simplistic measure which only considers reads called as s rather than the whole ensemble of reads. We speculate this coefficient should be able to compensate for a weakness of the ensemble methods as they are implemented. For efficiency reasons, we and others do not consider all possible sequences, but only those with non-zero observed counts. This approximation generally works well, but breaks down when there is a significant probability that a read comes from a sequence which has zero observed count. The result is that the ensemble methods are forced to accept all observed sequences which are isolated in the plausible misread graph.

IV. Results

For this study, we adopted the Support Vector Machine (SVM) classifier [15]. All the experiments were done with the SVM implementation under the klaR package for R [17], using a radial basis kernel function. The accuracy estimates were made by using leave-one-out cross-validation and selected from the best results from the following SVM's hyper-parameter ranges: C = (4, 8, 16, 32, 64, 128, 256) and γ = (0.5, 1, 2), yielding the combination: $C = 256$, $\gamma = 2$.

A. Prediction Accuracy

We measured the accuracy by using each feature by itself and with all six features. There are no hyper-parameters involved for generating the features except for the *entropy penalty*, for which we use $\beta = 100$. We use the simulator output when trained from *Beta vulgaris* dataset to test the accuracy. It contains 20,576 positive class and 25,632 negative class.

Table I shows the accuracy of prediction using single features. The best feature was the true count as inferred by the EM formulation. The observed count feature can be viewed as a sort of baseline. When we combined all six features we obtained a cross-validated accuracy of 96.35%.

Table I
ACCURACY USING ONLY SINGLE FEATURES

Feature	Accuracy
EM	94.73
EM + Entropy term	94.72
Observed Count	93.44
Expectation Matching	93.28
SCC	91.01
Log likelihood Ratio	89.70

B. Precision and Recall

One practical question biologists may ask when assessing next generation sequencing data is to ask how to select observed sequences which are correct at a given confidence level.

We attempt to answer this question by using the posterior probability given by the SVM classifier. Figure 1 shows a histogram of the posterior probability of true sequences. Although most sequences have probabilities near zero or one, some have probabilities near 0.9, which should perhaps be removed if the downstream analysis is highly sensitive to false positives.

We then determine two evaluation measures for the user to select the reads. Let θ be the user-defined confidence threshold. The precision and recall are defined as:

$$\mathrm{Precision} = \frac{\#\text{true seqs with post. prob} \geq \theta}{\#\text{seqs with post. prob.} \geq \theta}$$

$$\mathrm{Recall} = \frac{\#\text{true seqs with post. prob.} \geq \theta}{\#\text{true seqs}}$$

Precision gives the fraction of sequences classified as true that really are true sequences, while recall gives a fraction of true sequences that are classified as true.

Figure 2 shows two plots of precision and recall using simulations based on two different datasets. Both use six

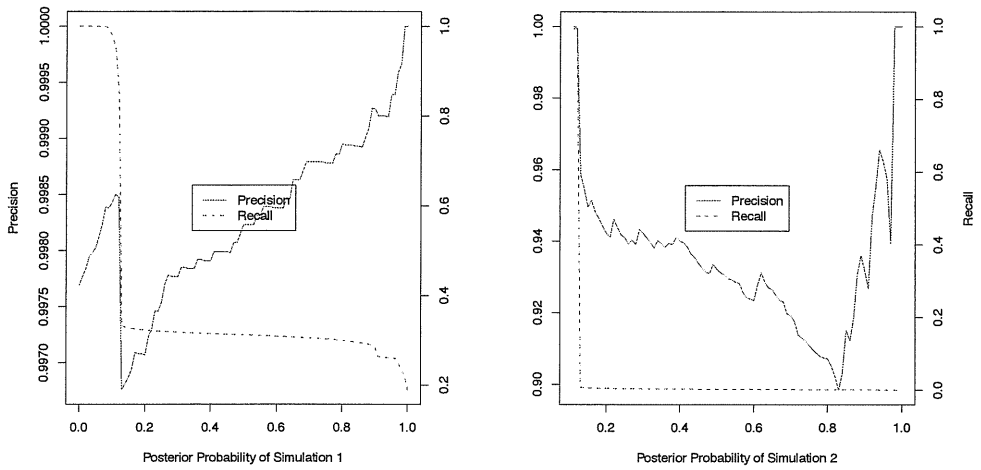

Figure 2. Precision and recall. The panel on the left (simulation 1) were generated using *Beta vulgaris* as dataset, and panel on the right (simulation 2) using a *Drosophila* dataset.

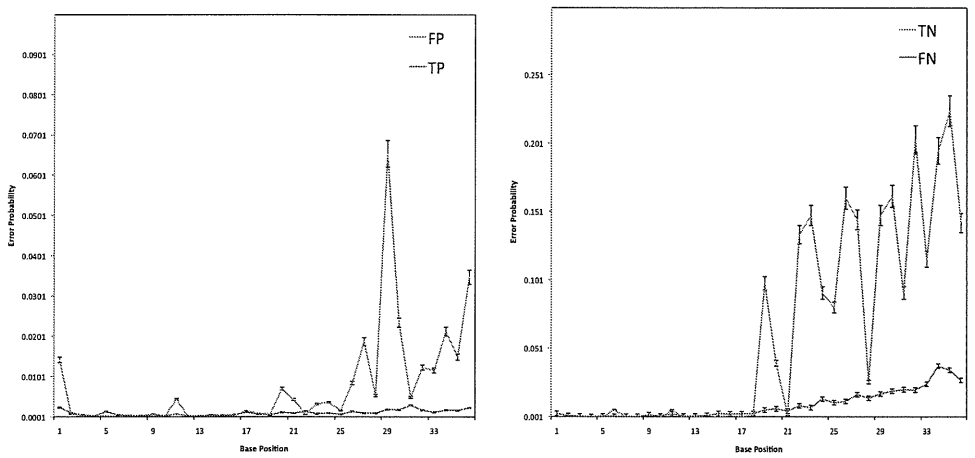

Figure 3. The average error probability by position (as given by the quality scores) is shown for correctly and incorrectly classified sequences

features for prediction. In these figures we can observe that by using posterior probability ≤ 0.12 the user can obtain reads up to 99.98% precision. Also with the same posterior probability threshold the user can obtain reads with at least 99.6% recall.

C. Computation Time

The running time for generating all features on 100K Solexa/Illumina reads is ~2.5 hours with ~20MB memory, using 2.9GHz, 32GB RAM Linux machine. The most time consuming step was the computation of the Log likelihood-Ratio, which required ~1.5 hours.

D. Characteristics of Misclassified Reads

Figure 3 shows the error probabilities per position of correctly and incorrectly classified sequences (for sequences called from multiple reads, their harmonic average is used). It is not surprising that the negative data (false positive and true negatives) have a higher probability of error than the positive data. Amongst the positive sequences, the ones with low error probability tend to be correctly predicted – they claim they are correct with high confidence and the classifier accepts them. On the other hand, amongst the negatives sequences the ones with high error probability tend to be correctly predicted – they admit they may be misreads and the classifier judges that they are.

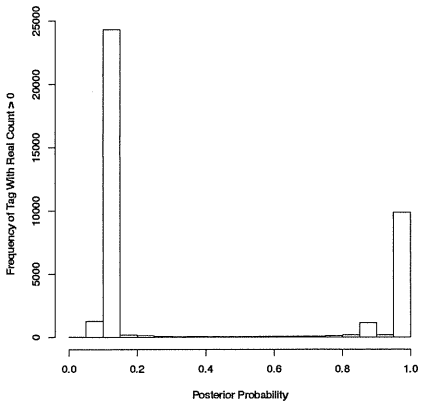

Figure 1. Posterior probability distribution of true sequences

In addition to the well known general trend for error probability to increase with position, there are some interesting and apparently non-random patterns regarding the position specific error rate [8], [11], [13]. Presumably this reflects details of the particular experimental sample we used to train our simulator.

V. Conclusion

In this paper we have demonstrated that identification of reliable sequences from next generation sequencing reads can be done accurately by machine learning classification techniques using observed count, estimate true count from error correction tool (RECOUNT), log likelihood with entropy penalty, log likelihood ratio, expectation matching score, and specific correction coefficient (SCC) as features. We also show that by using SVM posterior probability ~0.12 the biologist can select reads with high precision and recall.

One advantage of our approach is that it does not rely on a reference genome. This is useful because in some important cases a reliable reference genome is not available; e.g. metagenomic analysis, transcriptomic analysis of cancer cells (with possible chromosomal rearrangements), and analysis of highly polymorphous species. Also, even if we have a reliable reference genome, the sequences may not be easily alignable if the sequences have been spliced or otherwise processed. Our result should provide some aid in further analysis of next generation data such as assembly or mapping.

Acknowledgment

This work was supported by Grant-in-Aid for Scientific Research on Innovative Areas (221S002).

References

[1] Akmaev, V.R. and Wang, C. J, Correction of sequence-based artifacts in serial analysis of gene expression, *Bioinformatics*, 20:1254-1263, 2004.

[2] Beißbarth, T., *et al.*, Statistical modeling of sequencing errors in SAGE libraries, *Bioinformatics*, 20:i31-i39, 2004.

[3] Bianchetti, L. *et al.*, SAGETTARIUS: a program to reduce the number of tag mapped to multiple transcripts and to plan SAGE sequencing tags, *Nucleic Acids Research*, 35(18):e122, 2007.

[4] Colinge, J. and Feger, G. Detecting impact of sequencing errors on SAGE data, *Bioinformatics*, 17(9):840-842, 2001.

[5] Dempster, A., Laird, N., and Rubin, D. Maximum likelihood from incomplete data using the EM algorithm. *Journal of Royal Statistical Society*, (39): 1-38, 1977.

[6] Dohm, J. C, *et al.* Substantial biases in ultra-short read data sets from high-throughput DNA sequencing, *Nucleic Acids Research*, 36(16):e105, 2008.

[7] Ewing. B, and Green, P., Base-calling of automated sequencer traces using Phred II error probabilities, *Genome Research*, (8):186-194, 1998.

[8] Frith, M.C, Wan, R. and Horton, P. Incorporating sequence quality data into alignment improves DNA read mapping, *Nucleic Acids Research*, 38(7):e100.

[9] Hildebrand, B. F., *Introduction to Numerical Analysis: 2nd edition*, Dover Publications, 1987.

[10] Ghildiyal, M. *et al.* Endogenous siRNAs derived from transposons and mRNAs in Drosophila somatic cells, *Science*, (5879):1077-81, 2008.

[11] Qu, W., Hashimoto, S. and Morishita, S., Efficient frequency-based de novo short read clustering for error trimming in next-generation sequencing, *Genome Research*, (19):1309-1315, 2009.

[12] Yang X., Dorman K.S., Aluru S., Reptile: representative tiling for short read error correction, *Bioinformatics*, 26:2526-2533, 2010.

[13] Rougemont, J. *et al.*, Probabilistic base calling of Solexa sequencing data, *BMC Bioinformatics*, (9):431, 2008.

[14] Schröder J., Shröder H., Puglisi S., Sinha R., Schmidt B. SHREC: a short-read error correction method, *Bioinformatics*, 25:2157-2163, 2009.

[15] Vapnik, V. *The Nature of Statistical Learning Theory*, Springer, 1995.

[16] Velculescu, V.E. *et al.*, Analysis of human transcriptomes, *Nature Genetics*, (270):484-487, 1999.

[17] Weihs, C., Ligges, U., Luebke, K. and Raabe, N. klaR Analyzing German Business Cycles. In *Baier, D., Decker, R. and Schmidt-Thieme, L. (eds.). Data Analysis and Decision Support*, 335-343, Springer-Verlag, Berlin, 1996.

[18] Wijaya, E., Frith, M., Suzuki, Y., Horton, P., RECOUNT: Expectation maximization based error correction tool for next generation sequencing data, *Genome Informatics*,23:189-200, 2009.

OPEN ACCESS Freely available online

PLoS one

PeakRegressor Identifies Composite Sequence Motifs Responsible for STAT1 Binding Sites and Their Potential rSNPs

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

1 Computational Biology Research Center, Advanced Industrial Science and Technology (AIST), Tokyo, Japan, **2** Waseda University, Tokyo, Japan, **3** Mitsubishi Research Institute, Inc., Tokyo, Japan

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 RNA Polymerase II with correlation coefficients as high as 0.65 and 0.66, respectively. Using PeakRegressor, we could identify composite motifs for STAT1, as well as potential regulatory SNPs (rSNPs) involved in the regulation of transcription levels of neighboring genes. In addition, we show that among five regression methods, L1-norm log linear regression achieves the best performance with respect to binding motif identification, biological interpretability and computational efficiency.

Citation: Pessiot J-F, Chiba H, Hyakkoku H, Taniguchi T, Fujibuchi W (2010) PeakRegressor Identifies Composite Sequence Motifs Responsible for STAT1 Binding Sites and Their Potential rSNPs. PLoS ONE 5(8): e11881. doi:10.1371/journal.pone.0011881

Editor: Xiaolin Wu, National Cancer Institute at Frederick, United States of America

Received January 15, 2010; **Accepted** June 7, 2010; **Published** August 27, 2010

Funding: This work was supported by general subsidies from the National Institute of Advanced Industrial Science and Technology, Japan. This funder had no role in study design, data collection and analysis, decision to publish, or preparation of the manuscript. Takeaki Taniguchi is employed by the Mitsubishi Research Institute, Inc. His wages were funded by the company and he participated in performing the computational experiments. This funder also had no role in study design, data collection and analysis, decision to publish, or preparation of the manuscript.

Competing Interests: Takeaki Taniguchi is employed by the Mitsubishi Research Institute, Inc. There are no patents, products in development, or marketed products related to this research, and his involvement does not alter the adherence to all the PLoS ONE policies on sharing data and materials.

* E-mail: w.fujibuchi@aist.go.jp

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 [1,2]. 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. [3] 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 approximate binding affinity [4,5].

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 at performing peak signal regression based on candidate motif models. Because PeakRegressor is computationally efficient and the models are easy to interpret, it is usable with large-scale datasets. We apply PeakRegressor to two ChIP-Seq datasets and show its ability to recover motifs involved in the binding of STAT1 and RNA Polymerase II.

Results and Discussion

Results with PeakRegressor

Table 1 shows the correlation coefficients between the peak scores and their predicted values by PeakRegressor in the test dataset. We keep the highest correlation coefficient among various β for each iteration of the 30-fold cross-validation, and those 30 correlation coefficients are averaged and shown here. Obviously, the filtering with peak existence probability, i.e., Q-value, over the control experiment enhances the regressions. The filtering with promoter region proximity improves the regressions of RNA Polymerase II but not of STAT1.

In Figure 1, we plot the STAT1 peak scores with two filtering methods such as Q-value $<10^{-3}$ and promoter proximity in the test dataset against 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 existed only in a biased region, leading to worse prediction.

In Tables 2 and 3, we show the top ten motifs for STAT1 and RNA Polymerase 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,

Table 1. Influence of the peak filtering methods on the correlation coefficients between peak values and their predicted values in the test dataset.

Filtering method	*#Peaks (STAT1/Pol II)*	*STAT1*	*Pol II*
None	36998/24739	0.50	0.44
Promoter proximity	3,907/9,094	0.41	0.53
Q-value $<10^{-3}$	16639/17580	**0.65**	**0.66**

The correlation coefficients are averaged in 30-fold cross-validation.
doi:10.1371/journal.pone.0011881.t001

it is clear that our approach correctly identifies the classical GAS motif TTC[TC]N[GA]GAA as the main binding motif [6]. Meanwhile, the RNA Polymerase II binding motifs also contain known Downstream Promoter Element [AG]G[AT][CT][GAC] and Initiator Site [TC][TC]AN[TA][TC][TC] [7].

STAT1 composite motifs. As the most important feature of PeakRegressor, it can give us a list of 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 are responsible for STAT1 binding signals:

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

Comparison with other regression methods

PeakRegressor identifies potential TFBMs by solving a regression problem. This regression problem is defined by a set of peak vectors $\{\mathbf{x}_i\}_{i=1...N}$ and their corresponding peak scores $\{y_i\}_{i=1...N}$. The goal is to predict the peak scores from the peak vectors. The fitted regression model is then used to infer the TFBM candidates. We expect the regression method to have three properties. First, it should identify the true binding motifs. Second, it should identify the strengthening and weakening motifs. Third, it should be computationally efficient in order to cope with large ChIP-Seq datasets.

In PeakRegressor, we choose to use the L1-norm log linear regression to solve this problem. This approach favors sparse solutions (i.e., solutions with a small number of motifs) and therefore, we argue that it is more suitable for the TFBM identification problem. However, many other regression methods are available and can be used to solve the regression problem. How do these approaches compare with the L1-norm log linear regression with respect to the desired properties? In the following, we compare our L1-norm log linear regression based approach with other regression methods: linear least squares regression, ridge regression, partial least squares regression, and principal component regression. For each method, we evaluate its performance on the STAT1 and RNA Polymerase II datasets and discuss the results.

Linear least squares regression. In Tables 4 and 5, we show the top ten motifs identified by the linear least squares regression. In the case of STAT1 (Table 4), we can see that the true GAS motif appears within the top ten motifs. However, two problems appear. First, the regression coefficients of the GAS

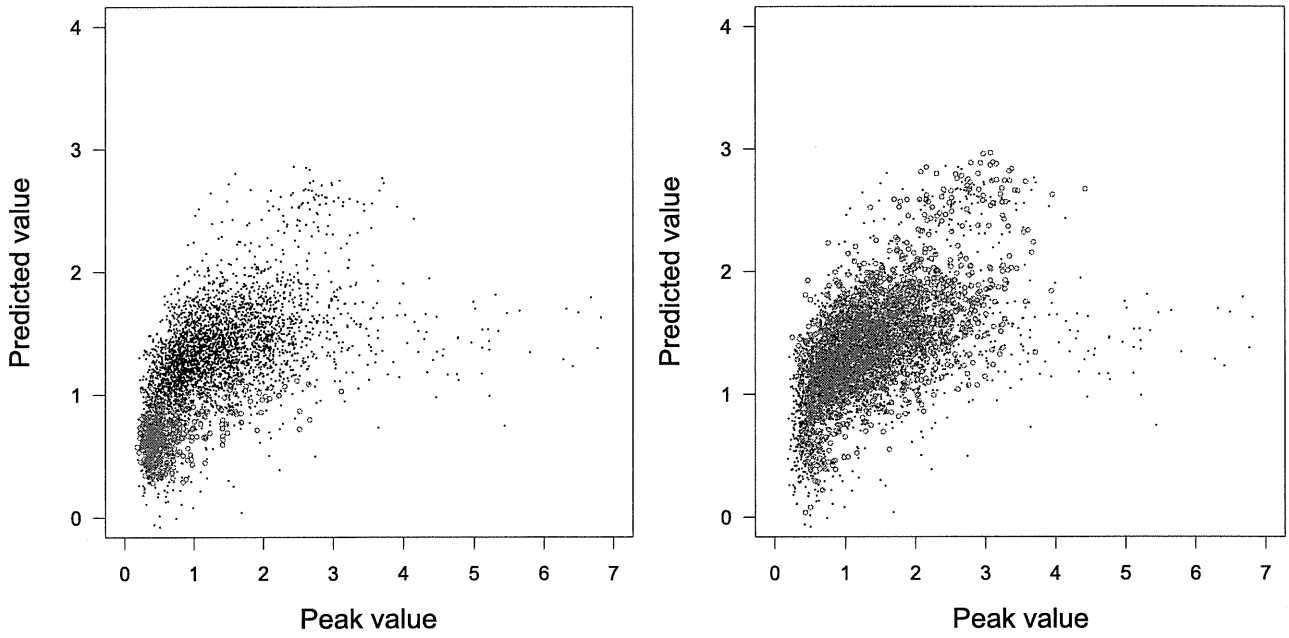

Figure 1. STAT1 regression results with two filtering methods: Q-value (right) and promoter proximity (left). The correlation coefficients on the test data between peak values and their predicted values are 0.65 and 0.41 for Q-value and promoter proximity filterings, respectively.
doi:10.1371/journal.pone.0011881.g001

Table 2. List of putative STAT1 binding motifs identified by PeakRegressor.

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

The classical GAS motifs are shown in boldface.
doi:10.1371/journal.pone.0011881.t002

motif are very low compared to those of the top motifs (between 0.01 and 0.02). This means that according to the linear least squares regression, the true GAS motif has only a minor effect on the binding, which contradicts existing biological knowledge. Second, the most important motifs according to the linear least squares regression are CCCCTCCC and CCCCACCC. However, each of them is associated with opposite coefficients (−1.0 and 0.94 for CCCCTCCC, 0.34 and −0.34 for CCCCACCC). Therefore, each of them is considered to have both a strengthening effect and a weakening effect on the binding, which is a contradictory result.

With the RNA Polymerase II dataset (Table 5), linear least squares regression is able to identify the initiator site and the downstream promoter element. However, the instances of the initiator site have opposite coefficients ([CA]CAGACT with 0.62, T[CT][TA]T[TG][AC][AT] with 0.62, and TT[TAC]TTT[CT] with −0.61). As they are instances of the same motif, we expect them to have the same sign i.e., to have the same effect on the binding. In summary, for both STAT1 and RNA Polymerase II datasets, the results of the linear least squares regression are difficult to interpret biologically. This is a typical situation where we would like to reduce the number of motifs used by the regression model. Clearly, this is not possible with the linear least squares regression approach.

Table 3. List of putative RNA Polymerase II binding motifs identified by PeakRegressor.

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

The known Downstream Promoter Element and Initiator site motifs are shown in boldface.
doi:10.1371/journal.pone.0011881.t003

Table 4. List of putative STAT1 binding motifs identified by linear least squares regression.

STAT1	*Normalized coef.*
CCCCTCCC	−1.0
CCCCTCCC	0.94
CCCCACCC	0.34
CCCCACCC	−0.34
CA[TC]GTGACT[TG]C	0.02
[TG]G[GTA][GC][AG]T**TT[CA]C[AGC] [GA]GAA**[AC][TG]G[GA][GC]	0.02
[CT][TC]CA[GT]**TTCCAGGAA**[AT]T[CG][CAT]C[CT]	0.01
GGAGGGCG	−0.01
TTC[CT][TG][GA]GAAAT[GC][CA][CA] [CAT][AT][TCG][CG][CT]	0.01
A[CT]**TTC[TC][TG]GGAA**	0.01

The classical GAS motifs are shown in boldface.
doi:10.1371/journal.pone.0011881.t004

Ridge regression. In Tables 6 and 7, we show the top ten motifs identified by the ridge regression. In the case of STAT1 (Table 6), we can see that the ridge regression and the L1-norm log linear regression identify very similar motifs. In both cases, the classical GAS motif is clearly identified as the main binding motif. Both regression methods also identify CA[TC]GTGACT[TG]C as a strengthening motif and GGAGGGCG as a weakening motif. In the case of RNA Polymerase II (Table 7), both methods are able to identify the initiator site (T[CT][TA]T[TG][AC][AT) and the downstream promoter element (A[GC][TAG]CA).

However, they differ greatly with respect to computational complexity. In [8], the authors present an algorithm for computing the L1-norm log linear regression solutions of many regularization parameters for the same computational cost as that of a single least

Table 5. List of putative RNA Polymerase II binding motifs identified by linear least squares regression.

RNA Polymerase II	*Normalized coef.*
T[AG]**A[GC][TAG]CA**[GCT]A[AC]AA	1.0
A[GA]AA[AC][CA]AA[AC]AAA	0.86
C[ACT]**[GT][CG][CT][TA]C**C[AGT]CC[TA]	0.81
C[CT][CG][AT]GGCTGG[AG]G	0.74
TTTCTGC[CT][CT]TT[GT]	0.74
GGAGGGAGGC[AG]G	0.69
AC[AC][CA][AC][AT][AG]AGAAA	0.64
T[TA]T[TC]**[CA]CAGACT**[AT]	0.62
TTTG**T[CT][TA]T[TG][AC][AT]**T	0.62
TT[TAC]TTT[CT]TT[TC]TT	−0.61

The known Downstream Promoter Element and Initiator site motifs are shown in boldface.
doi:10.1371/journal.pone.0011881.t005

Table 6. List of putative STAT1 binding motifs identified by ridge regression.

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]	0.89
GGAGGGCG	−0.69
[CT][TC]CA[GT]**TTCCAGGAA**[AT]T[CG][CAT]C[CT]	0.69
A[CT]**TTC[TC][TG]GGAA**	0.68
TTC[CT][TG][GA]GAAAT[GC][CA][CA] [CAT][AT][TCG][CG][CT]	0.65
TT[CA]C[TAG][GA]GAA[GA]T	0.59
TT[CA][TC][GA]GGAA[AG]	0.58
GGACGCCG	−0.57
G[TGC][CGT][AT][TG]**TTCC[TCA][GA][GT]AA**[AG]	0.53

The classical GAS motifs are shown in boldface.
doi:10.1371/journal.pone.0011881.t006

squares fit. As a consequence, using the same STAT1 dataset, a 30-fold cross-validation takes approximately 60 hours with the ridge regression, while it takes only 2.5 hours with the L1-norm log linear regression (i.e., 24 times faster). In summary, although both methods show very similar results with respect to binding motif identification, the ridge regression is slower and more difficult to use with large ChIP-Seq datasets than the L1-norm log linear regression.

Partial least squares regression and principal component regression. In Tables 8 and 9, we show the top ten motifs for STAT1 identified by the partial least squares regression and the principal component regression. We can see that both methods are able to identify the classical GAS motif. In Table 8, the partial least squares regression shows very similar results to the L1-norm log linear regression as both methods identify CA[TC]GTGACT-[TG]C as a strengthening motif and GGAGGGCG as a weakening motif. In Table 9, the principal component regression identifies only the GAS motif and fails to identify any other motifs involved in the binding. In the case of RNA Polymerase II, both partial least

Table 7. List of putative RNA Polymerase II binding motifs identified by ridge regression.

RNA Polymerase II	*Normalized coef.*
T[AG]**A[GC][TAG]CA**[GCT]A[AC]AA	1.0
A[GA]AA[AC][CA]AA[AC]AAA	0.86
C[ACT]**[GT][CG][CT][TA]C**C[AGT]CC[TA]	0.81
C[CT][CG][AT]GGCTGG[AG]G	0.75
TTTCTGC[CT][CT]TT[GT]	0.74
GGAGGGAGGC[AG]G	0.70
AC[AC][CA][AC][AT][AG]AGAAA	0.65
T[TA]T[TC]**[CA]CAGAC**T[AT]	0.63
TTTGT**[CT][TA]T[TG][AC][AT]**T	0.62
TT[TAC]TTT[CT]TT[TC]TT	0.61

The known Downstream Promoter Element and Initiator site motifs are shown in boldface.
doi:10.1371/journal.pone.0011881.t007

Table 8. List of putative STAT1 binding motifs identified by partial least squares regression.

STAT1	*Normalized coef.*
CA[TC]GTGACT[TG]C	1.0
[TG]G[GTA][GC][AG]T**TT[CA]C[AGC] [GA]GAA**[AC][TG]G[GA][GC]	0.80
TTC[CT][TG][GA]GAAAT[GC][CA] [CA][CAT][AT][TCG][CG][CT]	0.58
[CT][TC]CA[GT]**TTCCAGGAA**[AT]T[CG][CAT]C[CT]	0.56
[GA][AG]A[AG][AT][CTG][CA]A[GT][CG]T[GT][CG] [CA]T[TG][CT][CGT]T	0.50
TCACA[TG]G[ACG]	0.42
GGAGGGCG	−0.41
G[TGC][CGT][AT][TG]**TTCC[TCA][GA][GT]AA**[AG]	0.41
TT[CA]C[TAG][GA]GAA[GA]T	0.40
A[TA]**TTCC[CT][GA]GAA**[AC]T[CG][AC]	0.39

The classical GAS motifs are shown in boldface.
doi:10.1371/journal.pone.0011881.t008

squares regression (Table 10) and principal component regression (Table 11) are able to identify the initiator site and the downstream promoter element.

However, the results of the partial least squares regression and the principal component regression are difficult to interpret. In the former (Table 10), different instances of the downstream promoter element have positive or negative coefficients (T[TG]AACA-**CAGTT[TA]** with 1.0, [CT][CG]AG**A[GA]TCC**A[GA][CG] with −0.90, and A[AG][GA][AG]**GGA[GCA]G**A[GA]A with 0.87). As they are instances of the same motif, we expect them to have the same sign, i.e., to have the same effect on the binding. In the latter (Table 11), all the instances of the initiator site and the downstream promoter element have negative coefficients. However, these motifs should strengthen the binding and therefore, we expect their coefficients to be positive.

Table 9. List of putative STAT1 binding motifs identified by principal component regression.

STAT1	*Normalized coef.*
[TAC]**TTCC[CA][GA][GT]AA**[AG][TA]C	1.0
T**TTCC[CT][GA]GAA**AA[CT]TC[AC]TGAA	0.94
TT**TTC[CT][AG]GGAA**[AG][GT]GG[CG][TCA][GA]GG	0.87
T**TTC[TC][TG][GA][GAT]AA**[GA]	0.86
[TC]**TTCC[AC][AG]G[AC]A**	0.85
[GA]GAACC[TC][TG]CAG**TTC[CT][AG]GGAA**	0.82
CC[CTA][CGT]T**TTC[CT]T[GA]GAA**[AG][ACT][CG]	0.82
TTC[CT][TG][GA]GAAAT[GC][CA][CA][CAT]-[AT][TCG][CG][CT]	0.81
T**TTC[CT][AGT]GGAA**A[TG][GA][GA]G[TAC][GA]G	0.80
G[CT]**TT[CA][CT][GAT][GA]GAA**[AG][TG][AGC]-[GA][GCA][TGA]A[CG]	0.78

The classical GAS motifs are shown in boldface.
doi:10.1371/journal.pone.0011881.t009

Table 10. List of putative RNA Polymerase II binding motifs identified by partial least squares regression.

RNA Polymerase II	*Normalized coef.*
T[TG]AACAC**AGTT[TA]**	1.0
C[CT][CG][AT]GGCTGG[AG]G	0.99
G[AG]GG[CG]CCAGAGA	−0.97
[CT][CG]AG**A[GA]TCC**A[GA][CG]	−0.90
CTGG[AC]GCTG[TG][TC][ACG]	−0.89
A[AG][GA][AG]**GGA[GCA]G**A[GA]A	0.87
[CG][AT][CT]T**[GC]C[AT][CG]TCC**[AC]	0.86
GGAGGGAGGC[AG]G	0.86
A[GA]AA[AC][CA]AA[AC]AAA	0.85
[GT]GCCCAGG[CG][TG][GA]G	−0.81

The known Downstream Promoter Element and Initiator site motifs are shown in boldface.

doi:10.1371/journal.pone.0011881.t010

The lack of interpretability of the partial least squares regression and the principal component regression lies in the fact that the regression is performed in a low-dimensional feature space. In the original motif space, the vector representation of the peak sequences has a meaning and each component of a vector measures how similar a motif is to a peak sequence. However, in the low-dimensional feature space computed by the partial least squares regression and the principal component regression, the vector components lose their biological meaning. From the computational complexity perspective, we also mention that both methods are very slow. Using the STAT1 dataset, a 30-fold cross-validation of the partial least squares regression with 10 components takes approximately 240 hours. In summary, the partial least squares regression and the principal component regression are able to identify the classical GAS motif for STAT1 and the initiator site and the downstream promoter element for RNA Polymerase II. However, the results are difficult to interpret biologically and do not allow identification of strengthening or weakening motifs. In addition, they are too slow to be used with large ChIP-Seq datasets.

Table 11. List of putative RNA Polymerase II binding motifs identified by principal component regression.

RNA Polymerase II	*Normalized coef.*
GCT**GG[GT][AC][CT]**[CT]ACA	−1.0
[CG]GCGGCGGCGGC	0.97
GCCCAGGCTG[CG][TA]	−0.96
CA[AC]**AG[TG][GC]**CTG[GA]G	−0.94
CTGG[TC][CT]TCAAA[GC]	−0.90
CTGG[AG]G[TG]GC[AT]G[TG]	−0.89
CTGG**A[GA]T[GT]C**A[GA][TG]	−0.87
[TC]CCA[CA]AG[CAT][AG]CTG	−0.86
[TA]C[AC]T[GA][CG]CCTGT[GT]	−0.84
[CA]TG[AT]CCACAGA[AT]	−0.83

The known Downstream Promoter Element and Initiator site motifs are shown in boldface.

doi:10.1371/journal.pone.0011881.t011

Advantages of L1-norm log linear regression over other methods for TFBM identification. We considered the following regression methods for TFBM identification: L1-norm log linear regression, linear least squares regression, ridge regression, partial least squares regression, and principal component regression. In Table 12, we summarize the correlation coefficients averaged on the test sets. As we can see, all regression methods demonstrate similar performance and are able to identify the classical GAS motif for STAT1 and the initiator site and the downstream promoter element for RNA Polymerase II.

However, they exhibit marked differences with respect to biological interpretability and computational efficiency. The results of the linear least squares regression, the partial least squares regression, and the principal component regression do not allow identification of strengthening or weakening motifs. Therefore, they are difficult to use for binding motif identification. Both L1-norm log linear regression and ridge regression solve this problem by means of regularization. However, the ridge regression is very slow compared to the L1-norm log linear regression. Therefore, the ridge regression is difficult to use with large-scale ChIP-Seq datasets. In summary, the L1-norm log linear regression is the only method that can achieve all the desired goals for our task; it identifies the transcription factor binding motifs, the regression coefficients are easy to interpret biologically, and its implementation with the LASSO algorithm is fast and efficient. This justifies our choice of the L1-norm log linear regression in PeakRegressor.

Parameter setting

The performance of PeakRegressor depends on the choice of parameters that have to be set empirically. In this section, we explain how we choose two important parameters: the length of peak sequences and the number of motif candidates.

Length of peak sequences. In the dataset provided by [9], all the peaks correspond to various DNA sequences. These sequences have different lengths, ranging from 1 bp to several thousand bp. To conduct our analysis, we modify the peak sequences in the following way:

- We shorten long peak sequences for two reasons. First, when using long DNA sequences, the computations of the motif finding algorithm MEME take too much time. Second, finding good quality motifs with MEME is easier with short DNA sequences than with long ones.
- We widen short peak sequences. Due to the noisy nature of ChIP-Seq data, the motifs we are looking for may not be exactly on the provided peak sequence, but in the surrounding DNA neighborhood. Therefore, we decide to choose a uniform length for all the peak sequences. The choice of 200 bp is empirical; we try several values (100 bp, 200 bp, 400 bp, and 800 bp) and consider the one

Table 12. Different regression methods and their correlation coefficients averaged on the test sets.

Regression method	*STAT1 correlation coef.*	*Pol II correlation coef.*
L1-norm log linear regression	**0.65**	**0.66**
Linear least squares regression	0.64	0.64
Ridge regression	0.64	0.64
Partial least squares regression	0.64	0.65
Principal component regression	0.63	0.52

doi:10.1371/journal.pone.0011881.t012

that achieves the best performance, i.e., the highest correlation coefficients (results not shown for other peak lengths).

Number of motif candidates. In the first step of PeakRegressor, we use MEME to find over-represented DNA motifs in the peak sequences. This step results in 800 motif candidates for STAT1 and 880 for RNA Polymerase II. Given the large number of motif candidates, we empirically observe the presence of similar motifs in the set of motif candidates. We may wonder if this redundancy could affect the prediction performance of PeakRegressor. However, we show that this is not the case.

PeakRegressor uses a regression method called L1-norm log linear regression. In contrast with other regression methods, L1-norm log linear regression achieves its best prediction performance by removing redundant or uninformative motifs from the regression model. Therefore, the removal of redundant motifs is automatically performed when using L1-norm log linear regression. Table 2 shows the set of motifs that achieve the best correlation coefficient for STAT1. We can see that some motifs are similar. For example, the motifs A[CT]**TTC[TC][TG]GGAA, TT[CA]C[TAG][GA]GAA** [GA]T, A[TA]**TTCC[CT][GA]GAA**[AC]T[CG][AC], and **TT-[CA][TC][GA]GGAA**[AG] are short, similar motifs containing the STAT1 binding motif. In other experiments, we find that the prediction performance worsens when similar motifs are removed (results not shown). Hence, although the motifs appear similar and redundant, they actually contain complementary information for the prediction performance.

Moreover, the motif weights computed by PeakRegressor are all different (resp. 0.56, 0.55, 0.48, and 0.47). Hence, while other approaches, such as motif clustering, would consider all these motifs to be equally important, PeakRegressor is able to detect the relative importance of each motif and compute the corresponding weight. This is explained by the noisy nature of the DNA motifs found by MEME in step 1. For a given binding motif, PeakRegressor needs to use all the noisy PSSM approximations to achieve the best prediction performance. This is an important property of PeakRegressor, especially when the number of noisy motifs is very large.

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 neighboring gene transcription will be perturbed, thus leading to genetic diseases in some cases [10]. 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 (http://www.ncbi.nlm.nih.gov/SNP/). We find that 0.36% (147 for 40,395 bp) of mapped positions with 10 STAT1 motifs in Table 2 on the peak sequences contains SNPs, while as much as 0.53% (17,852 for 3,344,439 bp) of all positions contains SNPs on the peak sequences. The statistical difference between the above two ratios is highly significant such as $p<3.7^{-7}$ according to the hypergeometric distribution. 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.

Materials and Methods

PeakRegressor

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. In contrast with previous approaches, PeakRegressor uses the peak scores (provided by [9]) as a surrogate for the binding affinities. We argue that the peak scores provide more accurate approximations of the binding affinities than the methods based on transcription factor occupancy [4,5]. Therefore, using the peak scores lead to better identification of functional TFBMs. In addition, PeakRegressor identifies not only primary TFBM candidates but also secondary motifs that may often synergistically strengthen or weaken the binding. The workflow is summarized in Figure 2.

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 (http://meme.sdsc.edu/) 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 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.

Next, we want all the x_{id} to be positive for interpretability purpose. So we simply shift their values by substracting the lowest component: $x_{id} \leftarrow x_{id} - a$, where a is the minimum value of the original x_{id}. Finally, we normalize each data vector by dividing it with its euclidean norm: $x_i \leftarrow x_i / \|x_i\|^2$.

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:

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

where $\beta > 0$ is a user-defined regularization coefficient. The L1-norm log linear regression is able to remove redundant or uninformative features, and to select a small number of features that best explain the fitted quantity [11]. In our case, the features correspond to DNA motifs and hence, the result of this step is a set

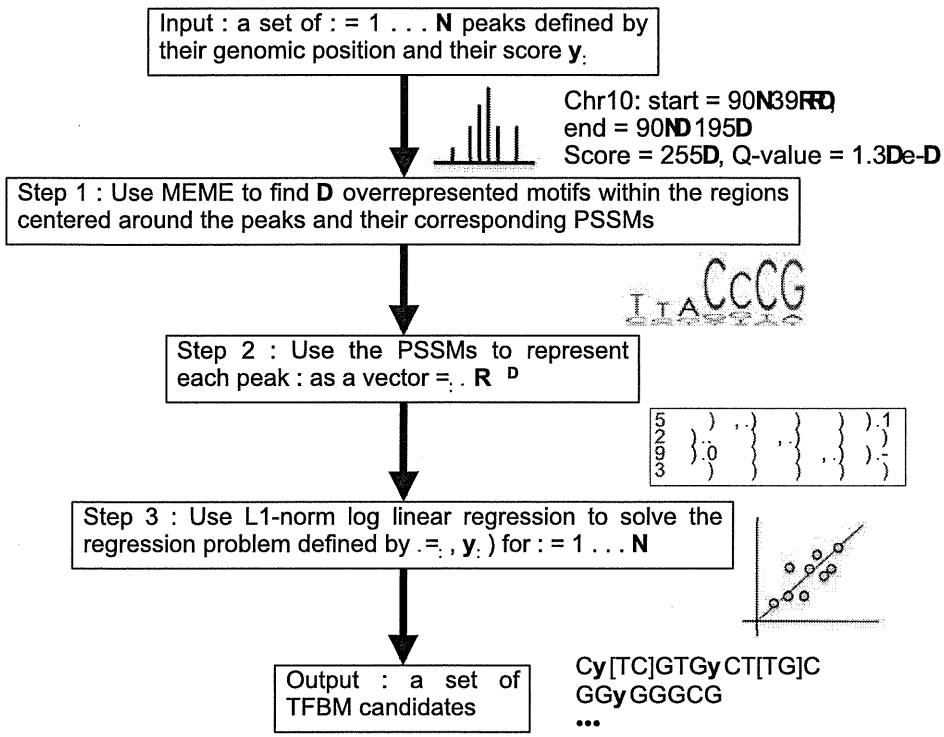

Figure 2. Schematic view of the workflow of PeakRegressor. PeakRegressor takes ChIP-Seq data as input and outputs a list of TFBM candidates and their weights that give the best regression accuracies.
doi:10.1371/journal.pone.0011881.g002

of motifs that best explain the binding signal values from ChIP-Seq dataset. We use Lasso, a popular algorithm for solving L1-norm log linear regression. Lasso is available as part of the LARS package for R (http://www-stat.stanford.edu/~hastie/Papers/LARS/).

Other regression methods

In this section, we present alternatives to the L1-norm log linear regression: linear least squares regression, ridge regression, partial least squares regression, and principal component regression. All these regression methods are used in the following way. Once a regression model is fitted to the peak dataset, we rank the regression coefficients with respect to their absolute values. Using this ranking, the top motifs are the potential TFBMs.

Linear least squares regression. The linear least squares regression is the simplest regression approach. It fits a linear model to the dataset by minimizing the sum of squared errors $\sum_{i=1}^{N}(y_i - b^T x_i)$. Its difference with the L1-norm log linear regression (equation 1) is the absence of a regularization term. Therefore, the linear least squares regression is more prone to overfitting when the regression problem contains more dimensions than samples.

Ridge regression. The ridge regression [12] minimizes $\|b\|^2 + \sum_{i=1}^{N}(y_i - b^T x_i)$, where the regularization term is $\|b\|^2 = \sum_{d=1}^{D} b_d^2$, i.e., the Euclidean norm of b. It is quite similar to the L1-norm log linear regression, and their main difference lies in the regularization term. The ridge regression seeks a solution with a low Euclidean norm. Although the Euclidean norm is a protection against overfitting, it does not favor sparse solutions (i.e., solutions with many motifs) as the L1-norm log linear regression does [11].

Partial least squares regression and principal component regression. The partial least squares regression [13] and the principal component regression are two approaches of the same idea; they perform linear regression using the low-dimensional data matrix Z instead of the initial data matrix X. This approach avoids overfitting problems. Therefore, the partial least squares regression and the principal component regression have been widely used in problems containing several dimensions (i.e., motifs) and few samples (i.e., peaks).

In the principal component regression, the low-dimensional data matrix Z contains the most information about the initial data matrix X (according to the singular value decomposition of X). In the partial least squares regression, the low-dimensional data matrix Z is calculated using both the initial data matrix X and the peak score vector y. In both cases, linear regression is performed using Z instead of the initial data matrix X. Both partial least squares regression and principal component regression are available as part of the PLS package for R (http://mevik.net/work/software/pls.html). Once the regression coefficients have been computed in the low-dimensional space, they are mapped back in the original motif space. Then, these coefficients can be used to identify potential binding motifs.

Input ChIP-Seq datasets

The ChIP-Seq dataset we used is provided by [9] and is publicly available (http://www.camda2009.org/). The dataset provides various information about each peak, including the peak score, the peak center (for STAT1), and the Q-value that reflects the significance of the peak. The Q-values are derived from the P-values. First, they compute the P-values that reflect the significance of peak enrichment in the number of DNA tags, compared to control samples. These P-values are computed using the binomial distribution. Then, to account for multiple hypothesis testing, the Q-values are derived from the P-values. See [9] for more details.

For STAT1, we use 200-bp windows around the peak centers to define the peak sequences. For RNA Polymerase 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 RNA Polymerase II peak sequence lengths lie between 200 and 4000 bp.

Evaluation of prediction performance

PeakRegressor predicts the peak scores and therefore, we have two different values for each peak. The "true" peak score is the score provided by [9], and is derived from the frequency of reads of ChIP-Seq data. The predicted score is computed by PeakRegressor using the peak sequence information. Ideally, the predicted score should be equal to the true score. We use correlation coefficients to evaluate the prediction quality of PeakRegressor.

Experimental protocol

For L1-norm log linear regression and ridge regression, 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-fold 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 results of L1-norm log linear regression and ridge regression are averaged over the 30-fold cross-validation.

For partial least squares regression and principal component regression, the experiments were limited by the slowness of both methods. First we have to set the number of components K used for regression. We tried $K = 1...10$, and performed a 30-fold cross-validation for each value of K. In each fold, we split the dataset into 50% for training and 50% for testing. All the results of partial least squares regression and principal component regression are averaged over the 30-fold cross-validation.

Acknowledgments

The authors thank the anonymous CAMDA reviewers for their helpful comments.

Author Contributions

Conceived and designed the experiments: JFP WF. Performed the experiments: JFP HH TT. Analyzed the data: JFP HC WF. Wrote the paper: JFP HC WF.

References

1. Bussemaker HJ, Li H, Siggia ED (2001) Regulatory element detection using correlation with expression. In: RECOMB '01: Proceedings of the fifth annual international conference on Computational biology. New York, NY, USA: ACM. 86 p. doi:http://doi.acm.org/10.1145/369133.369174.
2. Conlon EM, Liu XS, Lieb JD, Liu JS (2003) Integrating regulatory motif discovery and genome-wide expression analysis. PNAS.
3. Das D, Pellegrini M, Gray JW (2009) A primer on regression methods for decoding cis-regulatory logic. PLoS Comput Biol 5: e1000269.
4. Foat BC, Morozov AV, Bussemaker HJ (2006) Statistical mechanical modeling of genome-wide transcription factor occupancy data by matrixreduce. Bioinformatics 22: e141–e149.
5. Gao F, Foat BC, Bussemaker HJ (2004) Defining transcriptional networks through integrative modeling of mrna expression and transcription factor binding data. BMC Bioinformatics.
6. Robertson G, Hirst M, Bainbridge M, Bilenky M, Zhao Y, et al. (2007) Genome-wide profiles of stat1 dna association using chromatin immunoprecipitation and massively parallel sequencing. Nat Meth 4: 651–657.
7. Butler JE, Kadonaga JT (2002) The rna polymerase ii core promoter: a key component in the regulation of gene expression. Genes Dev 16: 2583–2592.
8. Efron B, Hastie T, Johnstone I, Tibshirani R (2004) Least angle regression. The Annals of Statistics.
9. 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 Biotech 27: 66–75.
10. Ameur A, Rada-Iglesias A, Komorowski J, Wadelius C (2009) Identification of candidate regulatory snps by combination of transcription-factor-binding site prediction, snp genotyping and haplochip. Nucleic acids research 37.
11. Tibshirani R (1996) Regression shrinkage and selection via the lasso. J Roy Statist Soc Ser B 58: 267–288.
12. Bishop CM (2006) Pattern Recognition and Machine Learning (Information Science and Statistics). Secaucus, NJ, USA: Springer-Verlag New York, Inc.
13. Frank IE, Friedman JH (1993) A statistical view of some chemometrics regression tools. Technometrics.

Anal. Chem. **2010**, *82*, 5768–5776

Sensitive and Convenient Yeast Reporter Assay for High-Throughput Analysis by Using a Secretory Luciferase from *Cypridina noctiluca*

Yuki Tochigi,[†] Natsuko Sato,[†] Takehiko Sahara,[†] Chun Wu,[‡] Shinya Saito,[§] Tsutomu Irie,[§] Wataru Fujibuchi,[∥] Takako Goda,[†] Ryoichi Yamaji,[○] Masahiro Ogawa,[○] Yoshihiro Ohmiya,[◆,⊥] and Satoru Ohgiya*[,†,▽]

Bioproduction Research Institute, National Institute of Advanced Industrial Science and Technology (AIST), 2-17-2-1 Tsukisamu-Higashi, Toyohira-ku, Sapporo 062-8517, Japan, Bioproduction Research Institute, National Institute of Advanced Industrial Science and Technology (AIST), Tsukuba Central 6, 1-1-1 Higashi, Tsukuba 305-8566, Japan, Health Research Institute, National Institute of Advanced Industrial Science and Technology (AIST), 1-8-31 Midorigaoka, Ikeda 563-8577, Japan, ATTO Corporation, 1-5-32 Yushima, Bunkyo-ku, Tokyo 113-0034, Japan, Computational Biology Research Center, National Institute of Advanced Industrial Science and Technology (AIST), 2-4-7 Aomi, Koto-ku, Tokyo 135-0064, Japan, and Laboratory for Nutrition Chemistry, Division of Applied Life Sciences, Graduate School of Life and Environmental Sciences, Osaka Prefecture University, 1-1 Gakuen-cho, Naka-ku, Sakai, Osaka 599-8531, Japan

The yeast reporter assay has been widely used in various applications such as detection of endocrine disruptors and analysis of protein−protein interactions by the yeast two-hybrid system. The molecular characteristics of the reporter enzyme are critical determinants for this assay. We herein report the establishment of a novel yeast reporter assay using a secretory luciferase, *Cypridina noctiluca* luciferase (CLuc), as an alternative to the conventional β-galactosidase. The CLuc reporter assay in yeast is more sensitive and convenient than the conventional assay. A yeast high-throughput reporter assay was established with a laboratory automation system, and the transcriptional activity of hundreds of yeast promoter fragments was comprehensively determined. Our results indicate that the yeast CLuc reporter assay is a promising tool for large-scale and sensitive analysis in the development of new drugs and in various fields of biotechnology research.

The yeast reporter assay has been widely used in various applications, including determination of promoter activities,[1] identification of *cis*-elements,[2−4] analysis of chemicals (i.e., yeast estrogen screen (YES) assay),[5,6] and detection of protein−protein interactions (two-hybrid system).[7] Relative to mammalian cultured cell-based reporter assays, the yeast reporter assay is suitable for analyzing the effects of chemicals and for the two-hybrid system because of its high performance and low cost for large-scale screening. For the yeast reporter system, the concentrations of chemicals or the binding affinities of interacting proteins are determined by the amount of protein or the activity of reporter enzyme produced. The gene for the reporter enzyme is located downstream of a promoter specifically designed for the phenomena. Therefore, the characteristics of the reporter enzyme are critical for the performance of the reporter assay.

The *Escherichia coli* enzyme β-galactosidase (β-Gal) is the most popular reporter enzyme used in the yeast reporter assays.[8] However, the β-Gal assay is not highly sensitive because its enzymatic activity is usually determined by a colorimetric method. This low sensitivity also prevents reduction of the scale of the assay. Furthermore, the procedure comprises multiple laborious steps because the enzyme is intracellularly produced. Yeast cells in liquid culture must first be harvested and disrupted with one or a combination of cell wall-degrading enzymes, beads, and/or detergent-containing reagents to expose the intracellular protein for enzymatic assay. The solution must then be centrifuged again to remove cell debris. Consequently, the β-Gal reporter assay is rarely applied for high-throughput analysis. Other reporter pro-

* To whom correspondence should be addressed. E-mail: s.ohigya@aist.go.jp.
[†] Bioproduction Research Institute, National Institute of Advanced Industrial Science and Technology (AIST), 2-17-2-1 Tsukisamu-Higashi, Toyohira-ku, Sapporo 062-8517, Japan.
[‡] Health Research Institute.
[§] ATTO Corporation.
[∥] Computational Biology Research Center.
[⊥] Graduate School of Medicine, Hokkaido University.
[▽] Graduate School of Life Sciences, Hokkaido University.
[○] Osaka Prefecture University.
[◆] Bioproduction Research Institute, National Institute of Advanced Industrial Science and Technology (AIST), Tsukuba Central 6, 1-1-1 Higashi, Tsukuba 305-8566, Japan.

(1) Funk, M.; Neidenthal, R.; Mumberg, D.; Brinkmann, K.; Rönicke, V.; Henkel, T. *Methods Enzymol.* **2002**, *350*, 248–257.
(2) Kuroda, S.; Otaka, S.; Fujisawa, Y. *J. Biol. Chem.* **1994**, *269*, 6153–6162.
(3) Hiraishi, H.; Miyake, T.; Ono, B. *Curr. Genet.* **2008**, *53*, 225–234.
(4) Schmitt, M.; Schwanewilm, P.; Ludwig, J.; Lichitenberg-Fraté, H. *Appl. Environ. Microbiol.* **2006**, *72*, 1515–1522.
(5) Routledge, E. J.; Sumpter, J. P. *Environ. Toxicol. Chem.* **1996**, *15*, 241–248.
(6) Bovee, T. F.; Helsdingen, R. J.; Koks, P. D.; Kuiper, H. A.; Hoogenboom, R. L.; Keijer, J. *Gene* **2004**, *21*, 187–200.
(7) Fields, S.; Song, O. *Nature* **1989**, *340*, 245–246.
(8) Rupp, S. *Methods Enzymol.* **2002**, *350*, 112–131.

10.1021/ac100832b
Published on Web 05/28/2010