April 2013

Interpretation, Stratification and Validation of Sequence Variants Affecting mRNA Splicing in Complete Human Genome Sequences

Ben C. Shirley
The University of Western Ontario

Supervisor
Dr. Peter K Rogan
The University of Western Ontario

Graduate Program in Computer Science

A thesis submitted in partial fulfillment of the requirements for the degree in Master of Science

Interpretation, Stratification and Validation of Sequence Variants Affecting mRNA Splicing in Complete Human Genome Sequences

(Thesis format: Monograph)

by

Ben Chambers Shirley

Graduate Program in Computer Science

A thesis submitted in partial fulfillment
of the requirements for the degree of
Master of Science

The School of Graduate and Postdoctoral Studies
The University of Western Ontario
London, Ontario, Canada

Abstract

The Shannon Human Splicing Pipeline software has been developed to analyze variants on a genome-scale. Evidence is provided that this software predicts variants affecting mRNA splicing. Variants are examined through information-based analysis and the context of novel mutations as well as common and rare SNPs with splicing effects are displayed. Potential natural and cryptic mRNA splicing variants are identified, and inactivating mutations are distinguished from leaky mutations. Mutations and rare SNPs were predicted in genomes of three cancer cell lines (U2OS, U251 and A431), supported by expression analyses. After filtering, tractable numbers of potentially deleterious variants are predicted by the software, suitable for further laboratory investigation. In these cell lines, novel functional variants comprised 6–17 inactivating mutations, 1–5 leaky mutations and 6–13 cryptic splicing mutations. Predicted effects were validated by RNA-seq data of the three cell lines, and expression microarray analysis of SNPs in HapMap cell lines.

Keywords

Mutation, mRNA splicing, information theory, next-generation sequencing, genome interpretation, bioinformatics, biological pathway analysis, cancer.

Co-Authorship Statement

Dr. Tyson Whitehead (SHARCNET) developed C libraries used by the Shannon pipeline. Eliseos J. Mucaki developed early versions of Perl scripts used to annotate variants. Dr. Pelin Akan and Paul I. Costea (Royal Institute of Technology – Science for Life Laboratory, Solna, Sweden) generated RNA-seq data on the USOS, A431, and U251 cell lines. Paul I. Costea examined the RNA-seq data and compared it with Shannon pipeline predictions. I repeated all comparisons of predicted pipeline results with RNA-seq with guidance from Dr. Peter Rogan. The work presented in this thesis has been published REF: 20.

Table of Contents

List of Tables

List of Figures

List of Appendices

List of Abbreviations, Symbols, Nomenclature

- A431 - epidermoid squamous carcinoma-derived cell line.
- API – A library of functions, data structures, classes, etc. which can be exploited by a programmer.
- ASSA - Automated splice site analysis server. A tool to predict the effects of sequence changes that alter mRNA splicing in human diseases.
- dbSNP – Single Nucleotide Polymorphism database. A public-domain archive of single nucleotide polymorphisms.
- GCC – GNU Compiler Collection. The standard compiler for most Unix-line operating systems.
- FASTA format – A text-based file containing nucleotide sequences (can also contain peptide sequences) for one or more region in a genome.
- hg18 – Also called NCBI36. The March 2006 human reference sequence.
- hg19 – Also called GRCh37. The February 2009 human reference sequence. The most recent patch is GRCh37.p11. Patch data generally contains alternate haplotype regions.
- HGNC – HUGO Gene Nomenclature Committee. HUGO- or HGNC-approved.
- HUGO – Human Genome Organisation.
- Indel - A genomic insertion or deletion. Indels range in size from a single nucleotide to multiple kilobases.
- Java Swing – The primary Java GUI toolkit. It is an API which provides graphical user interface design functionality.
- NGS – Next generation sequencing.
- NMD – Nonsense-mediated mRNA decay. A surveillance pathway which reduces errors in gene expression by eliminating mRNA transcripts that contain a premature stop codon.
- PCR - Polymerase chain reaction. Used to amplify a small number of (or one) DNA segments, resulting in many copies of the sequence.
- qPCR – Quantitative real-time polymerase chain reaction. For one or more specific sequences in a sample, sequences can be detected and quantified.
- R_i - Information content in bits.
- RNA-seq – Uses high throughput sequencing to sequence cDNA in order to measure the levels of RNA transcripts and their isoforms in a sample.
- SNP – Single nucleotide polymorphism. A single-nucleotide substitution in the genome. A SNV that has been characterized.
- SNV – Single nucleotide variant. A single-nucleotide substitution in the genome.
- U251 - glioblastoma-derived cell line
- U2OS - osteosarcoma-derived cell line
- VUS- Variant of Unknown Significance. A variant which has been documented but has no known pathogenic significance.

1 Introduction

The volume of human next-generation sequencing (NGS) data requiring bioinformatic analysis has necessitated development of high-performance software for genome scale assembly and analysis [1]. Genomic variations found in these analyses, particularly single nucleotide polymorphisms (SNPs), have traditionally been interpreted in terms of amino acid modifications in coding regions. Clinically-significant non-coding variants are a relatively unexplored source of pathogenic mutations and lack a general, high-throughput method to interpret their effects. In this thesis I present genome-scale software which I adapted and further developed to quantify the effect of mutations in the common classes of splice donor (U1) or acceptor (U2)-type sites in a high-throughput manner. Mutations predicted with this method will be useful for pinpointing potentially deleterious variants suitable for further laboratory investigation.

Clinical studies have deemed the vast majority of known variants in patients with Mendelian (single-gene) disorders to be of uncertain pathogenic significance (VUS) [2,3]. *Cis* mutations can affect protein translation, mRNA processing and initiation of transcription. *In silico* methods have been developed for the first two of these cases (e.g., [4,5]), but have only been routinely applied for protein coding changes in genome-scale applications (e.g., [6]). Many NGS studies classify splicing mutations only as those located within the highly conserved dinucleotides within each splice junction (*e.g.*,[7]). Although more sensitive methods have been developed which assess other conserved sequence elements [8-12], none have been scaled for the large numbers of variants generated by NGS and nor have they been validated for this data. Exonic variants in close proximity to splice junctions but outside of this window may be classified as synonymous, missense or nonsense substitutions, yet still have profound effects on splicing, which may be the predominant contributor to the phenotype. Unless multiple affected patients are reported with the same mutation, the mutations are transmitted through pedigrees, and functional assays verify their effects, these variants in patients are generally classified as VUS. mRNA splicing mutations are common in Mendelian diseases [13,14], and it is likely that they contribute to many complex disorders. Clearly, genome-scale predictive methods

that filter out benign or small changes in mRNA splicing due to sequence variation will be essential for mutation discovery in exomes, complete genomes and high-density targeted deep sequencing projects. Examination of individual variants in the laboratory with functional assays is both expensive and inefficient as many variants are not likely to be deleterious, or differ significantly in their pathogenicity.

The Automated Splice Site Analysis (ASSA) [5] server evaluates single mutations that change splice site strength with information-based models [15]. The average information, $R_{sequence}$, of a set of binding sites recognized by the same protein (such as U1 or U2) describes the conservation of these sequences. Sequences are ranked according to their individual information content (R_i in bits) [15-17]. Individual information content is a portable, universal measure which allows direct comparison of binding sites across the genome or transcriptome, regardless of the sequence or protein recognizer. Functional binding sites have $R_i > 0$, corresponding to $\Delta G < 0$ kcal/mol [18]. Strong binding sites have $R_i >> R_{sequence}$, while weak sites have $R_i << R_{sequence}$. Any sequence variation may change its protein binding affinity, which is reflected by a change in the computed R_i of that binding site. A 1-bit change in information content (ΔR_i) corresponds to a ≥ 2 fold change in binding affinity ($100/2^{\Delta Ri}$). The ASSA server has been widely used and its sensitivity and specificity has previously been extensively validated in hundreds of studies of individual mutations (http://tinyurl.com/splice-server-citations). However, it requires approximately 30 seconds to examine a single variant and is therefore not suitable for comprehensive analysis of whole-genome sequencing data. The Shannon pipeline was implemented using the same mathematical approach and information weight matrices as ASSA to carry out batch information-based analysis of thousands of mutations from the *BRCA1* and *BRCA2* genes in the Breast Cancer Information Core Database [19]. In the present study, the software has been adapted to perform a single matrix algebraic calculation across a genome with an efficient state machine that significantly increases computational speed over ASSA.

During the tenure of my M.Sc. I was involved in the release of three papers [20-22]. In this thesis, I will present work described in [20]. My contribution in [21] was the implementation an algorithm which determines single-copy regions (regions not repeated elsewhere in the

genome) without the use of a catalogue of repetitive sequences. This work allowed me to appreciate the scope and sheer size of genome-scale programming projects. In particular, although my program was executed on the Shared Hierarchical Academic Research Computing Network (SHARCNET) using 128 cores, the full execution time of the software was ~3 months. This inspired me to become involved with a software project operating on a genome-scale, which executed in a far shorter time. Additionally, my role in [22] was to assist in modifications to ASSA (and the newer ASSEDA). Implementing these modifications improved my understanding of molecular information theory and its applications related to splicing prediction.

Several years ago, C libraries were developed by Tyson Whitehead which calculate the information content of a genomic region based on information weight matrices [19,20]. These libraries were designed to execute very quickly while not sacrificing specificity. Chromosomes are stored in memory one at a time and each base is stored using 17 bytes. The longest human chromosomes can therefore be represented using a few gigabytes of memory. Reference sequences for each chromosome are read from FASTA files and parsed at disk speed (parsed as fast as the disk can read the file). As a consequence, the libraries require only several seconds to parse each chromosome. Sequence blocks, four kilobytes in size, are extracted from the appropriate region of a parsed FASTA file and information analysis is performed. As the appropriate regions of the genome are extracted from memory on demand there is no need for indexing, thus execution speed is highly optimized.

In this thesis I describe the development of software which extends these C libraries. Variants are annotated, stratified, ordered in terms of relevance, presented in a user-friendly manner, and the code is integrated with the CLC-Bio Workbench. Predicted deleterious mutations are compared with RNA-seq data from genomes of three cancer cell lines to assess their validity.

2 Literature Review

2.1 Overview of splicing mechanisms

The human genome contains approximately 3,000,000,000 nucleotides [23]. It is comprised of DNA which is made up of four nucleotides - adenine, thymine, cytosine, and guanine - denoted A, T, C, or G joined by phosphodiester bonds. These nucleotides form a code which eventually translates into proteins necessary for survival. Nucleotides (bases) can form base pairs with other nucleotides. For each base, there is another nucleotide which binds to it called its complementary base. The base A binds readily to the base T, and G binds with C. Base pairs are formed by hydrogen bonds (AT and GC base pairs experience 2 or 3 hydrogen bonds respectively). In the genome, DNA forms its double helix structure by pairing a strand to another complementary strand. Only ~1.1% of DNA directly codes for proteins and these regions are referred to as exons. Collections of exons and non-coding regions called introns between them (along with other regulatory elements such as promoters, enhancers, etc.) form genes. To ensure proper gene function, introns must be precisely removed to result in an mRNA suitable for translation to protein [24].

2.1.1 The spliceosome and the splicing process

Introns are bound by conserved sequences that define their 5′ and 3′ ends. The spliceosome is comprised of small nuclear RNA (snRNA) and protein and is a macromolecule which excises introns during splicing. During transcription to RNA, the spliceosome acts in two major steps. First, the 5′ splice site base pairs with the U1 snRNA which is part of the spliceosome and splicing factor 1 (SF1) binds to the branch point, located upstream of the 3′ splice site [25]. The 5′ splice site and branch point are drawn together, and the 5′ splice site undergoes a nucleophilic attack which breaks the phosphodiester bond at the splice junction and simultaneously forms a linkage between the branch point and the 5′ end of the intron. This results in an intron conformation structure similar to a lariat. In the second step, the newly released 3′ hydroxyl of the 5′ exon attacks the 3′ splice site. Again, the phosphodiester bond at the splice junction is broken, and in its place a bond is formed between the two exons [24].

2.1.2 Donor and acceptor splice sites

Located at each end of an intron is a splice site denoted as a donor site at the 5′ end of the intron or acceptor site at the 3′ end. Both of these sites are referred to as natural sites (the splice site used in the absence of mutation). The efficiency of splicing is partially determined by the highly conserved GT and AG dinucleotides present at the donor and acceptor sites, respectively. These dinucleotides are certainly not the sole determinants of normal splicing however. The length of donor and acceptor sites have been defined as 10bp (-3, +6) and 28bp (-25, +2) respectively where 0 refers to the first nucleotide of the splice junction [26]. In 1986, a study examined approximately 400 vertebrate genes and derived consensus sequences for both donor and acceptor sites [27]. The strong conservation of these regions was evident – even across species barriers – which implied an important role in splicing.

Point mutations within splice sites affecting pre-mRNA splicing account for approximately 15% of human genetic disease [28]. Mutation within any region of a donor or acceptor site can contribute to a reduction (or in rare cases, a strengthening) of the site's binding affinity to the spliceosome. Weakened natural sites may cause aberrant splicing.

2.1.3 Splice site recognition and variation

Proper definition of donor and acceptor sites is central to proper RNA and protein formation. Splicing machinery is tasked with locating exons (137 nucleotides long on average) separated by much longer intronic regions [29]. This task is made more difficult through the existence of "decoy" sites (cryptic splice sites). These sites contain similar sequences to splice sites and must be accounted for. If splicing machinery relied only on nucleotide sequence, cryptic sites would be frequently used in place of natural sites (sites which are generally used barring genetic variation). In 1994, a study catalogued instances mammalian splice site mutation [30]. In this study, four phenotypes were observed as a result of splice site mutation. Exon skipping, cryptic site use, creation of a pseudo-exon entirely within an intron, and intron retention were observed. In particular, 55% of observed phenotypes demonstrated exon skipping. The prevalence of exon skipping

(when only one splice site of an intron is affected) implied that splice sites are recognized as pairs [29]. In addition, the size of exons is also a factor. Only 3.5% of exons are of length >300 nucleotides and less than 1% >400. This again implies that the sequence of splice sites is not the only determining factor in splice site recognition.

Up to ~50% of deleterious alterations in genes may be caused by splicing mutations [31]. Although mutations located anywhere in a gene can impair the splicing process, most deleterious variants have been found in the GT/AG dinucleotides located at splice junctions [31]. The GT/AG nucleotides are highly conservered. However, any nucleotide substitutions within splice sites may alter splicing outcome. There are three main potentially deleterious outcomes which can occur as a result of variation within splice sites (**Figure 1**). A mutation may weaken a splice site to such a degree that it no longer functions effectively. This may cause the affected exon to be missing from the resulting mRNA (exon skipping) or the exon to extend past the natural site into the intron (intron inclusion). Other mutations may weaken a splice site to a degree insufficient to cause exon skipping. Mutations of this type may result in a lesser expression of a normal splice isoform (leaky). Finally, mutations that strengthen nearby cryptic sites or weaken a natural site with a cryptic site nearby may lead to cryptic site binding. Cryptic site binding may shorten or extend an exon. Multiple mRNA splice isoforms can be produced. Variation in the nucleotide sequence of donor or acceptor sites can increase or decrease the abundance of a specific isoform in the population of mRNA.

2.2 Molecular information theory

Information theory was first devised by Claude E. Shannon and published in his article "A Mathematical Theory of Communication, Part I." in 1948. It is a branch of mathematics used to quantify information and determine the information content in a system. Two main aspects of communication are addressed; 1) methods to measure information, and 2) determination of the maximum information which can be sent and received through a communications system.

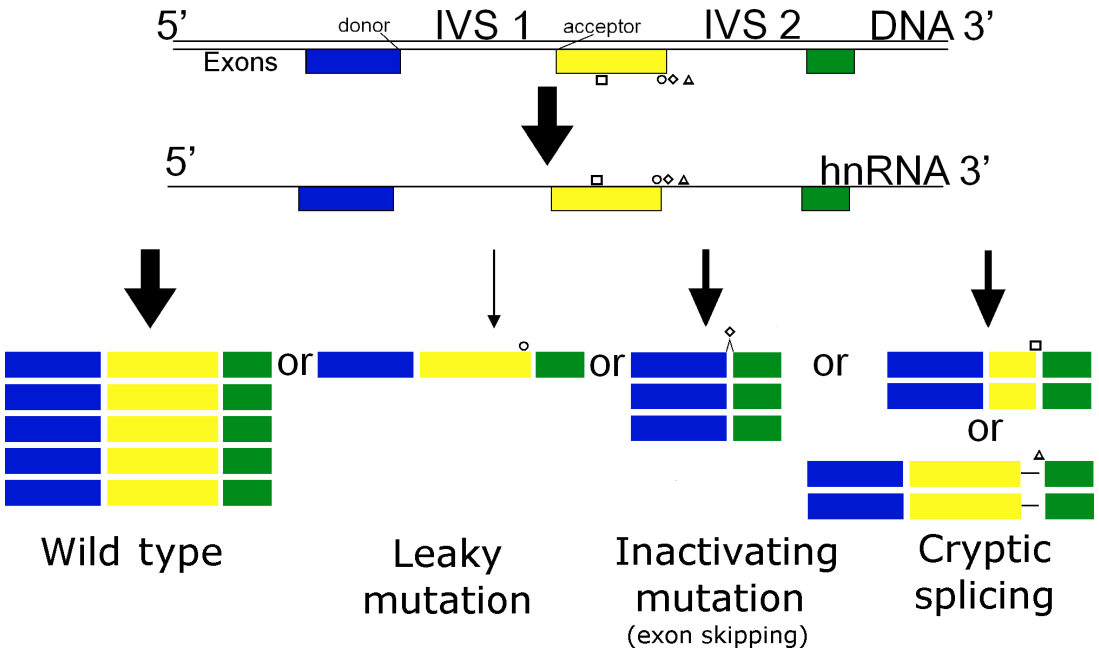


Figure 1. Types of splicing mutations that affect structure and/or abundance of resulting transcripts

The diagram illustrates potential outcomes of mRNA splicing mutations predicted by the Shannon pipeline. Intervening sequences (IVS) contain an intron and other nucleotides not present in the resulting mRNA. Variation within splice donor and/or acceptor sites may lead to altered splicing events such as exon skipping (◊), exonic (□) or intronic (△) cryptic site use, and/or reduction in the abundance of normally spliced mRNA forms, termed leaky mutations (○).

2.2.1 Basics of molecular information theory

Molecular information theory describes biological interactions by exploiting the mathematics of information theory [32] and applying it to biological systems. In particular, information-based methods can be used to calculate the information content of binding sites recognized by one kind of macromolecule (*e.g.*, the spliceosome). Known binding sites must be aligned and examined, however the contributions of individual positions are not ignored as they are in consensus sequences. A consensus sequence aligns sequences and reports the most prevalent base at each position. The information content of sites recognized by a single macromolecule is denoted $R_{sequence}$. Two sources of information are needed to compute $R_{sequence}$: 1) The nucleotide sequences where a macromolecule has demonstrated the ability to bind. 2) The sequences must be viewed in the context of an entire communication system. Thus, the nucleotide composition of the genome in which the macromolecule functions must be determined [33].

Sequences recognized by a single macromolecule within a genome known to experience macromolecule binding are aligned in a manner to allow the greatest homology between bases. The $R_{sequence}$ of the site can then be calculated. The general formula for uncertainty can be modified as follows to represent nucleotide sequences

$$H_s(L) = -\sum_{B=A}^{T} f(B,L)\log_2 f(B,L) \ \text{(bits per base)} \tag{2.1}$$

where B = {A,C,G,T}, and $f(B,L)$ is the frequency of base B in position L is found in the sequence. This equation can be applied to a full genome by exploiting existing data on the nucleotide content of the human genome. If a set of random nucleotides sequences were extracted from the human genome and aligned, all four bases would be observed, with probabilities $P(B)$. Thus, the equation can be modified as follows

$$H_g = -\sum_{B=A}^{T} P(B)\log_2 P(B) \ \text{(bits per base)} \tag{2.2}$$

When this formula is applied genome-wide, the resulting uncertainty is higher than when applied to only known binding sites. This implies there is a pattern in the nucleotide sequences located at splice sites. This was certainly an expected result, as splice site sequences were known to be conserved. For each position L, that decrease in uncertainty can be demonstrated by

$$R_{sequence} = \sum_{L} R_{sequence}(L) \text{ (bits per site)} \tag{2.3}$$

$R_{sequence}(L)$ is therefore a measure of the information gained (uncertainty lost) by aligning the binding sites. Information is additive [16], thus the total information gained is equal to the decrease in uncertainty across all sites

$$R_{sequence} = \sum_{L} \{E(H_{nb}) - H_s(L)\} \text{ (bits per site)} \tag{2.4}$$

where H_{nb} is the probability of obtaining a particular combination of n bases. Although information itself is additive, this equation has been simplified by assuming that the frequencies of bases observed at one position are statistically independent of any other.

2.2.2 Information weight matrices

Information content can be defined as the number of choices needed to describe a sequence pattern, using a logarithmic scale in bits [33]. These data can be represented as a weight matrix calculated by

$$R_{iw}(b,l) = 2 - (-\log_2 f(b,l) + e(n(l))) \text{ (bits per base)} \tag{2.5}$$

$R_{iw}(b,l)$ (also referred to as RIBL) is a two dimensional array containing b rows and l columns where b = {A,T,C,G}, l is the position in the splice site, and $e(n(l))$ is a sample size correction factor for the n sequences at position l used to create $f(b,l)$. The 2 represents the bits of uncertainty a recognizer has before binding to a site containing 4 possible bases ($\log_2(4)$). As a whole, this matrix represents the sequence conservation of each nucleotide, measured in bits and can be exploited to compare sites to one another, search for new sites, to compare sites to other quantitative data such as DNA-protein

binding strength, and other applications [34]. The most frequent base at each position of the weight matrix is assigned the largest individual information (R_i in bits) value. Therefore, a consensus sequence can be generated by selecting the highest R_i value at each position. The individual information of a sequence can be compared to an information weight matrix in the following way

$$R_i(j) = \sum_{l} \sum_{b=a}^{t} s(b,l,j) R_{iw}(b,l) \text{ (bits per site)} \tag{2.6}$$

where j is an individual sequence and $s(b,l,j)$ is a simple two dimensional array which represents the jth sequence. As j is a single sequence, frequencies are not involved in this matrix as such. Instead, elements in $s(b,l,j)$ contain the value 0 at every position with the exception of base b at position l which contains 1.

2.2.3 Information theory and human splicing site mutations

The human genome can be viewed as a system which contains information. As is widely known, DNA triplets code for the synthesis of specific amino acids. However, this is not the only kind of information stored within the genome. As discussed in chapter 2.1, splice sites are comprised of similar, conserved DNA sequences. Information can be described as a decrease in uncertainty, therefore the similarity of splice sites implies that they contain information.

The effects of genetic variation (base substitutions) within a sequence can be calculated by examining the R_i or the common and variant alleles. The difference between their respective information contents is denoted as ΔR_i. As R_i is on a logarithmic scale, the minimum change in binding affinity of two sites is $2^{\Delta Ri}$ [15]. $R_{iw}(b,l)$ matrices have been computed for 56,985 acceptor and 56,286 donor sites [35]. Matrices used by the Shannon pipeline are based on these models and were obtained using the same method, but are based on sites on both strands. Models in [35] were based on only on the positive (+) strand. Matrices used by the pipeline are based on 108,079 acceptor sites and 111,772 donor sites. The mean distribution of R_i values across these sites is denoted $R_{sequence}$ where the $R_{sequence}$ of donor and acceptor sites are computed separately. Therefore, $R_{sequence}$

represents the average information required for splicing to occur at a splice site. It also reflects the strength of the splice site. Those splice sites which have R_i values << $R_{sequence}$ are weak sites, while those with R_i >> $R_{sequence}$ are strong sites. Non functional sites have R_i values less than ~1.6 bits.

3 Shannon pipeline - Methods

3.1 Shannon pipeline software architecture

I have implemented the Shannon pipeline plugin using the CLC-Bio genomics developer toolkit to simplify access to this technology and interpretation by novice users. The same plugin can be executed on a single client computer, a remote server or a grid system, and benefits from automated software updates. The server version uses an architecture in which a Workbench client transmits variant data to the server, which performs the computations, and returns results that can be filtered and formatted on the client. A standalone version of the fully functional Genome Workbench plugin is also available. By contrast, the splicing mutation feature that is native in CLC-Bio Genomics' products is limited to detecting changes in dinucleotides at the exon boundaries, which represent fewer than 5% of all splicing mutations detected by the Shannon pipeline.

The Shannon pipeline uses an efficient algorithm coded in C to quickly analyze genome-scale data sources for information changes (**Figure 2**). Methods for computing R_i and ΔR_i values determine the dot product of an information weight matrix and the unitary sequence vector for each genomic window and comparing the resultant scalar values of the reference and variant sequences [36]. C libraries determine the information content of a position in the reference genome and after a variant is introduced. This method uses convolution-style sliding-window computation of all sequence changes for each complete chromosome sequence resident in RAM. To expedite processing, the software currently only handles single nucleotide variants (SNV) – which are the most prevalent type of variation. Changes in R_i introduced by genomic variation are computed by subtracting the initial R_i value of a position by the sum over a surrounding window, then adding the new value for each position (ΔR_i). Perl scripts wrap these C libraries and annotate output. Integration with the CLC-Bio Workbench environment was achieved through code written in Java utilizing the CLC-Bio developer API. This software is assembled as a client plugin requiring a connection to the server to execute, a server plugin and a standalone client plugin. Two additional dependency plugins contain a modified dbSNP135 (containing only variant, rsID and overall frequency), Ensembl Exon Data

(Build 66) and GRCh37 (hg19)/NCBI36 (hg18), respectively, allowing the software to execute with no active internet connection and incorporates all necessary annotations required to contextualize a potential mutation.

Input flat files containing sequence variants that differ from the reference genome are imported into the CLC-Bio Java environment. The file must be either Variant Call Format (VCF) [37] or a tab-delimited format with the following fields: [chromosome #] [unique identifier] [coordinate] [reference/variant]. Coordinates can be hg18 or hg19. All variants appearing in this study are hg19. Genomic insertions and deletions (indels) present in input files are not considered for analysis.

3.2 Perl scripts and modules

Two Perl scripts were previously written to perform variant annotation. I significantly modified these scripts by increasing memory/time efficiency and modularizing the code to simplify testing. To this end, I divided the two preexisting scripts into 5 Perl modules. I wrote an additional two Perl scripts and several modules to automate plugin installation and filtering of variants. I will briefly describe the functionality of the scripts and modules here. A straightforward Perl script that splits variants on each chromosome into separate files will not be described.

3.2.1.1 MainControl.pl

This script is executed from within Java code and serves as part of the connection between command-line code and the CLC-Bio Workbench. More details related to this interconnectivity can be found in section 3.3.1.6. Standard output and standard error are redirected stdout.log and stderr.log respectively. This was done to allow detailed error messages to be effectively communicated to the CLC-Bio Workbench. The main purpose of the script is to call a series of Perl modules which each append some annotation to an array of data. The array is passed by reference from each module back to MainControl.pl and passed to the next module. The script also updates a variable containing a rudimentary progress percentage which is sent to the Workbench every 5 seconds. It is used to update a progress bar displayed to the user. Between the execution of each module the progress percentage is updated to an increased value.

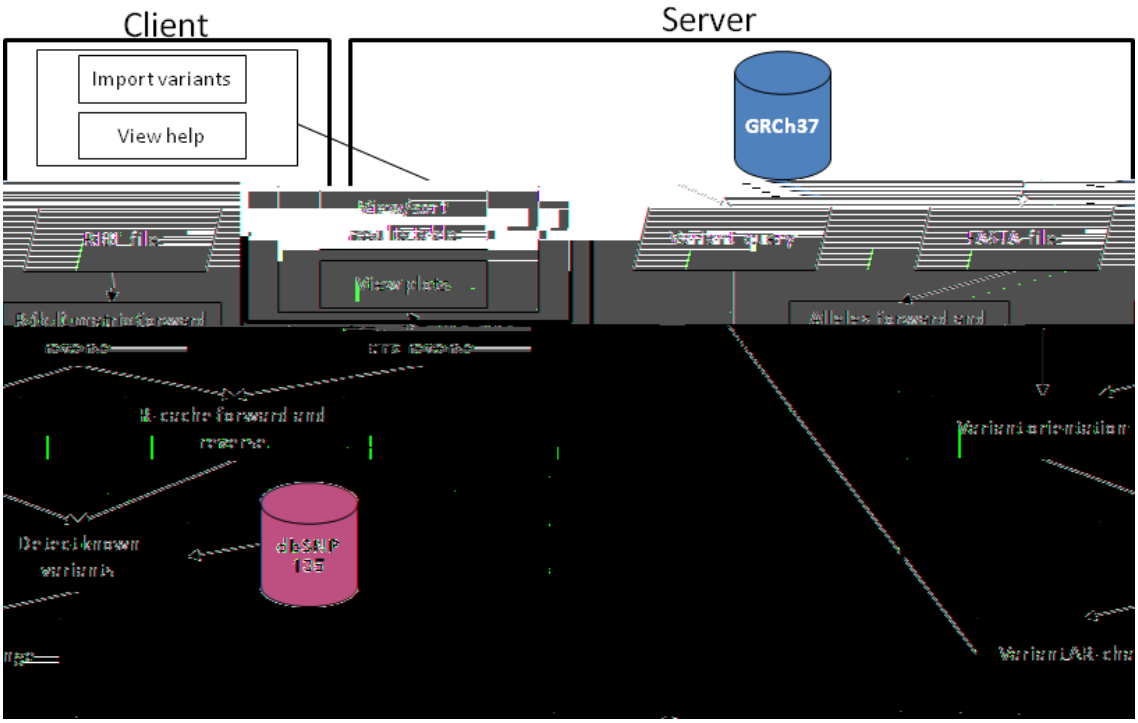


Figure 2. Flow chart of the Shannon Human Splicing Pipeline.

Client and server tasks are depicted separately. Interactions between them are denoted by arrows crossing the client/server barrier. The pipeline can also be executed on the client in a standalone manner. In that case, all server actions are performed on the client machine.

3.2.1.2 InstallShannonPipeline.pm

C libraries must be compiled before they can be executed. The C libraries are installed as Perl modules through the use of preexisting Perl wrappers. This Perl module automatically compiles the libraries if necessary. To check if the libraries are already installed I use "eval 'require Rogan::FASTA" (Rogan::FASTA is the name of one of the modules). If the module is already installed, installation will not take place. Otherwise, a series of commands will be run, some of which are used for potential error reporting. The current directory is changed to the location where the libraries will be installed and the date and current directory are sent to stdout.log. All external commands are run by calling Perl's system function using an array constructed specifically for each command. The commands are:

1) my $makeClean = qq(make clean 2>> ./stderr.log 1>> ./stdout.log);

2) my $createMakefile = qq(perl Makefile.PL LIB="./installeddir/" PREFIX="./extras/" 2>> ./stderr.log 1>> ./stdout.log);

3) my $makeCommand = qq(make 2>> ./stderr.log 1>> ./stdout.log);

4) my $makeInstall = qq(make install 2>> ./stderr.log 1>> ./stdout.log);

Command number 2 specifies LIB and PREFIX options to allow local installation without the need for root access. Each command is executed using system. System error codes are trapped and examined. If an error occurs in any of these steps, execution stops and error code 100 is sent to the Workbench, indicating a problem with installation.

3.2.1.3 Pipeline-Initial-Scan.pl

A parameters file created by Java code is examined to determine the location of appropriate FASTA files containing the reference genome, as well as the location of donor and acceptor information weight matrices. Chromosomes are examined one at a time. Before variants on a chromosome can be examined, the appropriate FASTA file is parsed into an efficient state machine using a C library. Donor and acceptor information weight matrices are also parsed. Each variant is examined using C libraries to determine its R_i before and after the contribution of a specific variant. The following information is written to a file for each variant: 1) chromosome, 2) variant unique ID, 3) splice site

coordinate, 4) R_i before variant contribution, 5) R_i after variant contribution, 6) donor or acceptor site, 7) strand, 8) variant coordinate, 9) variant (e.g., G/T). The process is repeated for all chromosomes.

3.2.1.4 WriteTracksAndFindIfWithinGene.pm

In addition to plot and tabular output, BED tracks are also generated by the pipeline. The tracks contain ΔR_i for each variant and can be viewed in a genome browser. The module reads the file generated by Pipeline-Initial-Scan.pl line by line. Hash tables are created which allow constant time searches named donorpos (positive strand, donor), donorneg, accpos, and accneg. Each variant is sent to a generalized function which accepts the appropriate hash, file handle to write to (appropriate track), chromosome number, and variant ΔR_i. This function simultaneously appends to the appropriate track file as well as adding each variant to the appropriate hash.

Ensembl Gene 66 is examined along with the list of variants. If a variant is found within transcript start and end coordinates, then it is within a gene. Variants meeting that requirement are added to an array which is returned by reference to MainControl.pl. Variants not found to be within an exon are not annotated further.

3.2.1.5 AnnotateNaturalSites.pm

This module determines if there is a nearby natural site close to the variant. The array of variants found to be within an exon by WriteTracksAndFindIfWithinGene.pm are examined along with natural site coordinates found in Ensembl Gene 66. Hash tables are built containing the locations of donor and acceptor natural sites on both strands. If a variant is found to affect a known natural site the variant is annotated as a natural site. Otherwise it is annotated as a cryptic site. The array is returned to the main Perl script for further annotation.

3.2.1.6 AnnotateExons.pm

Each variant within a cryptic site is examined to determine if there is a natural site nearby. The presence of a natural site allows the cryptic and natural site to be directly compared to one another. It can be determined if the cryptic site flanks the 3′ or 5′ end of

the exon and R_i values of the cryptic and natural site can be compared after the contribution of the cryptic variant is observed. The range to check for a nearby natural site is determined by a field in the parameters file created using Java code. Currently, this value is 300. Thus, a range of up to 300 bp around the cryptic site is examined and compared with natural sites from Ensembl Gene 66. Variants near a natural site are annotated 3′ or 5′ flanking and returned.

3.2.1.7 GetStrengthsOfNearestNaturalSites.pm

Nearby natural sites potentially found in AnnotateExons.pm are compared with the appropriate variant cryptic site to determine which site has the higher R_i value. If the nearby natural site has the higher R_i, the variant is annotated as greater. Otherwise the variant is annotated as less. If a variant does not have a nearby natural site within 300bp, a '-' is annotated to the variant as a placeholder. The resulting variants are returned for further annotation.

3.2.1.8 AnnotateKnownVariants.pm

It must be determined if a variant has been previously documented in dbSNP or if it is a novel variant. The array returned from the previous module is used to generate a hash containing the chromosome, coordinate, variant, and unique ID of each variant. Each entry in dbSNP is examined and compared to the hash. If there is a match at the same chromosome, coordinate, and variant then the variant is annotated with the appropriate rsID. This task is made more difficult as each coordinate in dbSNP can have multiple variants. Additionally, dbSNP reports variants as if they were located on the positive strand. Thus, if the strand is '-', the reverse complement of the variant must be compared to the dbSNP entry. Again, if no rsID is found, a '-' is annotated as a placeholder. There is no more annotation to perform at this point. The array is returned to the main script and written to a file which will be imported by Java code to be viewed in the CLC-Bio Workbench.

3.2.1.9 FilterOutputData.pl

The user may request that output contain variants on only the positive strand, negative strand, or both. Additionally, the user may request that only donor sites, acceptor sites, cryptic sites, or natural sites be displayed. This script accesses those preferences as command-line arguments and eliminates variants matching the request criteria from the file to be imported to the Workbench.

3.3 CLC-Bio integration

The CLC-Bio Genomics Workbench is a commercial workspace for genomics research (www.clcbio.com). Files are not generally used in this workspace, instead files are imported as ClcObjects (objects). These objects represent specific biological data and are associated with appropriate editors, viewers, and other object types. The Workbench is also a host to third-party applications (plugins) generally implemented using pure Java. As the Shannon pipeline was coded in C and Perl, a Java-based connection to the CLC environment was required. Additionally, as run-time is a paramount concern for the pipeline given the number of variants it examines simultaneously, the existing C libraries could not be converted to Java while maintaining necessary execution speed. Thus, I designed the Java code not only to communicate with the CLC-Bio Workbench, but also with the Perl and C code on the command-line.

3.3.1 Java classes

The CLC-Bio application programming interface (API) contains classes which smooth integration of Java code with their system and provide additional bioinformatics-related functionality. I developed a total of 16 classes in Java (**Table 1**) to provide functionality including object importers, a wizard to allow users to input pipeline preferences, help screens, a bridge between Java and command-line code, and additional miscellaneous classes. Some classes are very short and required very little modification from example code provided on the CLC-Bio developer's website (connection.clcdeveloper.com). These classes will not be described as they are straightforward to code and are necessary for all plugins.

3.3.1.1 Importing variant data

VCF is the standard format used by researchers and industry to store variants. It was necessary that I develop a Java class to import VCF files into a format compatible with the CLC-Bio Workbench. To reduce the degree to which error handling within the command-line portion of the plugin is required, checks are performed during the import process to ensure proper variant formatting. Although VCF files can contain a great deal of data for each variant, the Shannon pipeline requires only 5 fields of ≥ 9 potential fields in VCF. These fields are: 1) the chromosome within which the variant is located (CHROM), 2) genomic coordinates of the variant (POS), 3) a unique variant identifier (ID), 4) the reference nucleotide at the genomic coordinates specified (REF), and 5) the nucleotide observed at that position as a result of variation (ALT). VCF files are read entirely into memory and examined line by line (one variant on each line) to ensure proper formatting. First, the CHROM column is examined to ensure the chromosome specified is a valid chromosome. Valid chromosomes include {1…22,X,Y, human alternate locus/patch information for GRCh37.p11}. Preceding letters such as "Chr", "chr", or "ch" are removed and if the remaining chromosome field matches a valid chromosome, CHROM is valid. Genomic coordinate must be an integer within valid chromosome lengths for the chromosome specified. In several instances, hg18 chromosomes are longer than their hg19 counterparts. The Shannon pipeline accepts data using either hg18 or hg19 coordinates, however at the time of import the genome build is unknown. Therefore, a valid genomics coordinate is defined as 0 ≤ valid coordinate ≤ max(hg18 chromosome length, hg19 chromosome length). The ID column does not have to be examined as any input is valid. However, to ensure it is a unique identifier "-#" is appended to each variant ID, where # is the line number of the VCF file. The REF field must contain a single valid nucleotide. Multiple REF nucleotides imply an indel, the contribution of which the Shannon pipeline cannot currently predict. Multiple ALT nucleotides split the variant into multiple separate variants with single REF and ALT nucleotides. Valid variants are stored in a tabular format object in the CLC-Bio Workbench environment named "[name of VCF file][timestamp]". Those variants which fail any of these formatting requirements are stored in an object "[name of VCF file]_InvalidVariants" with appropriate error messages appended to each variant.

Table 1. Shannon pipeline Java class list and brief descriptions

Class name	Package	Significant modification required	Brief description
LaunchPipelineAlgo	Base	✔	Executes Shannon pipeline command-line code and imports results.
LaunchPipelineParameters	Base	✔	Allows wizard parameters to be accessed by LaunchPipelineAlgo
VCFImport	Base	✔	Imports VCF files as GeneralClcTabular
PlotImportDeltaRi	Base	✔	Imports Shannon pipeline results and creates ΔR_i plot objects.
PlotImportFinalRi	Base	✔	Imports Shannon pipeline results and creates R_i plot objects
PipelineOutputClcTabularImport	Base	✔	Imports Shannon pipeline command-line results as GeneralClcTabular
VariantClcTabularImport	Base	✔	Imports variants in 'Shannon Basic Format' as GeneralClcTabular
CommandLineExecutor	Base		Creates and executes a command-line process
InitClient	Client	✔	Creates a folder in the Workbench containing sample imported variants
LaunchPipelineRemoteAlgoAction	Client	✔	Creates instance of LaunchPipelineAlgo and launches preferences wizard
LaunchPipelineView	Client	✔	Creates layout of wizard screen three and sends user preferences to LaunchPipelineParameters
PipelineActionGroup	Client		Specify Shannon pipeline icon and add Shannon pipeline launcher to the Workbench toolbox
LaunchPipelineCommand	Server		Creates instance of LaunchPipelineAlgo to run on server machine

A second import class I developed which imports variants in the format [chromosome #] [unique identifier] [coordinate] [reference/variant] will not be described in detail. All formatting checks use the same methods as described in the VCF import class. This class was devised primarily for testing in the early stages of pipeline development.

3.3.1.2 Manhattan-style plot importer

Visual representations of data allow overall trends to be observed and individual outliers to be easily identified. CLC-Bio's API provides a class for this purpose named MAScatterPlot. My task was to prepare Shannon pipeline data for visualization as well as create plots using MAScatterPlot. It is required that data points are sorted before plot creation. I implemented a Quicksort algorithm to accomplish the sort. After sorting, data are separated into an array representing the X axis of the plot and an array representing the Y axis. MAScatterPlot allows tooltips to be displayed upon hovering the mouse pointer over a data point. Tooltips contents include chromosome, coordinate, ΔR_i, R_i after variant contribution, and rsID (if available). Separate plots are created to visualize both ΔR_i and final R_i for each chromosome {1..22,X,Y} as well as a genome-wide plot. Thus, if variants are present on all chromosomes a total of 50 plots are generated.

3.3.1.3 Tabular results importer

All Shannon pipeline results are imported and displayed in tabular format. Results generated during command-line execution are recorded in a tab delimited file. The tabular import class functions similarly to the variant importer. The Shannon pipeline results file is read into memory and examined line by line and data are reordered and formatted. Column headers are named as well as the resulting tables.

Variants are split into four separate tables. 'Complete Variant Information' contains all variants. 'Inactivating Variant Information' and 'Leaky Variant Information' contain variants predicted to be inactivating or leaky respectively. 'Cryptic Variant Information' contains all variants located within cryptic splice sites. Column headers in Inactivating Variant Information and Leaky Variant Information tables are Chromosome, Coordinate (of splice site), Strand, R_i-initial, R_i-final, ΔR_i, Type (donor or acceptor), Gene Name, Location (natural site or cryptic site), Input Coodinate (of variant), Input Variant, and

Input ID (unique variant ID). There are additional column headers in the Complete Variant Information and Cryptic Variant Information tables which are Location Type (intronic or exonic), Loc. Rel. to Exon (cryptic site is 5′-flanking or 3′-flanking compared to the nearest exon), Dist. From nearest nat. site, Loc. of nearest nat. site (coordinate of nearest natural site), R_i of Nearest Nat. Site, Cryptic R_i relative to nat. (greater or less), rsID if Available, and Average Heterozygosity (if rsID is available).

Columns containing real numbers are rounded to two decimal places. Strand is represented as '0' or '1' in Shannon pipeline results and is converted to '+' or '-' respectively. Some columns may have fields which contain no data such as Loc. Rel. to Exon if a nearby natural site is not found. In these cases the entry is filled with a null value to allow automatic sorting. By default, ClcGeneralTabular sorts columns when the header is clicked. To function intuitively, columns must contain only a single type of data or null values. If more than one type of data are present, sorting will default to lexicographical order.

3.3.1.4 User preferences

I created a wizard built upon CLC-Bio's class ClcWizardStepView. The first screen of the wizard is created entirely by CLC-Bio and determines whether the Shannon pipeline should run on a client (local computer), server, or grid system. The second and fourth screens through which the user selects the location of Shannon pipeline input and where to save Shannon pipeline results are also generated by CLC-Bio. I have restricted the object types which may be used as Shannon pipeline input to GeneralClcTabular objects. This object type is generated by the import classes discussed in 3.3.1.1.

I created the third wizard screen (**Figure 3**) using Java Swing. A JComboBox allows the user to indicate which genome build is appropriate. Variants are represented in hg18 or hg19 coordinates. Four groups of three JRadioButtons are used to determine the following: 1) In the types frame, donor, acceptors, or both may be displayed to the user after pipeline execution. 2) cryptic sites, donor sites, or both types of sites may be displayed. 3) In the 'Output Format' frame, the user can opt to create delta R_i plots, final R_i plots, or both. 4) The user can opt to view output containing one or both strands. By

default, the selections made are hg19, show donors and acceptors, show cryptic and natural sites, show both types of plots, and show both strands. Choices made at this step are recorded using the class LaunchPipelineParameters which extends CLC-Bio's class AlgoParametersInterpreter. I tailored functions in the class to accept the specific choices users are offered. This class allows parameters to be accessed by the LaunchPipelineAlgo class (discussed in 3.3.1.6).

3.3.1.5 Ensembl, and dbSNP, and human genome distributions

Four large databases are required for variant annotation and R_i prediction. Ensembl Gene 66 contains information needed for gene annotation. The Single Nucleotide Polymorphism Database 130/135 (dbSNP) is used to determine if a variant is novel. Two full reference genomes (hg18, hg19) in FASTA format are also required to examine variants on each respective genome build.

As CLC-Bio requires that a plugin can run with no internet connection, it was necessary to package the databases for release. In the initial stages of plugin development I attempted to package the necessary databases along with the plugin itself. However, this would have made updating the plugin difficult and would result in very large downloads for users every update. It was decided – with input from CLC-Bio – to instead package the databases in separate plugins. By employing this method, database plugins can be downloaded once by the user and any subsequent updates to the main plugin can be downloaded separately. This solution also lends itself to future updates of the reference genome versions and associated annotations, and the development of applications that enable mutation analysis in non-human genomes.

I created two additional plugins containing these databases. The first plugin contains hg18 FASTA files, dbSNP 130, and Ensembl Gene 66 (a 'lift-over' was performed to transform genomic coordinates to hg18). A second plugin was created which contains hg19 FASTA files, dbSNP 135, and Ensembl Gene 66. During execution of the main plugin, a check is performed that confirms the required database plugin is also installed. If the plugin has not been installed, main execution will halt and a message will be

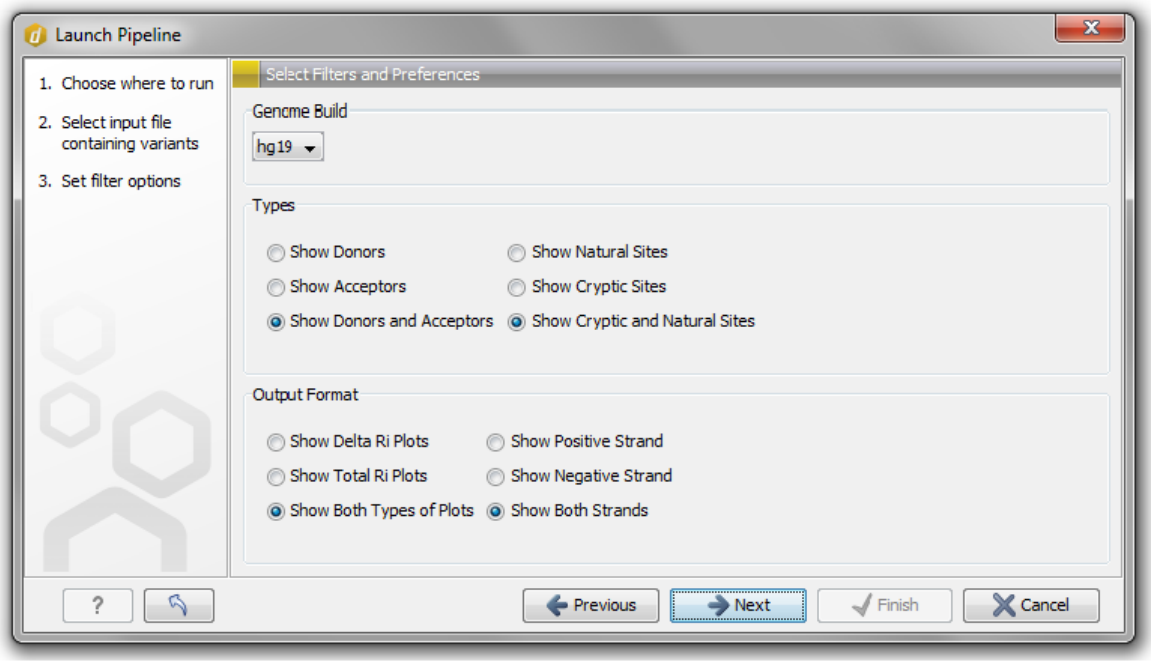


Figure 3. Shannon pipeline genome build, filtering, and display options.

This is a screenshot of the third wizard screen after selecting 'Launch Pipeline'. By default, the displayed choices are selected. Genome build may be hg18 or hg19. In the types frame, donor, acceptors, or both may be displayed to the user after pipeline execution. Similarly, cryptic sites, donor sites, or both types of sites may be displayed. In the 'Output Format' frame, the user can opt to view delta R_i plots, final R_i plots, or both. The user can opt to view tabular and plot output containing one or both strands.

displayed to the user (and in the error log), informing them that the appropriate database plugin must be installed.

3.3.1.6 LaunchPipelineAlgo class and the command-line

Development of this class required the creation of Perl code as well. Specifics of the necessary Perl scripts can be found in 3.2.1.1 and 3.2.1.2. The overall workflow of a Shannon pipeline execution from the Workbench is as follows: 1) Variants are imported using importer classes described in 3.3.1.1. 2) The user selects preferences using a wizard which are recorded using the LaunchPipelineParameters class and LaunchPipelineAlgo reads these parameters. 3) Perl and C code is automatically installed if necessary. 4) A parameters file is created and the command-line portion of the Shannon pipeline is executed taking these preferences into account. 5) Shannon pipeline results are imported. 6) Imported Shannon pipeline results (objects) can be viewed in standard CLC-Bio editors. The class LaunchPipelineAlgo is involved in steps 2-5.

Shannon pipeline C libraries must be installed before execution. Previously, a collection of Perl wrappers were designed to allow Perl code to make use of the C libraries. As a result of this interconnectivity, releasing compiled code would require separate distributions not only for different architectures (Linux 64bit, Linux 32bit, etc.) but also for different versions of Perl. Additionally, Perl Makefiles produce different executables based on whether the Perl installation used to created them is ‘threaded’ or ‘non-threaded’. Based on this variability, the choice was made to compile the code automatically on each machine. I created a Perl script that checks to see if libraries have been previously installed, and if not, creates a Perl Makefile and runs ‘make’ and ‘make install’ to install them. Additionally, standard output and standard error streams are redirected to files that can be viewed from the CLC-Bio Workbench in the case of error. LaunchPipelineAlgo executes this Perl script and waits to receive its return code. If the installation was unsuccessful, a non-zero code is returned. In particular, error code 100 is returned in the case of an unsuccessful installation. In this case, execution halts and “C library installation unsuccessful” is appended to the Shannon pipeline execution log. The libraries may have to be installed manually in this case. Otherwise, if the libraries are already installed or were installed successfully, execution continues. Library installation

has been tested on the following system configurations: Perl 5.8.8, 5.10.1, 5.12.3, 5.14.2, GCC 4.1.2, 4.2.1, 4.4.3, 4.6.3, Ubuntu 2.6.32, CentOS 2.6.18, Fedora 3.1.0-7, Mac OS X (Lion) 10.7.4.

A parameters file is created which can be accessed by the command-line portion of the Shannon pipeline. This file contains both user preferences and hard-coded settings for the pipeline. These parameters include the human genome version, Ensembl version, dbSNP version, distance from a modified cryptic site to attempt to locate a natural site, maximum distance a cryptic site can be from a natural site to display comparisons between them, the locations of Ensembl Gene 66, dbSNP 130/135, hg18, hg19, and splice site information weight matrices. The parameters file is placed in the directory containing command-line Shannon pipeline code.

After Shannon pipeline results are generated by the command-line code, results must be imported. Previously described tabular and plot import classes are invoked to accomplish this task. Standard output and standard error files are imported simply as ClcStrings. If any import is unsuccessful, execution will continue but a note will be made in the log describing the nature of the failure.

3.3.1.7 Help documentation

I implemented a series of help screens using JavaHelp. Help is accessed through a question mark on the bottom left of the preferences wizard (can be seen in Figure 3). An electronic version of help screens can be found online at http://www.clcbio.com/files/usermanuals/shannon_pipeline.pdf. Help sections include Quick Start, Tables, Plots, Tracks, FAQ, and requirements. In the online version, an additional section describing plugin installation is included.

3.3.1.8 Client-Server architecture and distributions

As documented in Table 1, three packages of Java classes form the basis of the client-server architecture of the Shannon pipeline. The base package is shared by both the client and server packages. This style avoids code duplication across client and server packages. Client and server distributions are created using Apache Ant (Ant). The client distribution

can be built in two different formats. The first format contains Perl and C command-line based code while it is removed in the other format. This results in a one client distribution which can be run independently from a server (standalone) and one which serves only as a front end for the server distribution (dependent). The dependent distribution is beneficial for those users running a client computer which does not meet the requirements to run the Shannon pipeline. The standalone distribution offers greater flexibility and allows pipeline execution on either the client or server machines.

I encountered several hurdles while attempting to create a standalone distribution. First, CLC-Bio requires that only one copy of the program can be executed at a time for each license a user has obtained. The Workbench has built-in functionality which disallows multiple executions of the same plugin on either the workbench or server. However, this functionality does not prevent a user from running the Shannon pipeline on their client machine and a server machine at the same time. To prevent this, I added code within LaunchPipelineAlgo which checks if the Shannon pipeline is already being executed on the client or server machines. If the pipeline is being executed in one location, submitting a job to the other is disallowed and an error message is displayed to the user.

Distributions created through Ant are sent to CLC-Bio where they are encoded and posted on their website and within the Workbench plugin section for download. C files are encrypted independently from CLC-Bio using openssl. Upon a signal from LaunchPipelineAlgo, if C libraries are not already installed they are decrypted, compiled, and the source is deleted. To use the Shannon pipeline, a user must enter a license key provided by CLC-Bio. In total, 5 distributions currently exist for download: 1) Shannon pipeline server 2) Shannon pipeline client (standalone) 3) Shannon pipeline client (dependent) 4) hg18 databases 5) hg19 databases. Uninstallation is accomplished through the plugin widget in the Workbench. In the case of hg18 and hg19 database plugins, databases are deleted upon uninstallation.

Early Shannon pipeline implementations did not run on Mac OS X. I made several minor modifications to support Mac execution. First, many OS X directories contain spaces. I updated the CommandLineExecutor class to submit command-line processes correctly if

spaces are present by submitting the command as an array. Doing this specifies where spaces in the command are by placing space delimited elements into the array. Spaces present in any single element of the array are automatically escaped properly. Secondly, most OS X distributions do not include GCC by default. GCC is required to compile the C libraries. Users must manually download either Xcode or other software containing GCC to be able to compile the libraries. This requirement is specified in the Shannon pipeline documentation.

3.4 Performance of the Shannon pipeline software

The unique identifier present in both the VCF or tab-delimited format serves several purposes. Input data may be stored in a hash allowing efficient annotation of individual variants or those originating from multiple exome or genome sequences. Given the minimum overhead from chromosome processing incurred to process each individual chromosome present in the input data (~1 hour if all chromosomes present in input file). This startup time is based largely on the annotation process. Unique identifiers allow input to be combined, thus reducing total run-time and required user interaction.

To assess performance, all point mutations detected in the complete genomes of the three cancer cell lines were analyzed using the pipeline. Variants in the cell lines U2OS (osteosarcoma-derived), A431 (epidermoid squamous carcinoma-derived) and U251 (glioblastoma-derived) were examined and filtered to create tractable sets of variants. Predicted splice-altering mutations not found in dbSNP135 (a list of ~54 million known nucleotide polymorphisms) and those with less than 1% average heterozygosity are reported (**Appendix A**).

The Shannon pipeline processes SNVs to identify and annotate splicing mutations with sufficient speed to analyze single or multiple genomes within a few hours. Analysis of all single nucleotide substitutions detected in the genome of the U2OS cell line (211,049 variants) is completed in 1 hour 12 minutes on an I7-based CPU in either Linux or Mac OS X (**Table 2**). The speed analysis is dependent on the number of chromosomes represented in the input data. The state machine facilitates the analysis of all variants on a single chromosome with the highest efficiency because genomic data for each

chromosome must be read and parsed. A complete analysis of 300 variants on a single small chromosome (*e.g.*, chromosome 22) can be completed in 5 minutes. Variants distributed throughout all chromosomes require at least one hour to process. The Shannon pipeline should be executed on a machine with sufficient RAM to store the largest human chromosomes in memory with each base requiring 17 bytes of memory (≥ 4 gigabytes). When all chromosomes are represented, increasing the number of mutations results in an approximately linear increase in actual computation time, after accounting for the overhead required for memory management of genome sequences and annotations. For example, 2 hours 35 minutes is required to analyse 1,872,893 sequence variants from the most recent data release on the Exome Variant Server (http://evs.gs.washington.edu/EVS/).

Increased speed comes at the expense of diminished ability to analyze complex mutations on the fly, such as insertions and deletions or multinucleotide substitutions. Such variation is significantly less common than SNPs in wildtype genome and exome sequences [38], but nevertheless can have consequences on gene function and phenotype. The ASSA server is capable of analyzing these categories of mutations; however it is considerably slower than the Shannon pipeline (30s per variant). In the future, the Shannon pipeline will be integrated with the ASSA server to examine complex variants seamlessly.

Table 2. Performance of Shannon Pipeline for mRNA splicing mutation prediction

Source of variants	Number of variants analyzed	Running time*
U2OS cell line	211,049	1h 12m
A431 cell line	290,589	1h 17m
U251 cell line	314,637	1h 20m
ESP 6500 Exomes	1,872,893	2h 35m

Note *Intel I7 CPU with 16 Gb RAM

4 Shannon pipeline - Results

4.1 Stratification of variants

Similar to ASSA, the pipeline analysis produces summary tables for different types of mutations (assuming each type is represented): 1) complete sets of all splicing variants, 2) mutations predicted to inactivate splice sites, 3) leaky splicing mutations that reduce but do not abolish splicing and 4) cryptic splice sites that are either activated, inactivated or reduced in strength. Inactivating variants are defined as those that reduce the R_i of the affected binding site below 1.6 bits [35]. Binding sites containing a leaky variant are defined as those, in which initial R_i is decreased upon mutation to $R_i > 1.6$. Finally, candidate cryptic sites encompass all sites with higher affinity for binding than a corresponding natural site based on comparison of their respective R_i values. Tabular data can be sorted by clicking the column header of each column. Data can be exported and viewed without modification in a spreadsheet program using CLC-Bio's built in export functionality.

The 5′ end of the first exon and the 3′ end of the last exon of a gene are not splice sites. They instead form the boundary of the gene. Therefore, the Shannon pipeline does not report mutations that affect their ΔR_i at these positions; the exception being genes that encode alternate splice forms using further upstream/downstream exons present in Ensembl 66. Variants which alter the strength of cryptic splice sites within the first and last exons are also considered. Use of a strengthened cryptic donor in the first exon or acceptor in the last exon could lead to a truncated exon. The Shannon pipeline considers the exonic cryptic sites of the opposite polarity (acceptors in first exons and donors for last exons), as their activation could potentially - but rarely - lead to the formation of a cryptic intron within these exons if a second pre-existing cryptic site of opposite polarity is present in the proper orientation.

Although Shannon pipeline output contains a vastly reduced number of potentially significant variants, further manual filtering is necessary to obtain the final set of functionally relevant sites. Pipeline output is generated for all variants that result in $\Delta R_i >$

±1 bit. One bit corresponds to an approximately 2 fold difference in binding affinity, which is the limit of detection of fold change by quantitative real-time polymerase chain reaction (qPCR) [39]. The user then filters out those variants least likely to be functionally relevant. For example, a natural site that has experienced an increase in information content will generally not be of interest. The increase will likely only serve to widen the existing gap in R_i between the natural and nearby cryptic sites. Thus, it is recommended those natural sites with positive ΔR_i values as well as cryptic sites with reductions in R_i value be removed. Pipeline generated annotations that are found in the tabular output help simplify the data filtering process. As discussed, tabular results are displayed in separate tables used to distinguish natural and cryptic splicing mutations. Recommended filters used for cryptic splicing mutations are based on criteria given in [34] (a) $\Delta R_i > 0$, (b) cryptic site is located within an exon or within an intron less than 300 bp from nearest natural site, (c) cryptic splice site R_i value exceeds the strength of the nearest natural site R_i of the same type and (d) intronic cryptic splice sites are selected 5′ to the exon if acceptors and 3′ to the exon, if donors. All reported variants are further categorized according to whether they had been previously reported or were novel by the Shannon pipeline. In **Table 3**, only novel and known variants < 1% average heterozygosity in dbSNP are reported. Variants < 1% average heterozygosity are more likely to be functionally significant due to selection (deleterious variants are selected against). Nevertheless, any threshold for filtering based on heterozygosity can be used by the user.

Filtering of cryptic splice sites exceeding the strength of and close to adjacent natural sites of the same phase eliminates many predicted unused cryptic sites with changes in R_i values. Finally, it is recommended that genes lacking HUGO-approved names or encoding non-coding RNAs, and pseudogenes should be filtered out. The manual filtering process (especially of cryptic splicing mutations) significantly enriches for likely mutations in the genomes of these cancer cell lines by the order of 10,000 fold.

Table 3. Enrichment for predicted splicing mutations after processing and filtering

Cell line	Initial variants analyzed	Novel Natural site	Novel Cryptic site	Natural site (SNP)*	Cryptic site (SNP)*	Overall Mutation fraction
A431	290,589	16	13	13	3	0.015%
U251	314,637	7	10	18	3	0.012%
U2OS	211,049	22	9	13	4	0.022%
Total	816,275	45	32	44	10	0.016%

Note *dbSNP135; <1**%** heterozygosity; minor allele

4.2 Displaying results

ΔR_i and final R_i values are plotted by chromosome location, similar to Manhattan-style representations, for either individual chromosomes or entire genomes. Hovering the cursor over data points generates tooltips containing information needed to find the complete entry within the corresponding tabular data. To locate interesting data points, a zoom function allows closer inspection of the plot. This visualization allows patterns to be observed and data points which stand out to be easily located and inspected more closely in tabular format or on the ASSA server.

Chromosome-specific, custom browser tracks indicating ΔR_i values in BED format are also generated. This enables visualization of predicted mutations in the context of other genome annotations, for example, mapped reads from RNA-seq, spliced expressed sequence tags (ESTs) and known mRNAs. **Figure 4** depicts three methods of displaying Shannon pipeline results.

4.3 Validation with RNA-seq expression data

RNA-seq analysis using published data from these cell lines [40] was used to compare Shannon pipeline results with expression data. TopHat [41] was executed with the following command-line options: -g 5 --solexa1.3-quals -p 8, and examined with the Integrative Genomics viewer (IGV) [42] to interrogate predictions made with the Shannon pipeline.

Several variants detected in genomes of U2OS, U251 and A431, which were predicted to affect splicing, were compared to the distribution of RNA-seq reads in their respective regions of the transcriptome. When interpreting these data, it is assumed that predicted mutations are present in a genetic background, in which the other parentally derived allele lacks the same variant (*i.e.*, heterozygous). Abnormal reads or exon skipping of the mutant allele is viewed in the context of a single allele and expected normal splicing of the corresponding exon. For mutations that are predicted to inactivate a splice site, it is assumed that a binomial distribution in the number of expected reads is present, based on

A

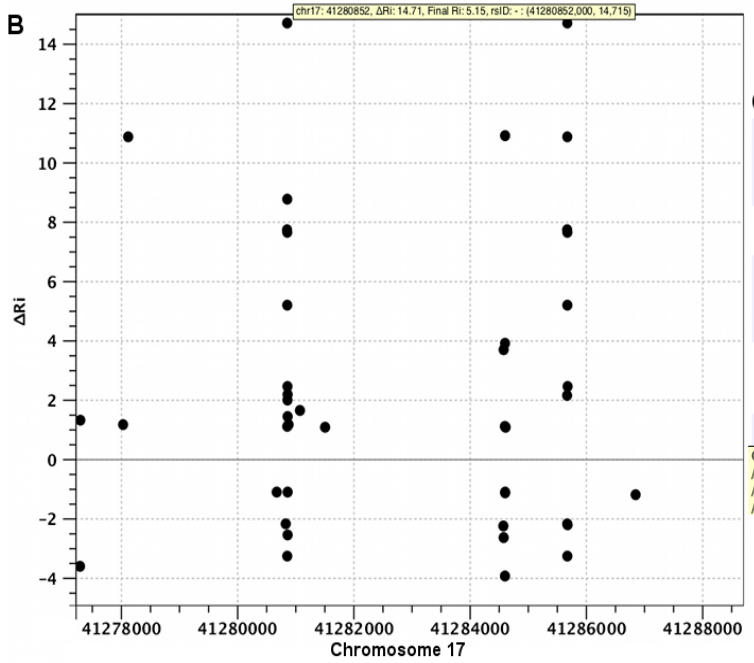
Rows: 134 / 22,197 Effect of variants on Ri and other relevant information Filter: Match any Match all

Dist. from nearest nat. site	doesn't contain	-
ΔRi	>	0
Cryptic Ri relative to nat.	=	GREATER
Ri-final	>	1.6

Apply

Chromoso...	Coordinate	Strand	Ri-initial	Ri-final	ΔRi	Type	Gene Name	Location	Location ...	Loc. Rel. t...	Dist. from...	Loc. of ne...	Ri of near...	Cryptic Ri ...	rsID if ava...	Average
1	1552560...	+	-16.41	2.22	18.63	DONOR	HCN3	CRYPTICS...	INTRONIC	3'-FLANKI...	263	1552557...	2.06	GREATER	rs14473...	0
2	2333068...	+	-8.17	10.46	18.63	DONOR	DIS3L2P1	CRYPTICS...	INTRONIC	3'-FLANKI...	19	2333068...	0.17	GREATER	rs790027	0.21293
5	1760831...	+	-13.73	4.91	18.63	DONOR	TSPAN17	CRYPTICS...	INTRONIC	3'-FLANKI...	48	1760831...	4.24	GREATER	rs6878977	0
6	44140320	+	-10.77	7.86	18.63	DONOR	CAPN11	CRYPTICS...	INTRONIC	3'-FLANKI...	162	44140158	6.5	GREATER	rs1418488	0.49578
7	1424695...	+	-12.86	5.77	18.63	DONOR	U66061....	CRYPTICS...	INTRONIC	3'-FLANKI...	85	1424694...	5.57	GREATER	rs14071...	0
9	88457937	+	-15.11	3.53	18.63	DONOR	RP11-21...	CRYPTICS...	INTRONIC	3'-FLANKI...	140	88457797	-37.04	GREATER	rs14201...	0
11	1234650...	+	-10.25	8.39	18.63	DONOR	GRAMD1B	CRYPTICS...	INTRONIC	3'-FLANKI...	161	1234648...	6.72	GREATER	-	-
12	8383433	+	-15.77	2.87	18.63	DONOR	ALG1L2	CRYPTICS...	INTRONIC	3'-FLANKI...	192	8383241	-43.03	GREATER	rs2970164	0.44444
12	97078749	+	-13.18	5.45	18.63	DONOR	C12orf63	CRYPTICS...	INTRONIC	3'-FLANKI...	235	97078514	5.45	GREATER	rs7306382	0.5
13	26104943	+	-12.97	5.66	18.63	DONOR	ATP8A2	CRYPTICS...	INTRONIC	3'-FLANKI...	144	26104799	3.91	GREATER	rs9581377	0.49931
18	12263067	+	-15.96	2.67	18.63	DONOR	CIDEA	CRYPTICS...	INTRONIC	3'-FLANKI...	98	12262969	1.45	GREATER	rs8090997	0.47363
19	56373578	+	-14.99	3.65	18.63	DONOR	NLRP4	CRYPTICS...	INTRONIC	3'-FLANKI...	52	56373526	3.65	GREATER	rs10853...	0.39669

B



C

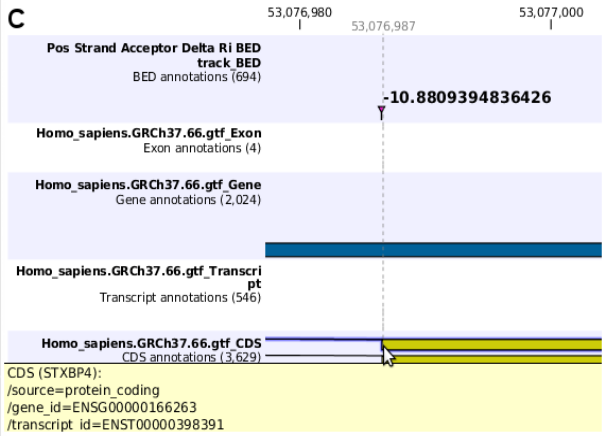


Figure 4. Twelve DNA sequences and their corresponding information changes.

The Shannon pipeline software generates the following types of output. **A.** Tabular results showing the first 12 of 134 changes in R_i values at different genomic coordinates predicted to be significant, after filtering for cryptic splicing mutations from all variants (n=22,197) in a complete genome sequence. The first filter eliminates exonic cryptic sites, the second selects cryptic sites with increased R_i values, the third ensures that the cryptic site is stronger than the corresponding natural site of the same phase and the final filter ensures that all remaining sites exceed the minimum R_i value of a functional splice site. **B.** Manhattan-like plot indicating the locations and changes in R_i of all variants which alter splice site information in a region within intron 1 of *BRCA1* (chr17:41277500-41288500) from different individuals with increased breast cancer risk. **C.** Custom track illustrating a cryptic splicing mutation detected in an ovarian serous carcinoma that inactivates the acceptor site of exon 4 in *STXBP4*, resulting in the activation a pre-existing, in frame, alternative splice site 6 nucleotides downstream.

the wild type allele. Natural splice site mutations are expected to significantly reduce the number of splice junction-spanning reads in relative to those in the adjacent exons, consistent with exon skipping. In some cases, intron inclusion adjacent to a splice site variant with lower R_i value may also be evidence of a splicing mutation. In U2OS, 10 of 13 novel inactivating variants found in mutated natural splice sites met these criteria, along with an additional 2 probable mutations (Appendix A, Table S1). The same criteria were met by 2 of 4 (with 1 additional probable) novel inactivating variants in U251 (Appendix A, Table S2), and 4 of 7 (with 1 additional probable) variants in A431 (Appendix A, Table S3).

Shannon pipeline predictions were supported by expression data for 1 of 7 activated cryptic site variants in U2OS, 1 of 14 variants in A431 and 0 of 10 in U251. Many of the predicted splice sites reside in intronic regions or alternative exons that map far upstream or downstream of constitutively expressed exons. They are unlikely to displace constitutive isoforms, since donor site recognition is processive [43] and the increased lengths of such cryptic exons would probably be suboptimal [44]. Often, these sites are associated with rare, alternatively spliced ESTs expressed in other tissues than these cell lines. Because these variants are often extra-exonic, changes in expression must be inferred indirectly from decreased read count, intron inclusion or increased exon skipping. Changes in reading frame from inclusion of out-of-phase intronic sequences may induce nonsense-mediated decay (NMD). Reads mapping to adjacent introns are expected to be reduced in number as a result of NMD. Sequencing reads that are concentrated in the intronic region adjacent to exon of interest are considered support for predicted mutations. NMD may also affect transcript read counts associated with severe leaky or inactivated natural donor sites, which produce exon skipping with frame-shifting. Several predicted splicing mutations confirmed by RNA-seq are well-known driver mutations that contribute to tumor phenotypes.

Interesting results include a unique natural donor site mutation within *RBBP8* (NM_203291.1:c.248G>A or chr18:20529676G>A; 6.2 $\rightarrow$ 3.2 bits [indicating the change in the R_i value of the donor site, before and after it is mutated]) in A431, a tumour suppressor gene mutated in numerous neoplasias with a role in endonucleolytic

processing of a covalent topoisomerase-DNA complexes. The mutation weakens but does not abolish the natural donor site from 6.2 to 3.2 bits. A cryptic mRNA splice form using a pre-existing donor site 24bp downstream to the weakened natural site is confirmed by RNA-seq (**Figure 5A**). The ASSA server predicts the activation of this intronic cryptic donor site, as well as a second site of equal strength further downstream to the mutated donor site (**Figure 5B**). There are a total of 56 reads that both encroach into the intron and overlap this variant. Forty-one of these cover the cryptic exon splice junction of interest (the aligned reads stop at the 3.2 bit cryptic site, which is 24 nt downstream of the natural site, and continue into the next natural exon). Thirty-one junction spanning reads also contain the A-allele. There are an additional 23 reads that cross into the intron, but do not extend as far as the cryptic site of interest. In 19 cases, these reads contain the A-allele. The remaining 4 intron-crossing reads which contain the G-allele appear to be misaligned, as they contain short matches ($\leq$ 3 nt) to the downstream exon. There are an additional 2 reads that span the junction between the downstream cryptic exon junction and the adjacent exon (31 nt downstream; also 3.2 bits). Finally, 12 reads are correctly spliced and contain the mutant A-allele, suggesting that the natural site is not completely inactivated by this nucleotide substitution, which is consistent with leaky splicing.

Changes in expression are also noted in other genes. *DDX11* is inactivated in U2OS (chr12:31242087T>G; 6.89 $\rightarrow$ -11.73 bits). DDX11 is a component of the cohesin complex which has a crucial role in chromosome segregation, and is essential for survival of advanced melanoma [45]. In U2OS, *WWOX*, a tumor suppressor gene in osteosarcoma [46], contains a leaky mutation (chr16:78312497C>A; 10.24 $\rightarrow$ 6.67 bits). Both alleles of *APIP*, an apoptosis associated gene, are inactivated in U251 (chr11:34905054G>C; 9.32 $\rightarrow$ 0.54 bits). Gene expression of *APIP* is down regulated in non-small cell lung carcinoma [47]. Amplification of *METTL2B*, which harbors a leaky mutation in U251 (chr7:128117227G>A; 5.48 $\rightarrow$ 2.47 bits), has been demonstrated in several cancers, including glioblastoma [48]. In A431, leaky mutations are also confirmed in the glioblastoma-initiating gene *TRRAP* (chr7:98533187T>G; 9.09 $\rightarrow$ 7.16 bits; [49]) and *USF1* (chr1:161013165G>T; 4.89 $\rightarrow$ 3.59 bits), which encodes a transcription regulator important for TGFβ2 expression in glioblastoma [50]. *SYNE2*, which is mutated in a

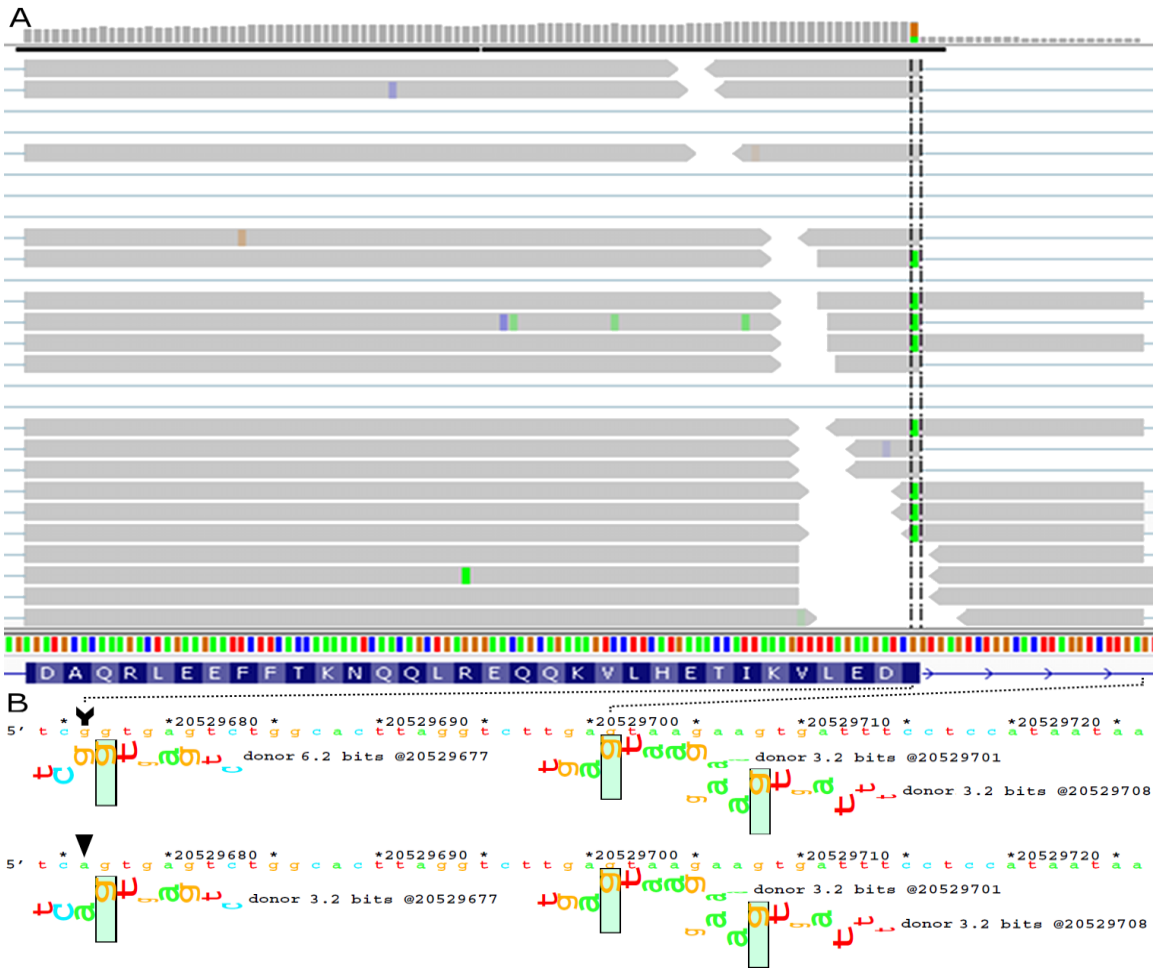


Figure 5. Predicted mutation splicing phenotype supported by RNA-seq

Predicted *RBBP8* splicing mutation, chr18:20529676G>A (NM_203291.1: c.248G>A), is related to transcripts mapped to this region. **A.** IVG genome browser display of read distribution at the exon 4/intron 4 junction. Green boxes within the vertical hashed lines indicate the presence of the A allele. **B.** The natural and cryptic splice sites illustrated by sequence walkers generated on the ASSA server. The arrow tail and head draw attention to the location and sequence of the reference and variant sequence. The mutation reduces the strength of the natural donor site from 6.2 to 3.2 bits. All but 3 of the 59 reads extending into the intron contain the variant allele, as indicated by the green positions within the reads. These reads extend into the exon and terminate at the closest intronic cryptic donor site (chr18:20529700). The mutated natural and cryptic sites are of equal strength, which explains splicing at both sites.

significant percentage of head and neck squamous cell carcinomas [51], contains an inactivating splice site variant in A431 (chr14:64669514T>A; 1.89 → -0.83 bits). *RRM2B*, an inducible DNA repair gene that has been implicated in squamous cell carcinoma [52], contains an inactivating mutation in A431 (chr8:103250667A>C; 3.6 → -15.02 bits). *SMARCD1*, encoding a chromatin modulator that interacts with nuclear receptor transcription factors, is also inactivated in A431 (chr12:50480538G>C; 8.46 → -3.21 bits), and has been shown to be mutated in hepato- and other carcinomas [53].

Several mutations were found in potential tumor-associated genes, with either suggestive or little supporting expression data. However, defects in many of these genes have been implicated in various neoplasias including glioblastoma, osteosarcoma, and epidermoid squamous carcinoma. In general, these were predicted leaky mutations, where effects (diminished read counts and exon skipping) were inferred against the confounding background of a presumably intact allele. Natural site mutations in *FANCD2* (NM_033084.3:c.3106-9T>A; 6.0 → 3.5 bits; delayed activation of the DNA damage response in gliomas [54]) and *MDC1* (NM_014641.2:c.2129-8G>C; 6.4 → 4.7 bits; mediator of the DNA damage checkpoint and underexpressed in many cancers [55]) were found in the U251 cells.

4.4 Characterization of defective pathways

Potential driver mutations affecting protein coding of genes from the A431, U2OS, and U251 cell lines have recently been reported [40]. Functionally significant driver mutations affecting splicing are expected to comprise many of the same pathways implicated by protein coding mutations that are predicted to be damaging. The gene set with combined driver point and copy number alteration was examined using Reactome [56]. Shannon pipeline results, supported by RNA-seq data, were added to gene sets proposed by [40] and the expanded gene set was examined with the overrepresentation analysis tool in Reactome. Of the genes containing transcript-validated splicing mutations, both datasets were consistent in 2 of 5 pathways in A431 (interferon signaling and cytokine signaling in immune system), 8 of 8 pathways in U2OS (cell cycle mitotic, cell cycle, DNA replication, mitotic M-M/G1 phases, M phase, kinetochore capture of astral microtubules,

mitotic prometaphase and apoptosis) and 0 of 2 pathways in U251. Affected pathways and relevant genes can be found in **Table 4**. The gene set including all inactivating and leaky variants (regardless of verification status) were found in 5 of 7 of the same pathways in A431 (additionally, a variant was found in the semaphorin interaction pathway), 8 of 12 of the same pathways in U2OS and 0 of 11 pathways in U251. In A431 and U2OS, these splicing mutation predictions enhance and strengthen the pathway analysis based on protein coding mutations alone.

Table 4. Enriched pathways containing genes predicted by the Shannon pipeline

Cell line	Genes from [40] in pathway	Additional gene in pathway predicted by the Shannon pipeline	Pathway name
A431	RANBP2, EIF4A2, NUP98	PIAS1	Interferon signaling
	RANBP2, EIF4A2, NUP98	PIAS1	Cytokine signaling in immune system
U2OS	RRM2, ZWINT, PLK1, PSMC3, AURKB, PKMYT1, TYMS, POLD2, CCNB2, MCM3, MCM5, UBE2C, CCNE2, KIF23, NEDD1, PRIM2, PSMD13, TUBA3D, MCM7, ERCC6L	CENPN	Cell cycle
	RRM2, ZWINT, PLK1, PSMC3, AURKB, PKMYT1, TYMS, POLD2, CCNB2, MCM3, MCM5, UBE2C, CCNE2, KIF23, NEDD1, PRIM2, PSMD13, TUBA3D, MCM7, ERCC6L	CENPN	Cell cycle, mitotic
	POLD2, MCM3, CCNB2, ZWINT, MCM5, PSMD13, PRIM2, PLK1, KIF23, TUBA3D, PSMC3, AURKB, MCM7, ERCC6L	CENPN	DNA replication
	MCM3, CCNB2, ZWINT, MCM5, PSMD13, PRIM2, PLK1, KIF23, TUBA3D, PSMC3, AURKB, MCM7, ERCC6L	CENPN	Mitotic M-M/G1 phases
	CCNB2, ZWINT, PLK1, KIF23, ERCC6L, TUBA3D, AURKB	CENPN	M phase
	ZWINT, PLK1, ERCC6L, TUBA3D, AURKB	CENPN	Kinetochore capture of astral microtubules
	ZWINT, PLK1, ERCC6L, TUBA3D, AURKB	CENPN	Mitotic prometaphase
	CAD, PSMD13, PSMC3	CTNNB1	Apoptosis

5 Discussion

Complete genome and exome sequencing detects numerous rare, non-recurrent mutations in different individuals with the same disease diagnosis. Making sense of genetically heterogeneous results requires detection and interpretation of mutations in many genomes. The identification of significant mutations in different driver genes, followed by a gene set or pathway analysis can reveal common, essential pathways in otherwise genetically heterogeneous diseases, such as cancer. Incomplete detection or reclassification of coding mutations will most likely impact the sensitivity of these analyses. Most existing methods to predict the effects of splice site variation lack scalability, transparency or portability, with respect to their scoring systems. Information content can be applied to any region of any adequately annotated genome. Change in information (ΔR_i) is a portable measure and its thermodynamic basis meaningfully estimates the effects of splicing variation. By contrast, other systems (*e.g.*, [57]) are not suited for genome scale analysis and produce results that are not directly related to splice site strength.

A recent study reported the genomic, transcriptomic and protein sequences in the cell lines that were the source of the data that I analyzed [40]. It described the same single splicing mutation in the *APIP* gene identified in the present study, but none of the others that predicted by the Shannon pipeline. Further, there was no overlap between the genes containing predicted protein coding mutations in [40] and those indicated from the current study. This was somewhat surprising, it was anticipated that some loss of function mutations in tumor suppressor genes would arise from compound heterozygosity. Instead, mutant genes from both studies tended to occur in the same pathways (for U2OS and A431).

Many of the predictions made by the Shannon pipeline were supported by the same RNA-seq data that identified only *APIP* [40]. Conventional splice junction mutation analysis of NGS data, which tends to emphasize only the significance of changes in

conserved splice junction, intronic dinucleotides does not appear to be as sensitive or comprehensive as the information-based Shannon pipeline [7]. Assuming the cell line genotypes faithfully reflect the tumor genetics, likely driver mutations in the tumors were missed. These genes contribute to the tumor signatures and in most instances, belong to major pathways that are dysfunctional in the tumor. A caveat is that many of these cancer-associated genes have been uncovered in other tumor types, rather than the tumors that gave rise to the cell lines studied here.

Many of the predicted mutations that are supported by expression data make sense in light of independent studies, which have suggested the same driver genes and pathways that are defective in these tumor types [51,58-60]. Note that the recommended filtering procedures eliminate and/or minimize inclusion of mutations in gene classes with no known connection to cancer disease etiology. The sensitivity and specificity of these predictions support use of the Shannon pipeline in other somatic genomic analyses, and possibly for a wider spectrum of heritable genetic disorders.

The interpretation of potential splicing mutations in complete genome data is also challenging because the source of annotations, Ensembl, contains many accurate but apparently irrelevant genomic features. These comprise exons called on the basis of a single or a few ESTs with deep intronic locations (relative to constitutive exons) [61,62], and predicted mutant ESTs that are in fact present in non- or low expression genes (due to tissue specificity of the gene). Such cases indicate that the predicted mutation acts only upon these alternative splice forms, rather than the major transcript produced by the gene containing this gene. Where the RNA-seq data are either insufficient or irrelevant, pseudogenes (or genes which are members of families containing pseudogenes) may contain mismapped reads for the non-functional copies that can produce false positive mutation calls. Automatic filtering of genes from the RNA-seq data prior to validating information-based predictions would significantly simplify post-hoc processing of the Shannon pipeline. Until such a workflow is available, individual predicted mutations have to be assessed manually, because cryptic sites that alter the strength of a “decoy” exon, while a technically legitimate result, is probably irrelevant as a potential disease-causing mutation.

6 Conclusion and future development

Accurate genome-scale mutation analysis of bulk sequencing data in a timeframe suitable for integration with prediction tools for other types of mutations will be needed to discover disease-related genes and pathways in large-scale genomic studies of many patients. The need to distinguish probable pathogenic from benign sequence changes has become acute [63]. Computing efficiency is essential for concurrent analysis of large sets of genome sequences [64]. As the volume of whole-genome next-generation sequencing data continues to increase, variants of unknown significance are also likely to become more common. The processing speed attained by the Shannon pipeline has distinct advantages over existing software for identifying functional non-coding variants detected in large multi-genomic analyses. The Shannon pipeline is not intended to replace laboratory examination of variants, rather it is meant to reduce time and money spent examining variants unlikely to be deleterious. In this way, researchers can use the pipeline to pinpoint those variants most likely to be deleterious for further examination in the laboratory. Thus far, the Shannon pipeline has been downloaded under an evaluation license by more than 200 researchers and has been purchased by the US National Cancer Institute. Some feedback has been received by researchers, in one case prompting me to fix a bug in the VCF import class which caused the import to fail if a '?' was present in the VCF header.

There are many opportunities for future improvements to the Shannon pipeline, some of which are already in the early stages of development: 1) I have begun converting Perl code to C++. This transition will improve performance, allow Windows compatibility without requiring specific versions of Perl (or any version of Perl), and facilitate the ability to distribute compiled code. This offers many advantages including increased code security and the removal of the requirement to download GCC or GCC equivalent on a Mac or Windows. 2) CLC-Bio has recently implemented their own VCF importer and exporter classes. This alleviates the need for the VCF importer I have developed and the Shannon pipeline must be modified to become compatible with the new classes. 3) I have

begun work on implementing a version of the Shannon pipeline used to analyze variants in the mouse genome. Since the pipeline was built generically, theoretically the pipeline can work on any sufficiently annotated genome with minor modifications. Work required to examine different genomes using the pipeline requires modifications to Ensembl Gene databases to make them compatible with the pipeline. 4) Implementation of indel analysis. C libraries exist which perform this analysis, however new code must be implemented to use those libraries and existing annotation code must be modified to properly annotate indels. 5) Automatic integration of pathway analysis would alleviate the need to manually use Reactome or other pathway analysis tools to examine pipeline results. 6) A weight matrix exists for an intronic sequence necessary for splicing called the branch point. The contribution of these sites are not observed in the current software, but functionality to do so may be included in a future release. 7) The ability for the user to submit a genome to act as the reference genome during calculation. In particular, this upgrade would be important for cancer genome analysis which involves comparison of variants present in a tumour against the reference sequence from normal DNA from the same individual.

Bibliography

1. Gullapalli RR, Desai KV, Santana-Santos L, Kant JA, Becich MJ. Next generation sequencing in clinical medicine: Challenges and lessons for pathology and biomedical informatics. *J Pathol Inform*. 2012;3:40-3539.103013. Epub 2012 Oct 31. doi: 10.4103/2153-3539.103013; 10.4103/2153-3539.103013.

2. Kavanagh D, Anderson HE. Interpretation of genetic variants of uncertain significance in atypical hemolytic uremic syndrome. *Kidney Int*. 2012;81(1):11-13. doi: 10.1038/ki.2011.330; 10.1038/ki.2011.330.

3. Spurdle AB, Healey S, Devereau A, et al. ENIGMA--evidence-based network for the interpretation of germline mutant alleles: An international initiative to evaluate risk and clinical significance associated with sequence variation in BRCA1 and BRCA2 genes. *Hum Mutat*. 2012;33(1):2-7. doi: 10.1002/humu.21628; 10.1002/humu.21628.

4. Adzhubei IA, Schmidt S, Peshkin L, et al. A method and server for predicting damaging missense mutations. *Nat Methods*. 2010;7(4):248-249. doi: 10.1038/nmeth0410-248; 10.1038/nmeth0410-248.

5. Nalla VK, Rogan PK. Automated splicing mutation analysis by information theory. *Hum Mutat*. 2005;25(4):334-342. doi: 10.1002/humu.20151.

6. Kumar A, White TA, MacKenzie AP, et al. Exome sequencing identifies a spectrum of mutation frequencies in advanced and lethal prostate cancers. *Proc Natl Acad Sci U S A*. 2011;108(41):17087-17092. doi: 10.1073/pnas.1108745108; 10.1073/pnas.1108745108.

7. O'Roak BJ, Vives L, Fu W, et al. Multiplex targeted sequencing identifies recurrently mutated genes in autism spectrum disorders. *Science*. 2012;338(6114):1619-1622. doi: 10.1126/science.1227764; 10.1126/science.1227764.

8. Churbanov A, Vorechovsky I, Hicks C. A method of predicting changes in human gene splicing induced by genetic variants in context of cis-acting elements. *BMC Bioinformatics*. 2010;11:22-2105-11-22. doi: 10.1186/1471-2105-11-22; 10.1186/1471-2105-11-22.

9. Churbanov A, Rogozin IB, Deogun JS, Ali H. Method of predicting splice sites based on signal interactions. *Biol Direct*. 2006;1:10. doi: 10.1186/1745-6150-1-10.

10. Yeo G, Burge CB. Maximum entropy modeling of short sequence motifs with applications to RNA splicing signals. *J Comput Biol*. 2004;11(2-3):377-394. doi: 10.1089/1066527041410418.

11. Reese MG, Eeckman FH, Kulp D, Haussler D. Improved splice site detection in genie. *J Comput Biol*. 1997;4(3):311-323.

12. Pertea M, Lin X, Salzberg SL. GeneSplicer: A new computational method for splice site prediction. *Nucleic Acids Res*. 2001;29(5):1185-1190.

13. Cooper TA, Mattox W. The regulation of splice-site selection, and its role in human disease. *Am J Hum Genet*. 1997;61(2):259-266. doi: 10.1086/514856.

14. Lopez-Bigas N, Audit B, Ouzounis C, Parra G, Guigo R. Are splicing mutations the most frequent cause of hereditary disease? *FEBS Lett*. 2005;579(9):1900-1903. doi: 10.1016/j.febslet.2005.02.047.

15. Schneider TD. Information content of individual genetic sequences. *J Theor Biol*. 1997;189(4):427-441. doi: 10.1006/jtbi.1997.0540.

16. Shannon CE. A mathematical theory of communication, part I. *Bell Systems Technical Journal*. 1948;27:379-423.

17. Shannon CE, Weaver W. *A mathematical model of communication.* Vol 27. Urbana, IL: University of Illinois Press; 1949:379-423.

18. Shultzaberger RK, Schneider TD. Using sequence logos and information analysis of lrp DNA binding sites to investigate discrepancies between natural selection and SELEX. *Nucleic Acids Res*. 1999;27(3):882-887.

19. Mucaki EJ, Ainsworth P, Rogan PK. Comprehensive prediction of mRNA splicing effects of BRCA1 and BRCA2 variants. *Hum Mutat*. 2011;32(7):735-742. doi: 10.1002/humu.21513; 10.1002/humu.21513.

20. Shirley BC, Mucaki EJ, Whitehead T, Costea PI, Akan P, Rogan PK. Interpretation, stratification and evidence for sequence variants affecting mRNA splicing in complete human genome sequences. *Genomics, Proteomics & Bioinformatics*. 2013(0):-. http://www.sciencedirect.com/science/article/pii/S1672022913000296. doi: 10.1016/j.gpb.2013.01.008".

21. Dorman SN, Shirley BC, Knoll JH, Rogan PK. Expanding probe repertoire and improving reproducibility in human genomic hybridization. *Nucleic Acids Res*. 2013. doi: 10.1093/nar/gkt048.

22. Mucaki EJ, Shirley BC, Rogan PK. Prediction of mutant mRNA splice isoforms by information theory-based exon definition. *Hum Mutat*. 2013. doi: 10.1002/humu.22277; 10.1002/humu.22277.

23. Venter JC, Adams MD, Myers EW, et al. The sequence of the human genome. *Science*. 2001;291(5507):1304-1351. doi: 10.1126/science.1058040.

24. Nilsen TW. The spliceosome: The most complex macromolecular machine in the cell? *Bioessays*. 2003;25(12):1147-1149. http://dx.doi.org/10.1002/bies.10394. doi: 10.1002/bies.10394.

25. Chen M, Manley JL. Mechanisms of alternative splicing regulation: Insights from molecular and genomics approaches. *Nat Rev Mol Cell Biol*. 2009;10(11):741-754. doi: 10.1038/nrm2777; 10.1038/nrm2777.

26. Stephens RM, Schneider TD. Features of spliceosome evolution and function inferred from an analysis of the information at human splice sites. *J Mol Biol*. 1992;228(4):1124-1136.

27. Padgett RA, Grabowski PJ, Konarska MM, Seiler S, Sharp PA. Splicing of messenger RNA precursors. *Annu Rev Biochem*. 1986;55:1119-1150. doi: 10.1146/annurev.bi.55.070186.005351.

28. Krawczak M, Reiss J, Cooper DN. The mutational spectrum of single base-pair substitutions in mRNA splice junctions of human genes: Causes and consequences. *Hum Genet*. 1992;90(1-2):41-54.

29. Berget SM. Exon recognition in vertebrate splicing. *J Biol Chem*. 1995;270(6):2411-2414.

30. Nakai K, Sakamoto H. Construction of a novel database containing aberrant splicing mutations of mammalian genes. *Gene*. 1994;141(2):171-177.

31. Divina P, Kvitkovicova A, Buratti E, Vorechovsky I. Ab initio prediction of mutation-induced cryptic splice-site activation and exon skipping. *Eur J Hum Genet*. 2009;17(6):759-765. doi: 10.1038/ejhg.2008.257; 10.1038/ejhg.2008.257.

32. Pierce JR. *An introduction to information theory: Symbols, signals \& noise*. Dover; 1980. http://books.google.ca/books?id=fXxde44_0zsC.

33. Schneider TD, Stormo GD, Gold L, Ehrenfeucht A. Information content of binding sites on nucleotide sequences. *J Mol Biol*. 1986;188(3):415-431.

34. Rogan PK, Faux BM, Schneider TD. Information analysis of human splice site mutations. *Hum Mutat*. 1998;12(3):153-171. doi: 2-I.

35. Rogan PK, Svojanovsky S, Leeder JS. Information theory-based analysis of CYP2C19, CYP2D6 and CYP3A5 splicing mutations. *Pharmacogenetics*. 2003;13(4):207-218. doi: 10.1097/01.fpc.0000054078.64000.de.

36. Schneider TD, Rogan PK, inventorsComputational analysis of nucleic acid information defines binding sites. patent 5867402. 1999, .

37. Danecek P, Auton A, Abecasis G, et al. The variant call format and VCFtools. *Bioinformatics*. 2011;27(15):2156-2158. doi: 10.1093/bioinformatics/btr330; 10.1093/bioinformatics/btr330.

38. Lescai F, Bonfiglio S, Bacchelli C, et al. Characterisation and validation of insertions and deletions in 173 patient exomes. *PLoS One*. 2012;7(12):e51292. doi: 10.1371/journal.pone.0051292; 10.1371/journal.pone.0051292.

39. Mucaki EJ, Rogan PK. Prediction and functional validation of expressed SNPs altering mRNA splicing. Poster presented at the American Society of Human Genetics Meeting; 2009 Oct 20-24; Honolulu HI.

40. Akan P, Alexeyenko A, Costea PI, et al. Comprehensive analysis of the genome transcriptome and proteome landscapes of three tumor cell lines. *Genome Med*. 2012;4(11):86. doi: 10.1186/gm387.

41. Trapnell C, Pachter L, Salzberg SL. TopHat: Discovering splice junctions with RNA-seq. *Bioinformatics*. 2009;25(9):1105-1111. doi: 10.1093/bioinformatics/btp120; 10.1093/bioinformatics/btp120.

42. Robinson JT, Thorvaldsdottir H, Winckler W, et al. Integrative genomics viewer. *Nat Biotechnol*. 2011;29(1):24-26. doi: 10.1038/nbt.1754; 10.1038/nbt.1754.

43. Robberson BL, Cote GJ, Berget SM. Exon definition may facilitate splice site selection in RNAs with multiple exons. *Mol Cell Biol*. 1990;10(1):84-94.

44. Sterner DA, Carlo T, Berget SM. Architectural limits on split genes. *Proc Natl Acad Sci U S A*. 1996;93(26):15081-15085.

45. Bhattacharya C, Wang X, Becker D. The DEAD/DEAH box helicase, DDX11, is essential for the survival of advanced melanomas. *Mol Cancer*. 2012;11:82-4598-11-82. doi: 10.1186/1476-4598-11-82; 10.1186/1476-4598-11-82.

46. Del Mare S, Kurek KC, Stein GS, Lian JB, Aqeilan RI. Role of the WWOX tumor suppressor gene in bone homeostasis and the pathogenesis of osteosarcoma. *Am J Cancer Res*. 2011;1(5):585-594.

47. Moravcikova E, Krepela E, Prochazka J, Rousalova I, Cermak J, Benkova K. Down-regulated expression of apoptosis-associated genes APIP and UACA in non-small cell lung carcinoma. *Int J Oncol*. 2012;40(6):2111-2121. doi: 10.3892/ijo.2012.1397; 10.3892/ijo.2012.1397.

48. Lee CH, Alpert BO, Sankaranarayanan P, Alter O. GSVD comparison of patient-matched normal and tumor aCGH profiles reveals global copy-number alterations predicting glioblastoma multiforme survival. *PLoS One*. 2012;7(1):e30098. doi: 10.1371/journal.pone.0030098; 10.1371/journal.pone.0030098.

49. Charles N, Holland EC. The perivascular niche microenvironment in brain tumor progression. *Cell Cycle*. 2010;9(15):3012-3021. doi: 10.4161/cc.9.15.12710; 10.4161/cc.9.15.12710.

50. Kingsley-Kallesen M, Luster TA, Rizzino A. Transcriptional regulation of the transforming growth factor-beta2 gene in glioblastoma cells. *In Vitro Cell Dev Biol Anim*. 2001;37(10):684-690. doi: 2.

51. Stransky N, Egloff AM, Tward AD, et al. The mutational landscape of head and neck squamous cell carcinoma. *Science*. 2011;333(6046):1157-1160. doi: 10.1126/science.1208130; 10.1126/science.1208130.

52. Sun Z, Yang P, Aubry MC, et al. Can gene expression profiling predict survival for patients with squamous cell carcinoma of the lung? *Mol Cancer*. 2004;3(1):35. doi: 10.1186/1476-4598-3-35.

53. Stephens PJ, Tarpey PS, Davies H, et al. The landscape of cancer genes and mutational processes in breast cancer. *Nature*. 2012;486(7403):400-404. doi: 10.1038/nature11017; 10.1038/nature11017.

54. Cappelli E, Vecchio D, Frosina G. Delayed formation of FancD2 foci in glioma stem cells treated with ionizing radiation. *J Cancer Res Clin Oncol*. 2012;138(5):897-899. doi: 10.1007/s00432-012-1217-z; 10.1007/s00432-012-1217-z.

55. Stewart GS, Wang B, Bignell CR, Taylor AM, Elledge SJ. MDC1 is a mediator of the mammalian DNA damage checkpoint. *Nature*. 2003;421(6926):961-966. doi: 10.1038/nature01446.

56. Vastrik I, D'Eustachio P, Schmidt E, et al. Reactome: A knowledge base of biologic pathways and processes. *Genome Biol*. 2007;8(3):R39. doi: 10.1186/gb-2007-8-3-r39.

57. Desmet FO, Hamroun D, Lalande M, Collod-Beroud G, Claustres M, Beroud C. Human splicing finder: An online bioinformatics tool to predict splicing signals. *Nucleic Acids Res*. 2009;37(9):e67. doi: 10.1093/nar/gkp215; 10.1093/nar/gkp215.

58. Hanahan D, Weinberg RA. Hallmarks of cancer: The next generation. *Cell*. 2011;144(5):646-674. doi: 10.1016/j.cell.2011.02.013; 10.1016/j.cell.2011.02.013.

59. Stratton MR, Campbell PJ, Futreal PA. The cancer genome. *Nature*. 2009;458(7239):719-724. doi: 10.1038/nature07943; 10.1038/nature07943.

60. Cancer Genome Atlas Research Network. Comprehensive genomic characterization defines human glioblastoma genes and core pathways. *Nature*. 2008;455(7216):1061-1068. doi: 10.1038/nature07385; 10.1038/nature07385.

61. Curwen V, Eyras E, Andrews TD, et al. The ensembl automatic gene annotation system. *Genome Res*. 2004;14(5):942-950. doi: 10.1101/gr.1858004.

62. Flicek P, Ahmed I, Amode MR, et al. Ensembl 2013. *Nucleic Acids Res*. 2013;41(Database issue):D48-55. doi: 10.1093/nar/gks1236; 10.1093/nar/gks1236.

63. Biesecker LG. Opportunities and challenges for the integration of massively parallel genomic sequencing into clinical practice: Lessons from the ClinSeq project. *Genet Med*. 2012;14(4):393-398. doi: 10.1038/gim.2011.78; 10.1038/gim.2011.78.

64. Richter BG, Sexton DP. Managing and analyzing next-generation sequence data. *PLoS Comput Biol*. 2009;5(6):e1000369. doi: 10.1371/journal.pcbi.1000369; 10.1371/journal.pcbi.1000369.

Appendices

Appendix A: Shannon pipeline output for the U2OS, A431, and U251 cell lines.

Table S1 Splicing mutations in cell line U2OS predicted by Shannon pipeline

Leaky	$R_{i,initial}$	$R_{i,final}$	ΔR_i	Site	Gene	rsID	Av. Hat.	Validated*
Novel Variants								
chr5:148630982G>A	10.04	7.03	-3.01	D	*ABLIM3*			Y
chr9:140507906C>G	9.45	6.83	-2.62	A	*ARRDC1*			LF
chr12:46355654A>C	7.66	5.61	-2.06	A	*SCAF11*			P
chr10:115470783A>C	8.72	6.80	-1.93	A	*RP11-211N11.5.1*			NE
chr16:81060105T>G	3.99	2.87	-1.12	A	*CENPN*			Y
Known Variants < 1% Av. Het.								
chr16:78312497C>A	10.24	6.67	-3.57	A	***WWOX***	rs8050128	0	Y
chr2:98177124T>G	7.96	4.70	-3.26	D	*ANKRD36B*	rs11681640	0	PS
chr5:78076488A>T	9.60	7.14	-2.45	A	*ARSB*	rs183651028	0	
chr22:36657628T>G	8.94	6.88	-2.06	A	*APOL1*	rs41368549	0	N
chr2:32961745T>A	8.99	7.14	-1.85	A	*TTC27*	rs10200333	0	P
chr15:68445912T>A	16.57	14.72	-1.85	A	*PIAS1*	rs11633620	0.005	Y^H
chr1:33318561T>C	5.86	4.55	-1.32	D	*S100PBP*	rs702836	0	N
chr17:73698557C>T	9.81	8.70	-1.12	A	*SAP30BP*	rs820232	0.009	Y

Inactivating	$R_{i,initial}$	$R_{i,final}$	ΔR_i	Site	Gene	rsID	Av. Het.	Validated
Novel Variants								
chrX:91518144T>G	9.64	-8.99	-18.62	D	*PCDH11X*			NE
chr17:47286205A>C	9.57	-9.06	-18.62	D	*GNGT2*			Y
chr3:41277214G>A	8.87	-2.02	-10.88	A	*CTNNB1*			Y

chr12:120613944A>C	8.78	-9.85	-18.62	D	*GCN1L1*			Y
chr3:52181002A>C	7.88	-10.75	-18.62	D	*POC1A*			Y
chr9:131133696T>G	7.88	-10.75	-18.62	D	*URM1*			P
chrX:49689926T>G	7.72	-10.90	-18.62	D	*CLCN5*			Y
chr2:118743543A>C	7.08	-11.54	-18.62	D	*CCDC93*			Y
chr12:31242087T>G	6.89	-11.73	-18.62	D	***DDX11***			Y
chr2:165600195A>C	6.36	-12.26	-18.62	D	*COBLL1*			Y
chr6:33256579A>C	5.79	-12.83	-18.62	D	*WDR46*			P
chr12:131487849T>G	5.32	-13.31	-18.62	D	*GPR133*			NE
chr11:20142118A>C	4.52	-14.11	-18.62	D	*NAV2-AS1*			N
chrX:153627303T>G	3.80	-14.82	-18.62	D	*RPL10*			N
chr3:119222366T>A	3.75	1.35	-2.40	A	*TIMMDC1*			Y
chr10:116698300T>G	3.21	-15.41	-18.62	D	*TRUB1*			N
chr13:111932609T>G	0.83	-1.52	-2.35	A	*ARHGEF7*			Y
Known Variants < 1% Av. Het.								
chr17:71229465G>A	7.82	-10.81	-18.63	D	*C17orf80*	rs113825288	0.007	Y
chr8:86131463A>T	7.07	-11.55	-18.62	D	*C8orf59*	rs67573812	0	NE
chr11:65624562A>C	5.84	-12.78	-18.62	D	*CFL1*	rs667555	0	N
chr11:20529886G>A	4.71	-13.92	-18.63	D	*PRMT3*	rs6483700	0.007	NE
chr6:88224673A>C	4.59	-14.04	-18.62	D	*RARS2*	rs77773960	0	Y
chr16:66547767T>C	2.15	-12.56	-14.71	A	*RP11-403P17.5.1*	rs2241619	0	Y

Cryptic Splicing	$R_{i,initial}$	$R_{i,final}$	ΔR_i	**Site**	**Gene**	**Location**	**rsID**	**Av. Het.**	**Validated**
Novel Variants									
chr9:94870122C>A	8.60	9.73	1.12	A	*SPTLC1*	EXON			N
chr3:39448804G>T	4.87	7.10	2.23	A	*RPSA*	INTRON			N

chr11:1782979A>C	4.07	5.19	1.11	A	*AC068580.6.1*	EXON			PS
chr15:75190136G>A	-10.80	3.92	14.71	A	*MPI*	EXON			N
chr1:153602415A>C	2.22	3.52	1.30	A	*S100A1*	EXON			N
chr1:44447781T>G	-0.43	3.17	3.59	D	*B4GALT2*	INTRON			N
Known Variants < 1% Av. Het.									
chr1:206647742A>G	6.67	9.68	3.01	D	*IKBKE*	EXON	rs1539242	0.006	Y
chr10:73039497G>A	-7.59	7.13	14.71	A	*UNC5B*	INTRON	rs41278006	0.006	N
chr9:131022776G>C	-14.10	4.53	18.63	D	*GOLGA2*	EXON	rs74686374	0.006	AE
chr3:12447814C>G	0.26	4.13	3.87	A	*PPARG*	EXON	rs1797895	0.005	N

* **Bolded** Gene names are previously established tumor driver genes. Legend to symbols in Validation column: Y: **yes validated**; Y^{H}: **yes validated**, based on HapMap Phase II Affymetrix Exon array; P: probable: consistent with mutation altering splicing, but not conclusive; N: not validated; NE: not expressed in cell line. Certain RNAseq results were not interpretable: LF: low fidelity splicing – significant intron inclusion; PS: multiple pseudogenes or duplicated exons; AE: rare alternate exon – insufficient data. Under site heading, D: donor, A: acceptor.

Table S2 Splicing mutations in cell line U251 predicted by Shannon pipeline

Leaky	$R_{i,initial}$	$R_{i,final}$	ΔR_i	Site	Gene	rsID	Av. Het.	Validated
Novel Variants								
chr3:10123021T>A	5.95	3.54	-2.41	A	*FANCD2*			N
Known Variants < 1% Av. Het.								
chr16:78312497C>A	10.24	6.67	-3.57	A	*WWOX*	rs8050128	0	NE
chr3:123554710C>T	8.47	5.29	-3.18	D	*MYLK*	rs144796555	0	NE
chr7:128117227G>A	5.48	2.47	-3.01	D	***METTL2B***	rs76349929	0	Y
chr6:121767966T>A	14.37	11.77	-2.60	A	*GJA1*	rs56199702	0	N
chr16:88051014A>G	6.87	4.34	-2.52	D	*BANP*	rs6540151	0	NE
chr7:98533187T>G	9.09	7.16	-1.93	A	***TRRAP***	rs62472016	0	Y
chr15:68445912T>A	16.57	14.72	-1.85	A	*PIAS1*	rs11633620	0.005	AM
chr2:32961745T>A	8.99	7.14	-1.85	A	*TTC27*	rs10200333	0	P
chr6:30679289G>C	6.44	4.72	-1.72	A	*MDC1*	rs147822906	0	P
chr1:65830299T>G	16.58	15.06	-1.52	A	*DNAJC6*	rs2296479	0	P
chr6:30679289G>C	3.41	1.96	-1.45	A	*MDC1*	rs147822906	0	NE
chrX:119402264A>C	11.63	10.30	-1.33	A	*FAM70A*	rs41300936	0	Y
chr1:33318561T>C	5.86	4.55	-1.32	D	*S100PBP*	rs702836	0	NE
chr2:37265193A>C	10.76	9.64	-1.12	A	*HEATR5B*	rs139662639	0	NE
chr17:73698557C>T	9.81	8.70	-1.12	A	*SAP30BP*	rs820232	0.009	NE
chr7:74152376T>C	7.73	6.64	-1.09	A	*GTF2I*	rs810377	0	N

Inactivating	$R_{i,initial}$	$R_{i,final}$	ΔR_i	Site	Gene	rsID	Av. Het.	Validated
Novel Variants								
chr14:32142782T>G	10.35	-8.27	-18.62	D	*NUBPL*			NE

chr19:33904479A>C	10.07	-8.55	-18.62	D	*PEPD*				N
chr11:34905054G>C	9.32	0.54	-8.78	A	***APIP***				Y
chr1:43308444A>C	9.07	-9.55	-18.62	D	*RP11-342M1.4.1*				NE
chr12:4700466T>G	6.83	-11.79	-18.62	D	*DYRK4*				P
chr3:40570938T>G	5.73	-12.90	-18.62	D	*ZNF621*				N
Known Variants < 1% Av. Het.									
chr12:122740009C>T	4.77	-6.11	-10.88	A	*RP11-512M8.6.1*	rs10744155	0		NE
chr6:88224673A>C	4.59	-14.04	-18.62	D	*RARS2*	rs77773960	0		Y

Cryptic Splicing	$R_{i,initial}$	$R_{i,final}$	ΔR_i	**Site**	**Gene**	**Location**	**rsID**	**Av. Het.**	**Validated**
Novel Variants									
chr14:75128861G>C	-2.43	9.24	11.67	A	*KIAA0317*	EXON			N
chr21:45539300A>G	4.77	7.77	3.01	D	*PWP2*	EXON			N
chr19:620369T>G	5.92	7.76	1.84	A	*POLRMT*	EXON			N
chr2:73228685T>G	4.16	6.00	1.84	A	*SFXN5*	EXON			N
chr13:73369248A>T	2.91	4.92	2.01	A	*PIBF1*	INTRON			N
chr17:30303021A>G	1.45	4.63	3.18	D	*SUZ12*	EXON			N
chr22:50954300T>G	-14.18	4.45	18.63	D	*NCAPH2*	INTRON			N
chr8:144876251G>C	0.15	3.88	3.73	D	*SCRIB*	EXON			NE
chr10:135104007G>A	1.62	2.71	1.09	A	*TUBGCP2*	EXON			NE
chr10:70661701T>G	-16.58	2.06	18.63	D	*DDX50*	INTRON			N
Known Variants < 1% Av. Het									
chr2:204150427G>C	4.03	5.50	1.47	A	*CYP20A1*	EXON	rs144732080	0.002	N
chr3:12447814C>G	0.26	4.13	3.87	D	*PPARG*	EXON	rs1797895	0.005	NE
chr13:20611221G>A	-0.56	2.04	2.61	D	*ZMYM2*	INTRON	rs75747995	0.009	N

* **Bolded** Gene names are previously established tumor driver genes. Legend to symbols in Validation column: Y: **yes validated**; P: probable: consistent with mutation altering splicing, but not conclusive; N: not validated; NE: not expressed in cell line. Certain RNAseq results were not interpretable; AM: ambiguous due to significant exon skipping/alternative splicing throughout gene. Under site heading, D: donor, A: acceptor.

Table S3 Splicing mutations in cell line A431 predicted by Shannon pipeline

Leaky	$R_{i,initial}$	$R_{i,final}$	ΔR_i	Site	Gene	rsID	Av. Het.	Validated
Novel Variants								
chr18:20529676G>A	6.17	3.16	-3.01	D	***RBBP8***			Y
chr1:179310449A>G	10.01	7.40	-2.61	D	*SOAT1*			P
chr1:12318166T>C	11.69	9.97	-1.72	D	*VPS13D*			Y
chr1:161013165G>T	4.89	3.59	-1.30	D	*USF1*			Y
chr1:36360707C>A	8.67	7.56	-1.11	A	*EIF2C1*			N
Known Variants < 1% Av. Het.								
chr2:136148401A>T	21.77	18.95	-2.82	A	*ZRANB3*	rs75842485	0	P
chr1:169272451A>T	9.96	8.49	-1.47	A	*NME7*	rs10800427	0	P
chr2:32961745T>A	8.99	7.14	-1.85	A	*TTC27*	rs10200333	0	Y
chr22:36657628T>G	8.94	6.88	-2.06	A	*APOL1*	rs41368549	0	Y
chr7:29700059G>T	9.43	5.83	-3.59	D	*AC007276.5.1*	rs74896403	0	PS
chr2:98177124T>G	7.96	4.70	-3.26	D	*ANKRD36B*	rs11681640	0	PS
chr1:33318561T>C	5.86	4.55	-1.32	D	*S100PBP*	rs702836	0	AE
chr7:75628461A>T	6.76	4.35	-2.41	A	*STYXL1*	rs4728538	0	AE
chr15:68445912T>A	16.57	14.72	-1.85	A	*PIAS1*	rs11633620	0.005	Y
chr17:73698557C>T	9.81	8.70	-1.12	A	*SAP30BP*	rs820232	0.009	P

Inactivating	$R_{i,initial}$	$R_{i,final}$	ΔR_i	Site	Gene	rsID	Av. Het.	Validated
Novel Variants								
chr1:12064177T>G	9.57	-9.06	-18.62	D	*MFN2*			Y
chr12:50480538G>C	8.46	-3.21	-11.67	A	***SMARCD1***			Y
chr11:62341301A>C	7.84	-10.78	-18.62	D	*EEF1G*			P

chr2:213886187A>C	7.74	-10.88	-18.62	D	*AC093865.1.1*			PS
chr2:165600195A>C	6.36	-12.26	-18.62	D	*COBLL1*			N
chr11:20142118A>C	4.52	-14.11	-18.62	D	*NAV2-AS1*			PS
chr7:98591160G>C	4.33	-7.34	-11.67	A	*TRRAP*			Y
chr8:103250667A>C	3.60	-15.02	-18.62	D	*RRM2B*			Y
chr20:55045807G>A	3.41	0.40	-3.01	D	*C20orf43*			PS
chr22:36722714G>C	2.90	1.18	-1.72	A	*MYH9*			N
chr12:107374398C>T	2.11	-16.52	-18.63	D	*MTERFD3*			N
Known Variants < 1% Av. Het.								
chr12:122740009C>T	4.77	-6.11	-10.88	A	*RP11-512M8.6.1*	rs10744155	0	PS
chr11:20529886G>A	4.71	-13.92	-18.63	D	*PRMT3*	rs6483700	0.007	PS
chr14:64669514T>A	1.89	-0.83	-2.72	A	*SYNE2*	rs189611387	0	Y

Cryptic Splicing	$R_{i,initial}$	$R_{i,final}$	ΔR_i	**Site**	**Gene**	**Location**	**rsID**	**Av. Het.**	**Validated**
Novel Variants									
chr5:177637135C>G	5.83	9.57	3.73	D	*HNRNPAB*	EXON			N
chr16:57180998G>C	-0.56	8.22	8.78	A	*CPNE2*	EXON			N
chr16:27790837G>C	-1.56	7.22	8.78	A	*KIAA0556*	EXON			N
chr19:14000568T>G	5.85	7.15	1.30	A	*C19orf57*	EXON			N
chr8:133822931G>C	5.02	6.47	1.45	A	*PHF20L1*	EXON			N
chr1:101437592A>C	4.63	6.36	1.73	A	*SLC30A7*	INTRON			N
chr10:3191408C>A	5.02	6.35	1.33	D	*PITRM1*	EXON			N
chr14:103871339A>G	-4.68	6.20	10.88	A	*MARK3*	INTRON			N
chr21:47676584A>G	4.29	5.89	1.60	D	*MCM3AP*	EXON			N
chr16:27790837G>C	3.15	5.77	2.62	A	*KIAA0556*	EXON			N
chr22:20113022T>G	-12.96	5.67	18.63	D	*RANBP1*	EXON			N

chr6:7227103T>G	-14.30	4.34	18.63	D	*RREB1*	INTRON			N
chr16:90051055G>A	1.05	3.58	2.52	D	*AFG3L1P*	EXON			PS
Known Variants < 1% Av. Het.									
chr1:206647742A>G	6.67	9.68	3.01	D	*IKBKE*	EXON	rs1539242	0.006	N
chr3:12447814C>G	0.26	4.13	3.87	D	*PPARG*	EXON	rs1797895	0.005	LF
chr2:232087474A>G	-15.64	3.00	18.63	D	*ARMC9*	EXON	rs1626450	0.0004	N

* **Bolded** Gene names are previously established tumor driver genes. Legend to symbols in Validation column: Y: **yes validated**; P: probable: consistent with mutation altering splicing, but not conclusive; N: not validated; LF: low fidelity splicing – significant intron inclusion; PS: multiple pseudogenes or duplicated exons; AE: rare alternate exon – insufficient data. Under site heading, D: donor, A: acceptor.

Curriculum Vitae

Name:	Ben Chambers Shirley
Post-secondary Education and Degrees:	Fanshawe College London, Ontario, Canada 2003-2005 Diploma (Computer programmer)
	The University of Western Ontario London, Ontario, Canada 2007-2011 B.Sc. (Computer Science – Bioinformatics)
	The University of Western Ontario London, Ontario, Canada 2011-Present M.Sc. (Computer Science – Bioinformatics)
Honours and Awards:	National Sciences and Engineering Research Council of Canada Undergraduate Student Research Award (NSERC – USRA) 2010
	University of Western Ontario Gold Medal – Bioinformatics 2011
	Computing Research Association Outstanding Undergraduate Researcher Award (Honorable Mention) 2011
	Mitacs-Accelerate Research internship 2012
	Ontario Graduate Scholarship (OGS) 2012-2013
Related Work Experience	Teaching Assistant – Computer Science The University of Western Ontario 2011-2012

Presentations:

Shirley BC. Finding non-coding mRNA splicing variants using the Shannon Human Splicing Pipeline. Hosted by Ontario Genomics Institute (OGI). [Video webcast] 2013.

Rogan PK, Mucaki EJ, Stuart A, Dovigi E, Viner C, Shirley BC, Knoll JH, Ainsworth P. Strategy for identification, prediction, and prioritization of non-coding variants of

uncertain significance in heritable breast cancer. Poster presented at Human Genome Meeting/International Congress of Genetics 2013 in Singapore by Rogan PK.

Shirley BC, Mucaki EJ, Akan P, Rogan PK. Genome-wide detection of mRNA splicing mutations using information theory-based binding site models. Poster presented at Bio-IT World 2012 in Boston, Massachusetts by Rogan PK, the Great Lakes Bioinformatics Conference (GLBIO) 2012 in Ann Arbor, Michigan by Shirley BC, and SHARCNET Research Day 2012 in Guelph, Ontario by Shirley BC.

Dorman SN, Shirley BC, Caminsky NG, Mucaki EJ, Khan WA, Guo L, Knoll JH, Rogan PK. Next generation genomic microarrays and custom FISH probes for molecular cytogenetic analysis designed by ab initio sequence analysis. Poster presented at the 12th International Congress of Human Genetics/ASHG 61st Annual Meeting 2011 in Montreal, Canada and the London Health Research Day 2012 in London, Canada by Dorman SN.

Dorman SN, Caminsky NG, Shirley BC, Khan WA, Guo L, Knoll JH, Rogan PK. Development of single copy FISH probes to detect chromosomal abnormalities in small tumour suppressor and oncogenes. Poster presented at the Department of Oncology Research & Education Day 2011 in London, Canada by Dorman SN.

Dorman SN, Shirley BC, Caminsky NG, Rogan PK. Developing single copy probes for fluorescence in-situ hybridization and array comparative genomic hybridization microarray design. Poster presented at the 6th Annual Canadian Student Conference on Biomedical Computing and Engineering 2011 in London, Canada by Dorman SN.

Shirley BC, Dorman SN, Patrick JC, Rogan PK. Definition of unique intervals in genomes through novel ab initio copy number determination. Poster presented at the 6th Annual Canadian Student Conference on Biomedical Computing and Engineering 2011 in London, Canada by Shirley BC.

Patrick JC, Shirley BC, Dorman SN, Rogan PK. Identifying unique sequences directly from the human genome reference. Poster presented at the American Society of Human Genetics 60th Annual Meeting 2010 in Washington, DC and the Canada Research Chair 10th Anniversary Conference 2010 in Toronto, Canada by Rogan PK.

Publications:

Shirley BC, Mucaki EJ, Whitehead T, Costea PI, Akan P, Rogan PK. Interpretation, Stratification and Evidence for Sequence Variants Affecting mRNA Splicing in Complete Human Genome Sequences. Genomics Proteomics Bioinformatics 2013 Jan. doi: 10.1016/j.gpb.2013.01.008

Mucaki EJ, Shirley BC, Rogan PK. Prediction of Mutant mRNA Splice Isoforms by Information Theory-Based Exon Definition. Hum Mutat 2013 Jan. doi: 10.1002/humu.22277.

Dorman SN, Shirley BC, Knoll JH, Rogan PK. Expanding probe repertoire and improving reproducibility in human genomic hybridization. Nucleic Acids Res 2013 Feb. doi: 10.1093/nar/gkt048.