RegTools: Integrated analysis of genomic and transcriptomic data for the discovery of splicing variants in cancer

Kelsy C. Cotto[1,2,†], Yang-Yang Feng[2,†], Avinash Ramu[3], Zachary L. Skidmore[1,2], Jason Kunisaki[2], Megan Richters[1,2], Sharon Freshour[1,2], Yiing Lin[4], William C. Chapman[4], Ravindra Uppaluri[5,6], Ramaswamy Govindan[1,7], Obi L. Griffith[1,2,3,7*], Malachi Griffith[1,2,3,7*]

[†] denotes co-first authors.
[*] denotes corresponding authors.

Correspondence to Obi L. Griffith (obigriffith@wustl.edu) and Malachi Griffith (mgriffit@wustl.edu).

Affiliations:
1. Division of Oncology, Department of Medicine, Washington University School of Medicine, St. Louis, MO, USA
2. McDonnell Genome Institute, Washington University School of Medicine, St. Louis, MO, USA
3. Department of Genetics, Washington University School of Medicine, St. Louis, MO, USA
4. Department of Surgery, Washington University School of Medicine, St. Louis, MO, USA
5. Department of Surgery, Brigham and Women's Hospital, Boston, MA, USA
6. Department of Medical Oncology, Dana-Farber Cancer Institute, Boston, MA, USA
7. Siteman Cancer Center, Washington University School of Medicine, St. Louis, MO, USA

Abstract

Somatic mutations in non-coding regions and even in exons may have unidentified regulatory consequences which are often overlooked in analysis workflows. Here we present RegTools (www.regtools.org), a free, open-source software package designed to integrate analysis of somatic variants from genomic data with splice junctions from transcriptomic data to identify variants that may cause aberrant splicing. RegTools was applied to over 9,000 tumor samples with both tumor DNA and RNA sequence data. We discovered 235,778 events where a variant significantly increased the splicing of a particular junction, across 158,200 unique variants and 131,212 unique junctions. To characterize these somatic variants and their associated splice isoforms, we annotated them with the Variant Effect Predictor (VEP), SpliceAI, and Genotype-Tissue Expression (GTEx) junction counts and compared our results to other tools that integrate genomic and transcriptomic data. While certain events can be identified by the aforementioned tools, the unbiased nature of RegTools has allowed us to identify novel splice variants and previously unreported patterns of splicing disruption in known cancer drivers, such as *TP53*, *CDKN2A*, and *B2M,* as well as in genes not previously considered cancer-relevant, such as *RNF145*.

Introduction

Alternative splicing of messenger RNA allows a single gene to encode multiple gene products, increasing a cell's functional diversity and regulatory precision. However, splicing malfunction can lead to imbalances in transcriptional output or even the presence of novel oncogenic transcripts[1]. The interpretation of variants in cancer is frequently focused on direct protein-coding alterations[2]. However, most somatic mutations arise in intronic and intergenic regions, and exonic mutations may also have unidentified regulatory consequences[3,4,5,6]. For example, mutations can affect splicing either in trans, by acting on splicing effectors, or in cis, by altering the splicing signals located on the transcripts themselves[7].

Increasingly, we are identifying the importance of splice variants in disease processes, including in cancer[8,9]. However, our understanding of the landscape of these variants is currently limited, and few tools exist for their discovery. One approach to elucidating the role of splice variants has been to predict the strength of putative splice sites in pre-mRNA from genomic sequences, such as the method used by the SpliceAI tool[10–13]. With the advent of efficient and affordable RNA-seq, we are also seeing the complementary approach of evaluating alternative splicing events (ASEs) directly from RNA sequencing data. Various tools exist which allow the identification of significant ASEs from transcript-level data within sample cohorts, including SUPPA2 and SPLADDER[14,15]. Many of these tools have also evaluated the role of trans-acting splice mutations[16]. However, few tools are directed at linking specific aberrant RNA splicing events to specific genomic variants in cis to investigate the splice regulatory impact of these variants. Those few relevant tools that do exist have significant limitations that preclude them from broad applications. The sQTL-based approach taken by LeafCutter and other tools is designed for relatively frequent single-nucleotide polymorphisms. It is thus ill-suited to studying somatic variants, or any case in which the frequency of a particular variant is very low (often unique) in a given sample population[17–19]. Recent tools that have been created for large-scale analysis of cancer-specific data, such as MiSplice and Veridical, ignore certain types of ASEs, are tailored to specific analysis strategies and sets of hypotheses, or are otherwise inaccessible to the end-user due to issues such as lack of documentation, difficulty with installation and integration with existing pipelines, limited computing efficiency, or licensing issues[20–22]. To address these needs, we have developed RegTools, a free, open-source (MIT license) software package that is well-documented, modularized for ease of use, and designed to efficiently identify potential cis-acting splice-relevant variants in tumors (www.regtools.org).

RegTools is a suite of tools designed to aid users in a broad range of splicing-related analyses. At the highest level, it contains three sub-modules: a variants module to annotate variant calls with respect to their potential splicing relevance, a junctions module to analyze aligned RNA-seq data and associated splicing events, and a cis-splice-effects module that integrates genomic variant calls and transcriptomic sequencing data to identify potential splice-altering variants. Each sub-module contains one or more commands, which can be used individually or integrated into regulatory variant analysis pipelines.

To demonstrate the utility of RegTools in identifying potential splice-relevant variants from tumor data, we analyzed a combination of data available from the McDonnell Genome Institute (MGI) at Washington University School of Medicine and The Cancer Genome Atlas (TCGA) project. In

total, we applied RegTools to 9,173 samples across 35 cancer types. We contrasted our results with other tools that integrate genomic and transcriptomic data to identify potential splice altering variants, specifically Veridical, MiSplice, and SAVNet[20,21,23]. Novel junctions identified by RegTools were compared to data from The Genotype-Tissue Expression (GTEx) project to assess whether these junctions are present in normal tissues[24]. Variants significantly associated with novel junctions were processed through VEP and Illumina's SpliceAI tool to compare our findings with splicing consequences predicted based on the variant information alone[13,25]. With this additional analysis, we were able to more easily identify both variants in known cancer drivers, whose splicing consequences have not been previously reported in the literature, and potentially novel cancer drivers, whose disruption relies on splice-altering mutations

Results

The RegTools tool suite supports splice regulatory variant discovery by the integration of genome and transcriptome data.

RegTools is a suite of tools designed to aid users in a broad range of splicing-related analyses. The variants module contains the annotate command. The *variants annotate* command takes a VCF of somatic variant calls and a GTF of transcriptome annotations as input. RegTools does not have any particular preference for variant callers or reference annotations. Each variant is annotated by RegTools with known overlapping genes and transcripts, and is categorized into one of several user-configurable "variant types", based on position relative to the edges of known exons. The variant type annotation depends on the stringency for splicing-relevance that the user sets with the "splice variant window" setting. By default, RegTools marks intronic variants within 2 bp of the exon edge as "splicing intronic", exonic variants within 3 bp as "splicing exonic", other intronic variants as "intronic", and other exonic variants simply as "exonic." RegTools considers only "splicing intronic" and "splicing exonic" as important. To allow for discovery of an arbitrarily expansive set of variants, RegTools allows the user to customize the size of the exonic/intronic windows individually (e.g. -i 50 -e 5 for intronic variants 50 bp from an exon edge and exonic variants 5 bp from an exon edge) or even consider all exonic/intronic variants as potentially splicing-relevant (e.g. -E or -I) (**Figure 1A**).

The junctions module contains the extract and annotate commands. The *junctions extract* command takes an alignment file containing aligned RNA-seq reads, infers the exon-exon boundaries based on the CIGAR strings[26], and outputs each "junction" as a feature in BED12 format. The *junctions annotate* command takes a file of junctions in BED12 format (such as the one output by *junctions extract)*, a FASTA file containing the reference genome, and a GTF file containing reference transcriptome annotations and generates a TSV file, annotating each junction with: the number of acceptor sites, donor sites, and exons skipped, and the identities of known overlapping transcripts and genes. We also annotate the "junction type", which denotes if and how the junction is novel (i.e. different compared to provided transcript annotations). If the donor is known, but the acceptor is not or vice-versa, it is marked as "D" or "A", respectively. If both are known, but the pairing is not known, it is marked as "NDA", whereas if both are

unknown, it is marked as "N". If the junction is not novel (i.e. it appears in at least one transcript in the supplied GTF), it is marked as "DA" (**Figure 1B**).

The cis-splice-effects module contains the *identify* command, which identifies potential splice-altering variants from sequencing data. The following are required as input: a VCF file containing variant calls, an alignment file containing aligned RNA-sequencing reads, a reference genome FASTA file, and a reference transcriptome GTF file. The *identify* pipeline internally relies on *variants annotate*, *junctions extract*, and *junctions annotate* to output a TSV containing junctions proximal to putatively splicing-relevant variants. The *identify* pipeline can be customized using the same parameters as in the individual commands. Briefly, *cis-splice-effects identify* first performs *variants annotate* to determine the splicing-relevance of each variant in the input VCF. For each variant, a "splice junction region" is determined by finding the largest span of sequence space between the exons that flank the exon associated with the variant. From here, *junctions extract* identifies splicing junctions present in the RNA-seq BAM. Next, *junctions annotate* labels each extracted junction with information from the reference transcriptome as described above and its associated variants based on splice junction region overlap (**Figure 1C**).

For our analysis, we annotated the pairs of associated variants and junctions identified by RegTools, which we refer to as "events", with additional information such as whether this association was identified by a comparable tool, the junction was found in GTEx, and whether the event occurred in a cancer gene according to Cancer Gene Census (CGC) (**Figure 1C**)[24,27]. Finally, we created IGV sessions for each event identified by RegTools that contained a bed file with the junction, a VCF file with the variant, and an alignment (BAM) file for each sample that contained the variant[28]. These IGV sessions were used to manually review candidate events to assess whether the association between the variant and junction makes sense in a biological context.

RegTools is designed for broad applicability and computational efficiency. By relying on well-established standards for sequence alignments, annotation files, and variant calls and by remaining agnostic to downstream statistical methods and comparisons, our tool can be applied to a broad set of scientific queries and datasets. Moreover, performance tests show that *cis-splice-effects identify* can process a typical candidate variant list of 1,500,000 variants and a corresponding RNA-seq BAM file of 82,807,868 reads in just ~8 minutes (**Supplementary Figure 1**).

Pan-cancer analysis of 35 tumor types identifies somatic variants that alter canonical splicing

RegTools was applied to 9,173 samples over 35 cancer types. 32 of these cohorts came from TCGA while the remaining three were obtained from other projects being conducted at MGI. Cohort sizes ranged from 21 to 1,022 samples. In total, 6,370,631 variants (**Figure 2A**) and 2,387,989,201 junction observations (**Figure 2B**) were analyzed by RegTools. By comparing the number of initial variants per cohort to the number of statistically significant variants, we

were able to show that RegTools produces a prioritized list of potential splice relevant variants **(Supplementary Figure 2)**. Additionally, when analyzing the junctions within each sample, we found that junctions present in the reference transcriptome are frequently seen within GTEx data while junctions observed from a sample's own transcriptome data that were not present in the reference are rarely seen within GTEx **(Supplementary Figure 3)**. 235,778 significant variant junction pairings were found for junctions that use a known donor and novel acceptor (D), novel donor and known acceptor (A), or novel combination of a known donor and a known acceptor (NDA), with novel here meaning that the junction was not found in the reference transcriptome (**Methods**, **Figure 2C, Supplemental Files 1 and 2**). While our analysis primarily focuses on variants in relation to novel splice events because of the potential importance of these events within tumor processes, we also wanted to assess how often a variant was significantly associated with a known junction. 5,157 variant junction pairings were found for junctions known to the reference (DA junctions) (**Supplemental Files 3 and 4**). This finding indicates that while splice variants usually result in a novel junction occurring, they sometimes alter the expression of known junctions. Generally, significant events were evenly split among each of the novel junction types considered (D, A, and NDA). The number of significant events increased as the splice variant window size increased, with both the E and I results being comparable in number. Notably, hepatocellular carcinoma (HCC) was the only cohort that had whole genome sequencing (WGS) data available and, as expected, it exhibited a marked increase in the number of significant events for its results within the "I" splice variant window. This observation highlights the low sequence coverage of intronic regions that occurs with WES which subsequently leads to underpowered discovery of potential splice altering variants within introns.

Variants were analyzed across tumor types for how often they result in either a single or multiple novel junctions (**Figure 3A**). While a single variant resulting in a single novel junction is most commonly observed (72.27-83.78%), a single variant also commonly results in multiple junctions being created, either of the same type (6.56-10.94%) or of different types (9.66-16.79%) (**Figure 3B**). Variants that are associated with multiple novel junctions of different types were further investigated to identify how often a particular junction type occurred with another (**Figure 3C**). Most commonly, we observed an alternate donor or acceptor site being used in conjunction with an exon skipping event. These events were particularly common within the default window (2 intronic bases or 3 exonic bases from the exon edge), as a SNV or indel within these positions has a high probability of disrupting the natural splice site, thus causing the splicing machinery to use a cryptic splice site nearby or skip the splice site entirely. The next most common event was an alternate donor site and an alternate acceptor site both being used as the result of a single variant. The combination of a novel acceptor site and novel donor site being used in conjunction with an exon-skipping event occurred the least and occurrence of this type of event remains fairly low, even as the search space increases within the larger splice variant windows. This finding indicates the low likelihood of a single variant resulting in simultaneous disruption of a splice acceptor and donor as well as complete skipping of an exon. Overall, this analysis highlights that there is evidence that a single variant can lead to multiple novel junctions being expressed. Tools that only allow for a single junction to be predicted or

associated with a variant therefore may not be completely describing the effect of the variant in question in up to ~27% of cases.

RegTools identifies splice altering variants missed by other splice variant predictors and annotators

To evaluate the performance of RegTools, we compared our results to those of SAVNet, MiSplice, Veridical, VEP, and SpliceAI[13,20,21,23,25]. These tools vary in their inputs and methodology for identifying splice altering variants (**Figure 4A**). Both VEP and SpliceAI only consider information about the variant and its genomic sequence context and do not consider information from a sample's transcriptome. A variant is considered to be splice relevant according to VEP if it occurs within 1-3 bases on the exonic side or 1-8 bases on the intronic side of a splice site. SpliceAI does not have restrictions on where the variant can occur in relation to the splice site but by default, it predicts one new donor and acceptor site within 50 bp of the variant, based on reference transcript sequences from GENCODE. Like RegTools, SAVNet, MiSplice, and Veridical integrate genomic and transcriptomic data in order to identify splice altering variants. MiSplice only considers junctions that occur within 20 bp of the variant. Additionally, SAVNet, MiSplice, and Veridical filter out any transcripts found within the reference transcriptome. SAVNet, MiSplice, and Veridical employ different statistical methods for the identification of splice altering variants. In contrast to RegTools, none of the mentioned tools allow the user to set a custom window in which they wish to focus splice altering variant discovery (e.g. around the splice site, all exonic variants, etc.). These tools have different levels of code availability. MiSplice is available via GitHub as a collection of Perl scripts that are built to run via Load Sharing Facility (LSF) job scheduling. To run MiSplice without an LSF cluster, the authors mention code changes are required. Veridical is available via a subscription through CytoGnomix's MutationForecaster. Similar to RegTools, SAVNet is available via GitHub or through a Docker image. However, SAVNet relies on splicing junction files generated by STAR[29] whereas RegTools can use RNA-Seq alignment files from HISAT2[30], TopHat2[31], or STAR, thus allowing it to be integrated into bioinformatic workflows more easily.

In their recent publications, SAVNet[23], MiSplice[20], and Veridical[21,22] also analyzed data from TCGA, with only minor differences in the number of samples included for each study. VEP and SpliceAI results were obtained by running each tool on all starting variants for the 35 cohorts included in this study. In order to efficiently compare this data, an UpSet plot (**Figure 4B**) was created[32]. Only 343 variants are identified as splice altering by all six tools. Comparatively, MiSplice and SAVNet find few splice altering variants, potentially indicating that these tools are overlooking the complete set of variants that have an effect on splicing. In contrast, Veridical identifies by far the most splice altering variants across all tools, with 94.54 percent of its calls being found by it alone. SpliceAI and VEP called a large number of variants, either alone or in agreement, that none of the tools that integrate transcriptomic data from samples identify. This highlights a limitation of using tools that only focus on genomic data, particularly in a disease context where transcripts are unlikely to have been annotated before. RegTools addresses these short-comings by identifying what pieces of information to extract from a sample's genome and transcriptome in a very basic, unbiased way that allows for generalization. Other

tools either only analyze genomic data, focus on junctions where either the canonical donor or acceptor site is affected (missing junctions that result from complete exon skipping), or consider only those variants within a very narrow distance from known splice sites. RegTools can include any kind of junction type, including exon-exon junctions that have ends that are not known donor/acceptor sites according to the GTF file (N junction according to RegTools), any distance size to make variant-junction associations, and any window size in which to consider variants. Due to these advantages, RegTools identified events missed by one or multiple of the tools to which we compared **(Figure 4B; Supplementary Figures 4 and 5)**.

Pan-cancer analysis reveals novel splicing patterns within known cancer genes and potential cancer drivers

While efforts have been made to associate variants with specific cancer types, there has been little focus on identifying such associations in splice-altering variants, even those in known cancer genes. *TP53* is a rare example whose splice-altering variants are well characterized in numerous cancer types[33]. As such, we further analyzed significant events to identify genes that had recurrent splice altering variants. Within each cohort, we looked for recurrent genes using two separate metrics: a binomial test p-value and the fraction of samples (see Methods). For ranking and selecting the most recurrent genes, each metric was computed by pooling across all cohorts. For assessing cancer-type specificity, each metric was then also computed using only results from a given cancer cohort. Since the mechanisms underlying the creation of novel junctions versus the disruption of existing splicing patterns may be different, analysis was performed separately for D/A/NDA junctions **(Figure 5, Supplementary Figure 6, Supplementary File 5)** and DA junctions **(Supplementary Figure 7, Supplementary File 6)**, which allowed multiple test correction in accordance with the noise of the respective data. We identified 6,954 genes in which there was least one variant predicted to influence the splicing of a D/A/NDA junction. The 99th percentile of these genes, when ranked by either metric, are significantly enriched for known cancer genes, as annotated by the CGC (p=1.26E-19, ranked by binomial p-values, p=2.97E-24, ranked by fraction of samples; hypergeometric test). We also identified 3,643 genes in which there was least one variant predicted to influence the splicing of a DA (known) junction. The 99th percentile of these genes, when ranked by either metric, are also significantly enriched for known cancer genes, as annotated by the Cancer Gene Census (p=1.00E-04, ranked by binomial p-values, p=3.56E-07, ranked by fraction of samples; hypergeometric test). We also performed the same analyses using either the TCGA or MGI cohorts alone. The TCGA-only analyses gave very similar results to the combined analyses, with the 99th percentile of genes found in the D/A/NDA and DA analyses again being enriched for cancer genes **(Supplementary Figures 8 and 9; Supplemental Files 5 and 6)**. Due to small cohort sizes, in the MGI-only analyses, we identified only 329 and 208 genes in the D/A/NDA and DA analyses, respectively. The 99th percentile of genes from these analyses, respectively, were not significantly enriched for cancer genes **(Supplementary Figures 10 and 11; Supplemental Files 5 and 6)**.

When analyzing D, A, and NDA junctions, we saw an enrichment for known tumor suppressor genes among the most splice disrupted genes, including several examples where splice

disruption is a known mechanism such as *TP53*, *PTEN*, *CDKN2A*, and *RB1*. Specifically, in the case of *TP53*, we identified 428 variants that were significantly associated with at least one novel splicing event. One such example is the intronic SNV (GRCh38, chr17:g.7673609C>A) that was identified in an OSCC sample and was associated with an exon skipping event and an alternate acceptor site usage event, with 23 and 41 reads of support, respectively **(Supplemental Figure 12)**. The cancer types in which we find splice disruption of *TP53* and other known cancer genes is in concordance with associations between genes and cancer types described by CGC and CHASMplus[27,34]. Our analysis's recovery of known drivers, many of which with known susceptibilities to splicing dysregulation in cancer, indicates the ability of our method to identify true splicing effects that are likely cancer-relevant.

Another cancer gene that we found to have a recurrence of splicing altering variants was *B2M*. Specifically, we identified six samples with intronic variants on either side of exon 2 **(Figure 6)**. While mutations have been identified and studied within exon 2, we did not find literature that specifically identified intronic variants near exon 2 as a mechanism for disrupting B2M[35]. These mutations were identified by VEP to be either splice acceptor variant or a splice donor variant and were also identified by Veridical. MiSplice was able to predict one of the novel junctions for each variant but failed to predict additional novel junctions due to the limitation of that tool to only predict one novel acceptor and donor site per variant. Notably, 4 out of the 6 samples that these variants were found in are MSI-H (Microsatellite instability-high) tumors[36]. Mutations in *B2M*, particularly within colorectal MSI-H tumors, have been identified as a method for tumors to become incapable of HLA class I antigen-mediated presentation[37]. Furthermore, in a study of patients treated with immune checkpoint blockade (ICB) therapy, defects to B2M were observed in 29.4% of patients with progressing disease[38]. In the same study, *B2M* mutations were exclusively seen in pretreatment samples from patients who did not respond to ICB or in post-progression samples after initial response to ICB[38]. There are several genes that are responsible for the processing, loading, and presentation of antigens, and have been shown to be mutated in cancers[39]. However, no proteins can be substituted for B2M in HLA class I presentation, thus making the loss of B2M a particularly robust method for ICB resistance[40]. We also observe exonic variants and variants further in intronic regions that disrupt canonical splicing of B2M. These findings indicate that intronic variants that result in alternative splice products within B2M may be a mechanism for immune escape within tumor samples.

We also identify recurrent splice altering variants in genes not known to be cancer genes (according to CGC), such as *RNF145*. RegTools identified a recurrent single base pair deletion that results in an exon skipping event of exon 8 (**Supplementary Figure 13**). This gene is a paralog of *RNF139*, which has been found to be mutated in several cancer types[41]. This variant junction association was found in STAD, UCEC, COAD, and ESCA tumors, all of which are considered to be MSI-H tumors[36]. After analyzing the effect of the exon skipping event on the mRNA sequence, we concluded that the reading frame remains intact, possibly leading to a gain of function event. Additionally, the skipping of exon 8 leads to the removal of a transmembrane domain and a phosphorylation site, S352, which could be important for the regulation of this gene[42]. Based on these findings, *RNF145* may play a role similar to *RNF139* and may be an important driver event in certain tumor samples.

While most of our analysis focused on splice altering variants that resulted in D, A, NDA junctions, we also wanted to investigate variants that shifted the usage of known donor and acceptor sites. Through this analysis, we identified *CDKN2A*, a tumor suppressor gene that is frequently mutated in numerous cancers[43], to have several variants that led to alternate donor usage (**Supplementary Figure 14)**. When these variants are present, an alternate known donor site is used that leads to the formation of the transcript ENST00000579122.1 instead of ENST00000304494.9, the transcript that encodes for $p16^{ink4a}$, a known tumor suppressor. The transcript that results from use of this alternate donor site is missing the last twenty-eight amino acids that form the C-terminal end of $p16^{ink4a}$. Notably, this removes two phosphorylation sites within the p16 protein, S140 and S152, which when phosphorylated promotes the association of $p16^{ink4a}$ with CDK4[44]. This finding highlights the importance of including known transcripts in alternative splicing analyses as variants may alter splice site usage in a way that results in a known but pathogenic transcript product.

Discussion

Splice associated variants are often overlooked in traditional genomic analysis. To address this limitation, we created RegTools, a software suite for the analysis of variants and junctions in a splicing context. By relying on well-established standards for analyzing genomic and transcriptomic data and allowing flexible analysis parameters, we enable users to apply RegTools to a wide set of scientific methodologies and datasets. To ease the use and integration of RegTools into analysis workflows, we provide documentation and example workflows via (regtools.org) and provide a Docker image with all necessary software installed.

In order to demonstrate the utility of our tool, we applied RegTools to 9,173 tumor samples across 35 tumor types to profile the landscape of this category of variants. From this analysis, we report 133,987 variants that cause novel splicing events that were missed by VEP or SpliceAI. Only 1.4 percent of these mutations were previously discovered by similar attempts, while 98.6 percent are novel findings. We demonstrate that there are splice altering variants that occur beyond the splice site consensus sequence, shift transcript usage between known transcripts, and create novel exon-exon junctions that have not been previously described. Specifically, we describe notable findings within *B2M*, *RNF145, and CDKN2A*. These results demonstrate the utility of RegTools in discovering novel splice-altering mutations and confirm the importance of integrating RNA and DNA sequencing data in understanding the consequences of somatic mutations in cancer. To allow further investigation of these identified events, we make all of our annotated result files (**Supplemental Files 1-4**) and recurrence analysis files (**Supplemental Files 5-6**) available.

Understanding the splicing landscape is crucial for unlocking potential therapeutic avenues in precision medicine and elucidating the basic mechanisms of splicing. The exploration of novel tumor-specific junctions will undoubtedly lead to translational applications, from discovering novel tumor drivers, diagnostic and prognostic biomarkers, and drug targets, to identifying a previously untapped source of neoantigens for personalized immunotherapy. While our analysis

focuses on splice altering variants within cancers, we believe RegTools will play an important role in answering this broad range of questions by helping users extract splicing information from transcriptome data and linking it to somatic (or germline) variant calls. The computational efficiency of RegTools and increasing availability and size of such datasets may also allow for improved understanding of splice regulatory motifs that have proven difficult to accurately define such as exonic and intronic splicing enhancers and silencers. Any group with paired DNA and RNA-seq data for the same samples stands to benefit from the functionality of RegTools.

Methods

Software implementation

RegTools is written in C++. CMake is used to build the executable from source code. We have designed the RegTools package to be self-contained in order to minimize external software dependencies. A Unix platform with a C++ compiler and CMake is the minimum prerequisite for installing RegTools. Documentation for RegTools is maintained as text files within the source repository to minimize divergence from the code. We have implemented common file handling tasks in RegTools with the help of open-source code from Samtools/HTSlib[26] and BEDTools[45] in an effort to ensure fast performance, consistent file handling, and interoperability with any aligner that adheres to the BAM specification. Statistical tests are conducted within RegTools using the RMath framework. Travis CI and Coveralls are used to automate and monitor software compilation and unit tests to ensure software functionality. We utilized the Google Test framework to write unit tests.

RegTools consists of a core set of modules for variant annotation, junction extraction, junction annotation, and GTF utilities. Higher level modules such as *cis-splice-effects* make use of the lower level modules to perform more complex analyses. We hope that bioinformaticians familiar with C/C++ can re-use or adapt the RegTools code to implement similar tasks.

Benchmarking

Performance metrics were calculated for all RegTools commands. Each command was run with default parameters on a single blade server (Intel(R) Xeon(R) CPU E5-2660 v2 @ 2.20GHz) with 10 GB of RAM and 10 replicates for each data point **(Supplementary Figure 1)**. Specifically for *cis-splice-effects identify*, we started with random selections of somatic variants, ranging from 10,000-1,500,000, across 8 data subsets. Using the output from *cis-splice-effects identify*, *variants annotate* was run on somatic variants from the 8 subsets (range: 0-17,742) predicted to have a splicing consequence. The function *junctions extract* was performed on the HCC1395 tumor RNA-seq data aligned with HISAT to GRCh37 and randomly downsampled at intervals ranging from 10-100%. Using output from *junctions extract*, *junctions annotate* was performed for 7 data subsets ranging from 1,000-500,000 randomly selected junctions.

Benchmark tests revealed an approximately linear performance for all functions. Variance between real and CPU time is highly dependent on the I/O speed of the write-disk and could account for artificially inflated real time values given multiple jobs writing to the same disk at once. The most computationally expensive function in a typical analysis workflow was *junctions extract*, which on average processed 33,091 reads/second (CPU) and took an average of 43.4 real vs 41.7 CPU minutes to run on a full bam file (82,807,868 reads total). The function *junctions annotate* was the next most computationally intensive function and took an average of 33.0 real/8.55 CPU minutes to run on 500,000 junctions, processing 975 junctions/second (CPU). The other functions were comparatively faster with *cis-splice-effects identify* and *variants annotate* able to process 3,105 and 118 variants per second (CPU), respectively. To process a typical candidate variant list of 1,500,000 variants and a corresponding RNA-seq BAM file of 82,807,868 reads with *cis-splice-effects identify* takes ~ 8.20 real/8.05 CPU minutes **(Supplementary Figure 1)**.

Performance metrics were also calculated for the statistics script and its associated wrapper script that handles dividing the variants into smaller chunks for processing to limit RAM usage. This command, *compare_junctions*, was benchmarked in January 2020 using Amazon Web Services (AWS) on a m5.4xlarge instance, based on the Amazon Linux 2 AMI, with 64 Gb of RAM, 16 vCPUs, and a mounted 1 TB SSD EBS volume with 3000 IOPS. These data were generated from running *compare_junctions* on each of the included cohorts, with the largest being our BRCA cohort (1022 sample) which processed 3.64 events per second (CPU).

Using RegTools to identify cis-acting, splice altering variants

RegTools contains three sub-modules: “variants”, “junctions”, and “cis-splice-effects”. For complete instructions on usage, including a detailed workflow for how to analyze cohorts using RegTools, please visit regtools.org.

Variants annotate

This command takes a list of variants in VCF format. The file should be gzipped and indexed with Tabix[46]. The user must also supply a GTF file that specifies the reference transcriptome used to annotate the variants.

The INFO column of each line in the VCF is populated with comma-separated lists of the variant-overlapping genes, variant-overlapping transcripts, the distance between the variant and the associated exon edge for each transcript (i.e. each start or end of an exon whose splice variant window included the variant) defined as *min*(distance_from_start_of_exon, distance_from_end_of_exon), and the variant type for each transcript.

Internally, this function relies on HTSlib to parse the VCF file and search for features in the GTF file which overlap the variant. The splice variant window size (i.e. the maximum distance from the edge of an exon used to consider a variant as splicing-relevant) can be set by the options “-e <number of bases>” and “-i <number of bases>” for exonic and intronic variants, respectively. The variant type for each variant thus depends on the options used to set the splice variant window size. Variants captured by the window set by “-e” or “-i” are annotated as

"splicing_exonic" and "splicing_intronic", respectively. Alternatively, to analyze all exonic or intronic variants, the "-E" and "-I" options can be used. Otherwise, the "-E" and "-I" options themselves do not change the variant type annotation, and variants found in these windows are labeled simply as "exonic" or "intronic". By default, single exon transcripts are ignored, but they can be included with the "-S" option. By default, output is written to STDOUT in VCF format. To write to a file, use the option "-o <PATH/TO/FILE>".

Junctions extract

This command takes an alignment file containing aligned RNA-seq reads and infers junctions (i.e. exon-exon boundaries) based on skipped regions in alignments as determined by the CIGAR string operator codes. These junctions are written to STDOUT in BED12 format. Alternatively, the output can be redirected to a file with the "-o <PATH/TO/FILE>". RegTools ascertains strand information based on the XS tags set by the aligner, but can also determine the inferred strand of transcription based on the BAM flags if a stranded library strategy was employed. In the latter case, the strand specificity of the library can be provided using "-s <INT>" where 0 = unstranded, 1 = first-strand/RF, 2 = second-strand/FR. We suggest that users align their RNA-seq data with HISAT2[30], TopHat2[31], or STAR[29], as these are the aligners we have tested to date. If RNA-seq data is unstranded and aligned with STAR, users must run STAR with the --outSAMattributes option to include XS tags in the BAM output.

Users can set thresholds for minimum anchor length and minimum/maximum intron length. The minimum anchor length determines how many contiguous, matched base pairs on either side of the junction are required to include it in the final output. The required overlap can be observed amongst separated reads, whose union determines the thickStart and thickEnd of the BED feature. By default, a junction must have 8 bp anchors on each side to be counted but this can be set using the option "-a <minimum anchor length>". The intron length is simply the end coordinate of the junction minus the start coordinate. By default, the junction must be between 70 bp and 500,000 bp, but the minimum and maximum can be set using "-i <minimum intron length>" and "-I <maximum intron length>", respectively.

For efficiency, this tool can be used to process only alignments in a particular region as opposed to analyzing the entire BAM file. The option "-r <chr>:<start>-<stop>" can be used to set a single contiguous region of interest. Multiple jobs can be run in parallel to analyze separate non-contiguous regions.

Junctions annotate

This command takes a list of junctions in BED12 format as input and annotates them with respect to a reference transcriptome in GTF format. The observed splice-sites used are recorded based on a reference genome sequence in FASTA format. The output is written to STDOUT in TSV format, with separate columns for the number of splicing acceptors skipped, number of splicing donors skipped, number of exons skipped, the junction type, whether the donor site is known, whether the acceptor site is known, whether this junction is known, the overlapping transcripts, and the overlapping genes, in addition to the chromosome, start, stop, junction name, junction score, and strand taken from the input BED12 file. This output can be

redirected to a file with “-o /PATH/TO/FILE”. By default, single exon transcripts are ignored in the GTF but can be included with the option “-S”.

Cis-splice-effects identify
This command combines the above utilities into a pipeline for identifying variants which may cause aberrant splicing events by altering splicing motifs in *cis*. As such, it relies on essentially the same inputs: a gzipped and Tabix-indexed VCF file containing a list of variants, an alignment file containing aligned RNA-seq reads, a GTF file containing the reference transcriptome of interest, and a FASTA file containing the reference genome sequence of interest.

First, the list of variants is annotated. The splice variant window size is set using the options “-e”, “-i”, “-E”, and “-I”, just as in *variants annotate*. The splice junction region size (i.e. the range around a particular variant in which an overlapping junction is associated with the variant) can be set using “-w <splice junction region size>”. By default, this range is not a particular number of bases but is calculated individually for each variant, depending on the variant type annotation. For “splicing_exonic”, “splicing_intronic”, and “exonic” variants, the region extends from the 3’ end of the exon directly upstream of the variant-associated exon to the 5’ end of the exon directly downstream of it. For “intronic” variants, the region is limited to the intron containing the variant. Single-exons can be kept with the “-S” option. The annotated list of variants in VCF format (analogous to the output of *variants annotate*) can be written to a file with “-v /PATH/TO/FILE”.

The BAM file is then processed in the splice junction regions to produce the list of junctions. A file containing these junctions in BED12 format (analogous to the output of *junctions extract*) can be written using “-j /PATH/TO/FILE”. The minimum anchor length, minimum intron length, and maximum intron length can be set using “-a”, “-i”, and “-I” options, just as in *junctions extract*.

The list of junctions produced by the preceding step is then annotated with the information presented in *junctions annotate*. Additionally, each junction is annotated with a list of associated variants (i.e. variants whose splice junction regions overlapped the junction). The final output is written to STDOUT in TSV format (analogous to the output of *junctions annotate*) or can be redirected to a file with “-o /PATH/TO/FILE”.

Cis-splice-effects associate
This command is similar to *cis-splice-effects identify,* but takes the BED output of *junctions extract* in lieu of an alignment file with RNA alignments. As with *cis-splice-effects identify*, each junction is annotated with a list of associated variants (i.e. variants whose splice junction regions overlapped the junction). The resulting output is then the same as *cis-splice-effects identify*, but limited to the junctions provided as input.

Analysis

Dataset Description

32 cancer cohorts were analyzed from TCGA. These cancer types are Adrenocortical carcinoma (ACC), Bladder Urothelial Carcinoma (BLCA), Brain Lower Grade Glioma (LGG), Breast invasive carcinoma (BRCA), Cervical squamous cell carcinoma and endocervical adenocarcinoma (CESC), Cholangiocarcinoma (CHOL), Colon adenocarcinoma (COAD), Esophageal carcinoma (ESCA), Glioblastoma multiforme (GBM), Head and Neck squamous cell carcinoma (HNSC), Kidney Chromophobe (KICH), Kidney renal clear cell carcinoma (KIRC), Kidney renal papillary cell carcinoma (KIRP), Liver hepatocellular carcinoma (LIHC), Lung adenocarcinoma (LUAD), Lung squamous cell carcinoma (LUSC), Lymphoid Neoplasm Diffuse Large B cell Lymphoma (DLBC), Mesothelioma (MESO), Ovarian serous cystadenocarcinoma (OV), Pancreatic adenocarcinoma (PAAD), Pheochromocytoma and Paraganglioma (PCPG), Prostate adenocarcinoma (PRAD), Rectum adenocarcinoma (READ), Sarcoma (SARC), Skin Cutaneous Melanoma (SKCM), Stomach adenocarcinoma (STAD), Testicular Germ Cell Tumors (TGCT), Thymoma (THYM), Thyroid carcinoma (THCA), Uterine Carcinosarcoma (UCS), Uterine Corpus Endometrial Carcinoma (UCEC), and Uveal Melanoma (UVM). Three cohorts were derived from patients at Washington University in St. Louis. These cohorts are Hepatocellular Carcinoma (HCC), Oral Squamous Cell Carcinoma (OSCC), and Small Cell Lung Cancer (SCLC).

Sample processing

We applied RegTools to 35 tumor cohorts. Genomic and transcriptomic data for 32 cohorts were obtained from The Cancer Genome Atlas (TCGA). Information regarding the alignment and variant calling for these samples is described by the Genomic Data Commons data harmonization effort[47]. Whole exome sequencing (WES) mutation calls for these samples from MuSE[48], MuTect2[49], VarScan2[50], and SomaticSniper[51], were left-aligned, trimmed, and decomposed to ensure the correct representation of the variants across the multiple callers.

Samples for the remaining three cohorts, HCC, SCLC, and OSCC, were sequenced at Washington University in St. Louis. Genomic data were produced by WES for SCLC and OSCC and whole genome sequencing (WGS) for HCC. Normal genomic data of the same sequencing type and tumor RNA-seq data were also available for all subjects. Sequence data were aligned using the Genome Modeling System (GMS)[52] using TopHat2 for RNA and BWA-MEM[53] for DNA. HCC and SCLC were aligned to GRCh37 while OSCC was aligned to GRCh38. Somatic variant calls were made using Samtools v0.1.1[26], SomaticSniper2 v1.0.2[51], Strelka V0.4.6.2[54], and VarScan v2.2.6[50,54] through the GMS. High-quality mutations for all samples were then selected by requiring that a variant be called by two of the four variant callers.

Candidate junction filtering

To generate results for 4 splice variant window sizes, we ran *cis-splice-effects identify* with 4 sets of splice variant window parameters. For our “i2e3” window (RegTools default), to examine intronic variants within 2 bases and exonic variants within 3 bases of the exon edge, we set “-i 2 -e 3”. Similarly, for “i50e5”, to examine intronic variants within 50 bases and exonic variants within 5 bases of the exon edge, we set “-i 50 -e 5”. To view all exonic variants, we simply set “-

E", without "-i" or "-e" options. To view all intronic variants, we simply set "-l", without "-i" or "-e" options. TCGA samples were processed with GRCh38.d1.vd1.fa (downloaded from the GDC reference file page at https://gdc.cancer.gov/about-data/gdc-data-processing/gdc-reference-files) as the reference fasta file and gencode.v29.annotation.gtf (downloaded via the GENCODE FTP) as the reference transcriptome. OSCC was processed with Homo_sapiens.GRCh38.dna_sm.primary_assembly.fa and Homo_sapiens.GRCh38.79.gtf (both downloaded from Ensembl). HCC and SCLC were processed with Homo_sapiens.GRCh37.dna_sm.primary_assembly.fa and Homo_sapiens.GRCh37.87.gtf (both downloaded from Ensembl).

Statistical filtering of candidate events

We refer to a statistical association between a variant and a junction as an "event". For each event identified by RegTools, a normalized score (norm_score) was calculated for the junction of the event by dividing the number of reads supporting that junction by the sum of all reads for all junctions within the splice junction region for the variant of interest. This metric is conceptually similar to a "percent-spliced in" (PSI) index, but measures the presence of entire exon-exon junctions, instead of just the inclusion of individual exons. If there were multiple samples that contained the variant for the event, then the mean of the normalized scores for the samples was computed (mean_norm_score). If only one sample contained the variant, its mean_norm_score was thus equal to its norm_score. This value was then compared to the distribution of samples which did not contain the variant to calculate a p-value as the percentage of the norm_scores from these samples which are at least as high as the mean_norm_score computed for the variant-containing samples. We performed separate analyses for events involving canonical junctions (DA) and those involving novel junctions which used at least one known splice site (D/A/NDA), based on annotations in the corresponding reference GTF. For this study, we filtered out any junctions which did not use at least one known splice site (N) and junctions which did not have at least 5 reads of evidence across variant-containing samples. The Benjamini-Hochberg procedure was then applied to the remaining events. Following correction, an event was considered significant if its adjusted p-value was ≤ 0.05.

Annotation with GTEx junction data and other splice prediction tools

Events identified by RegTools as significant were annotated with information from GTEx, VEP, SpliceAI, MiSplice, and Veridical. GTEx junction information was obtained from the GTEx Portal. Specifically, the exon-exon junction read counts file from the v8 release was used for data aligned to GRCh38 while the same file from the v7 release was used for the data aligned to

GRCh37. Mappings between tumor cohorts and GTEx tissues can be found in **Supplemental File 7**. We annotated all starting variants with VEP in the "per_gene" and "pick" modes. The "per_gene" setting outputs only the most severe consequence per gene while the "pick" setting picks one line or block of consequence data per variant. We considered any variant with at least one splicing-related annotation to be "VEP significant". All variants were also processed with SpliceAI using the default options. A variant was considered to be "SpliceAI significant" if it had at least one score greater than 0.2, the developers' value for high recall of their model. Variants identified by MiSplice[20] were obtained from the paper supplemental tables and were lifted over to GRCh38. Variants identified by SAVNet[23] were obtained from the paper supplemental tables and were lifted over to GRCh38. Variants identified by Veridical[21,22] were obtained via download from the link reference within the manuscript and lifted over to GRCh38.

Visual exploration of statistically significant candidate events

IGV sessions were created for each event identified by RegTools that was statistically significant. Each IGV session file contained a bed file with the junction, a vcf file with the variant, and an alignment file for each sample that contained the variant. Additional information, such as the splice sites predicted by SpliceAI, were also added to these session files to enhance the exploration of these events. Events of interest were manually reviewed in IGV to assess whether the association between the variant and junction made sense in a biological context (e.g. affected a known splice site, altered a genomic sequence to look more like a canonical splice site, or the novel junction disrupted active or regulatory domains of the protein product). An extensive review of literature and visualizations of junction usage in the presence and absence of the variant were also used to identify novel, biologically relevant events.

Identification of genes with recurrent splice altering variants

For each cohort, we calculated a p-value to assess whether the splicing profile from a particular gene was significantly more likely to be altered by somatic variants. Specifically, we performed a 1-tailed binomial test, considering the number of samples in a cohort as the number of attempts. Success was defined by whether the sample had evidence of at least one splice-altering variant in that gene. The null probability of success, p_{null} was calculated as

$$p_{null} = 1 - (1 - Pr(V \wedge A))^s$$

where *s* is the total number of base positions residing in any of the gene's splice variant windows, *V* is the event that a somatic variant occurred at such a base position, and *A* is the event that this variant was deemed to be significantly associated with at least one junction in our analysis. The joint probability that both V and A occurred was estimated by dividing the total of events across all samples in which each junction was detected by *s*. The value of *s* was computed based on the exon and transcript definitions in the reference GTF used for performing RegTools analyses on a given cohort.

We also calculated overall metrics, in order to rank genes. For each set of cohorts (e.g. TCGA-only, MGI-only, combined), an overall p-value was computed for each gene according to the above formula, pooling all of the samples across the included cohorts, and the fraction of samples was simply calculated by dividing the number of samples in which an event occurred within the given gene by the total number of samples, pooled across the included cohorts. The reference GTF used for analyzing the TCGA samples (i.e. gencode.v29.annotation.gtf) was used for all sets of cohorts.

Code availability

RegTools is open source (MIT license) and available at https://github.com/griffithlab/regtools/. All scripts used in the analyses presented here are also provided. For ease of use, a Docker container has been created with RegTools, R, and Python 3 installed (https://hub.docker.com/r/griffithlab/regtools/). This Docker container allows a user to run the workflow we outline at https://regtools.readthedocs.io/en/latest/workflow/. Docker is an open-source software platform that enables applications to be readily installed and run on any system. The availability of RegTools with all its dependencies as a Docker container also facilitates the integration of the RegTools software into workflow pipelines that support Docker images.

Data availability

Sequence data for each cohort analyzed in this study are available through dbGaP at the following accession IDs: phs000178 for TCGA cohorts, phs001106 for HCC, phs001049 for SCLC, and phs001623 for OSCC. Statistically significant events for D, A, and NDA junctions across the four variant splicing windows used are available via **Supplemental Files 1 and 2**. Statistically significant events for DA junctions are available as **Supplemental Files 3 and 4**. Complete results of gene recurrence analysis are available as **Supplemental Files 5 and 6**.

Acknowledgments

We thank the patients and their families for donation of their samples and participation in clinical trials. We would like to thank Donald Conrad for his initial idea to compare to variant effect predictor tools. Kelsy Cotto was supported by Siteman Cancer Center under fund number #3477-92400 and T32CA113275. Avinash Ramu was supported by the ‘Burroughs Wellcome Fund Institutional Program Unifying Population and Laboratory Based Sciences Award’ at Washington University. Malachi Griffith was supported by the National Human Genome Research Institute (NHGRI) of the National Institutes of Health (NIH) under Award Number R00HG007940. Malachi Griffith and Obi Griffith were supported by the NIH National Cancer Institute (NCI) under Award Numbers U01CA209936, U01CA231844, U01CA248235 U24CA237719. Malachi Griffith and Megan Richters were supported by the V Foundation for Cancer Research under Award Number V2018-007. The results published here are in whole or part based upon data generated by the TCGA Research Network: https://www.cancer.gov/tcga.

Contributions

K.C.C. and Y.-Y.F. were involved in all aspects of this study, including designing methodology, developing and testing the tool software, analyzing and interpreting data, and writing the

manuscript, with input from A.R., Z.L.S., M.R., S.F., J.K., O.L.G., and M.G. A.R. designed the tool and led software development efforts. Y.L., W.C.C., R.U., and R.G. provided unpublished tumor datasets and provided critical feedback on the manuscript. O.L.G. and M.G. supervised the study. All authors read and approved the final manuscript.

Conflicts of Interest

W. Chapman serves on the advisory board for Novartis Pharmaceutical and reports intellectual property with Pathfinder Therapeutics. R. Uppaluri reports grants and personal fees from Merck Inc. R. Govindan served as consultant for Horizon Pharmaceuticals and GenePlus.

References

1. Chabot, B. & Shkreta, L. Defective control of pre-messenger RNA splicing in human disease. *J. Cell Biol.* **212**, 13–27 (2016).
2. Vogelstein, B. *et al.* Cancer genome landscapes. *Science* **339**, 1546–1558 (2013).
3. Soemedi, R. *et al.* Pathogenic variants that alter protein code often disrupt splicing. *Nat. Genet.* **49**, 848–855 (2017).
4. Supek, F., Miñana, B., Valcárcel, J., Gabaldón, T. & Lehner, B. Synonymous mutations frequently act as driver mutations in human cancers. *Cell* **156**, 1324–1335 (2014).
5. Jung, H. *et al.* Intron retention is a widespread mechanism of tumor-suppressor inactivation. *Nat. Genet.* **47**, 1242–1248 (2015).
6. Venables, J. P. Aberrant and alternative splicing in cancer. *Cancer Res.* **64**, 7647–7654 (2004).
7. Climente-González, H., Porta-Pardo, E., Godzik, A. & Eyras, E. The Functional Impact of Alternative Splicing in Cancer. *Cell Rep.* **20**, 2215–2226 (2017).
8. Chen, J. & Weiss, W. A. Alternative splicing in cancer: implications for biology and therapy. *Oncogene* **34**, 1–14 (2015).
9. Xiong, H. Y. *et al.* RNA splicing. The human splicing code reveals new insights into the genetic determinants of disease. *Science* **347**, 1254806 (2015).
10. Yeo, G. & Burge, C. B. Maximum entropy modeling of short sequence motifs with applications to RNA splicing signals. *J. Comput. Biol.* **11**, 377–394 (2004).

11. Fairbrother, W. G., Yeh, R.-F., Sharp, P. A. & Burge, C. B. Predictive identification of exonic splicing enhancers in human genes. *Science* **297**, 1007–1013 (2002).
12. Wang, Z. *et al.* Systematic identification and analysis of exonic splicing silencers. *Cell* **119**, 831–845 (2004).
13. Jaganathan, K. *et al.* Predicting Splicing from Primary Sequence with Deep Learning. *Cell* **176**, 535–548.e24 (2019).
14. Kahles, A., Ong, C. S., Zhong, Y. & Rätsch, G. SplAdder: identification, quantification and testing of alternative splicing events from RNA-Seq data. *Bioinformatics* **32**, 1840–1847 (2016).
15. Trincado, J. L. *et al.* SUPPA2: fast, accurate, and uncertainty-aware differential splicing analysis across multiple conditions. *Genome Biol.* **19**, 40 (2018).
16. Kahles, A. *et al.* Comprehensive Analysis of Alternative Splicing Across Tumors from 8,705 Patients. *Cancer Cell* **34**, 211–224.e6 (2018).
17. Li, Y. I. *et al.* Annotation-free quantification of RNA splicing using LeafCutter. *Nat. Genet.* **50**, 151–158 (2018).
18. Monlong, J., Calvo, M., Ferreira, P. G. & Guigó, R. Identification of genetic variants associated with alternative splicing using sQTLseekeR. *Nat. Commun.* **5**, 4698 (2014).
19. Li, Y. I. *et al.* RNA splicing is a primary link between genetic variation and disease. *Science* **352**, 600–604 (2016).
20. Jayasinghe, R. G. *et al.* Systematic Analysis of Splice-Site-Creating Mutations in Cancer. *Cell Rep.* **23**, 270–281.e3 (2018).
21. Viner, C., Dorman, S. N., Shirley, B. C. & Rogan, P. K. Validation of predicted mRNA splicing mutations using high-throughput transcriptome data. *F1000Res.* **3**, (2014).
22. Shirley, B. C., Mucaki, E. J. & Rogan, P. K. Pan-cancer repository of validated natural and cryptic mRNA splicing mutations. *F1000Res.* **7**, 1908 (2018).
23. Shiraishi, Y. *et al.* A comprehensive characterization of cis-acting splicing-associated

variants in human cancer. *Genome Res.* **28**, 1111–1125 (2018).

24. GTEx Consortium. The Genotype-Tissue Expression (GTEx) project. *Nat. Genet.* **45**, 580–585 (2013).
25. McLaren, W. *et al.* The Ensembl Variant Effect Predictor. *Genome Biol.* **17**, 122 (2016).
26. Li, H. *et al.* The Sequence Alignment/Map format and SAMtools. *Bioinformatics* **25**, 2078–2079 (2009).
27. Sondka, Z. *et al.* The COSMIC Cancer Gene Census: describing genetic dysfunction across all human cancers. *Nat. Rev. Cancer* **18**, 696–705 (2018).
28. Robinson, J. T. *et al.* Integrative genomics viewer. *Nat. Biotechnol.* **29**, 24–26 (2011).
29. Dobin, A. *et al.* STAR: ultrafast universal RNA-seq aligner. *Bioinformatics* **29**, 15–21 (2013).
30. Kim, D., Langmead, B. & Salzberg, S. L. HISAT: a fast spliced aligner with low memory requirements. *Nat. Methods* **12**, 357–360 (2015).
31. Kim, D. *et al.* TopHat2: accurate alignment of transcriptomes in the presence of insertions, deletions and gene fusions. *Genome Biol.* **14**, R36 (2013).
32. Conway, J. R., Lex, A. & Gehlenborg, N. UpSetR: an R package for the visualization of intersecting sets and their properties. *Bioinformatics* **33**, 2938–2940 (2017).
33. Surget, S., Khoury, M. P. & Bourdon, J.-C. Uncovering the role of p53 splice variants in human malignancy: a clinical perspective. *Onco. Targets. Ther.* **7**, 57–68 (2013).
34. Tokheim, C. & Karchin, R. CHASMplus Reveals the Scope of Somatic Missense Mutations Driving Human Cancers. *Cell Syst* **9**, 9–23.e8 (2019).
35. Bicknell, D. C., Kaklamanis, L., Hampson, R., Bodmer, W. F. & Karran, P. Selection for β2-microglobulin mutation in mismatch repair-defective colorectal carcinomas. *Curr. Biol.* **6**, 1695–1697 (1996).
36. Bonneville, R. *et al.* Landscape of Microsatellite Instability Across 39 Cancer Types. *JCO Precis Oncol* **2017**, (2017).
37. Kloor, M. *et al.* Immunoselective pressure and human leukocyte antigen class I antigen

machinery defects in microsatellite unstable colorectal cancers. *Cancer Res.* **65**, 6418–6424 (2005).

38. Sade-Feldman, M. *et al.* Resistance to checkpoint blockade therapy through inactivation of antigen presentation. *Nat. Commun.* **8**, 1136 (2017).
39. Seliger, B., Maeurer, M. J. & Ferrone, S. Antigen-processing machinery breakdown and tumor growth. *Immunol. Today* **21**, 455–464 (2000).
40. Güssow, D. *et al.* The human beta 2-microglobulin gene. Primary structure and definition of the transcriptional unit. *J. Immunol.* **139**, 3132–3138 (1987).
41. Wang, L., Yin, W. & Shi, C. E3 ubiquitin ligase, RNF139, inhibits the progression of tongue cancer. *BMC Cancer* **17**, 452 (2017).
42. Hornbeck, P. V. *et al.* PhosphoSitePlus, 2014: mutations, PTMs and recalibrations. *Nucleic Acids Res.* **43**, D512–20 (2015).
43. Zhao, R., Choi, B. Y., Lee, M.-H., Bode, A. M. & Dong, Z. Implications of Genetic and Epigenetic Alterations of CDKN2A (p16(INK4a)) in Cancer. *EBioMedicine* **8**, 30–39 (2016).
44. Gump, J., Stokoe, D. & McCormick, F. Phosphorylation of p16 INK4A Correlates with Cdk4 Association. *J. Biol. Chem.* **278**, 6619–6622 (2003).
45. Quinlan, A. R. BEDTools: The Swiss-Army Tool for Genome Feature Analysis. *Curr. Protoc. Bioinformatics* **47**, 11.12.1–34 (2014).
46. Li, H. Tabix: fast retrieval of sequence features from generic TAB-delimited files. *Bioinformatics* **27**, 718–719 (2011).
47. GDC Data Processing. https://gdc.cancer.gov/about-data/gdc-data-processing.
48. Fan, Y. *et al.* Accounting for tumor heterogeneity using a sample-specific error model improves sensitivity and specificity in mutation calling for sequencing data. *bioRxiv* 055467 (2016) doi:10.1101/055467.
49. Cibulskis, K. *et al.* Sensitive detection of somatic point mutations in impure and heterogeneous cancer samples. *Nat. Biotechnol.* **31**, 213–219 (2013).

50. Koboldt, D. C. *et al.* VarScan 2: somatic mutation and copy number alteration discovery in cancer by exome sequencing. *Genome Res.* **22**, 568–576 (2012).
51. Larson, D. E. *et al.* SomaticSniper: identification of somatic point mutations in whole genome sequencing data. *Bioinformatics* **28**, 311–317 (2012).
52. Griffith, M. *et al.* Genome Modeling System: A Knowledge Management Platform for Genomics. *PLoS Comput. Biol.* **11**, e1004274 (2015).
53. Li, H. & Durbin, R. Fast and accurate short read alignment with Burrows-Wheeler transform. *Bioinformatics* **25**, 1754–1760 (2009).
54. Saunders, C. T. *et al.* Strelka: accurate somatic small-variant calling from sequenced tumor–normal sample pairs. *Bioinformatics* **28**, 1811–1817 (2012).

Main Figures

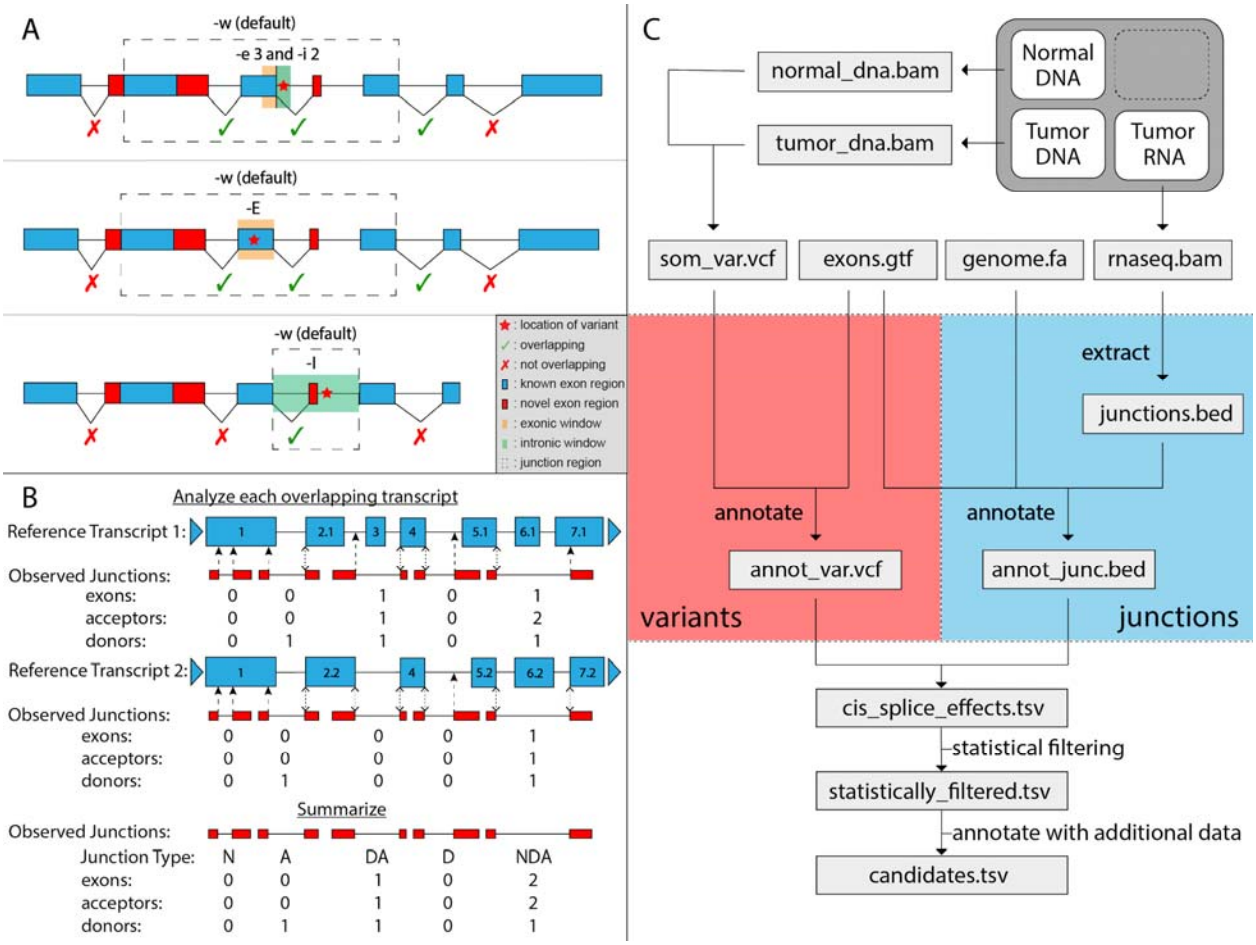


Figure 1: Flexible, streamlined discovery of cis-acting splice variants with RegTools modules and *cis-splice-effects identify workflow*.

A) By default, *variants annotate* marks variants within 3bp on the exonic side and 2bp on the intronic side of an exon edge as potentially splicing-relevant. This "splice variant window" can be modified individually for the exonic side and intronic side using the "-e" and "-i" options, respectively. With *cis-splice-effects identify*, for each variant in the splice variant window, a "splice junction region" is determined by finding the largest span of sequence space between exons which flank the exon associated with the splicing-relevant variant. The splice junction region can also be set manually to contain the entire sequence space *n* bases upstream and downstream of the variant using the "-w" option. Junctions overlapping the splice junction region are associated with the variant. Using the -E option considers all exonic variants as potentially splicing-relevant, but is otherwise the same. The -I option considers all intronic variants and also limits the splice junction region to the intronic region in which the variant is found, excluding the flanking exons. **B)** *Cis-splice-effects identify* and the underlying *junctions annotate* command annotate splicing events based on whether the donor and acceptor site combination is found in the reference transcriptome GTF. In this example, there are two known transcripts (shown in blue) which overlap a set of junctions from RNAseq data (depicted as junction supporting reads

in red). Comparing the observed junctions to the reference junctions in the first transcript (top panel), RegTools checks to see if the observed donor and acceptor splice sites are found in any of the reference exons and also counts the number of exons, acceptors, and donors skipped by a particular junction. Double arrows represent matches between observed and reference acceptor/donor sites while single arrows show novel splice sites. These steps are repeated for the rest of the relevant transcripts, keeping track of whether there are known acceptor-donor combinations. Junctions with a known donor but novel acceptor or vice-versa are annotated as "D" or "A", respectively. If both sites are known but do not appear in combination in any transcripts, the junction is annotated as "NDA", whereas if both sites are unknown, the junction is annotated as "N". If the junction is known to the reference GTF, it is marked as "DA". **C)** The *cis-splice-effects identify* command relies on the *variants annotate*, *junctions extract*, and *junctions annotate* submodules. This pipeline takes variant calls and RNA-seq alignments along with genome and transcriptome references and outputs information about novel junctions and associated potential cis splice-altering sequence variants. RegTools is agnostic to downstream research goals and its output can be filtered through user-specific methods and thus can be applied to a broad set of scientific questions.

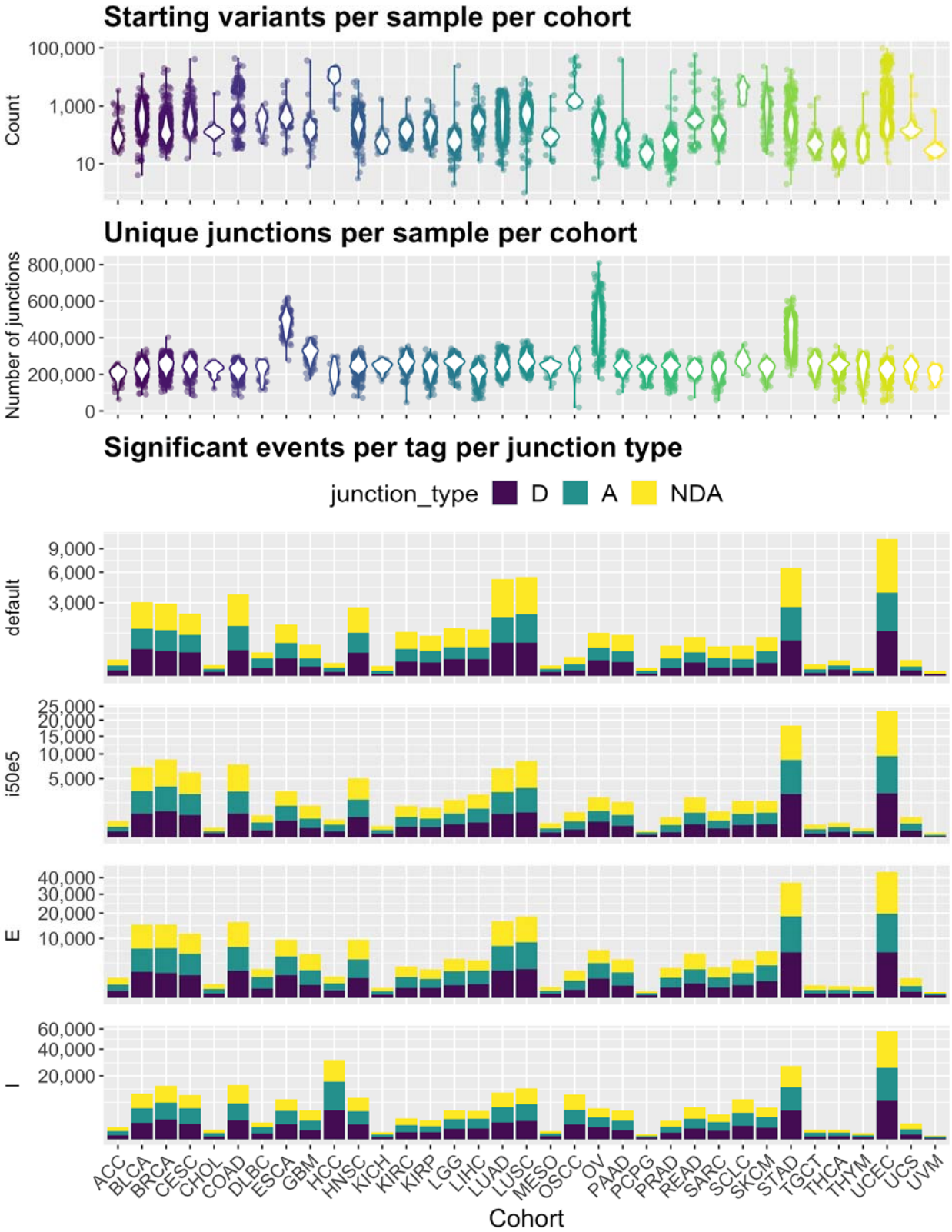


Figure 2. Overview of input data considered and significant events identified by RegTools for each tumor type.

A) Summary of initial variants considered for analysis by RegTools per sample per tumor cohort. Each sample's variant count is plotted and violin plots are overlaid for each cohort. **B)** Summary unique exon-exon junction observations for each sample. Each sample's unique junction count is plotted and violin plots are overlaid for each cohort. **C)** Summary of significant junction types for each cohort across each of the variant window sizes that were used in this analysis.

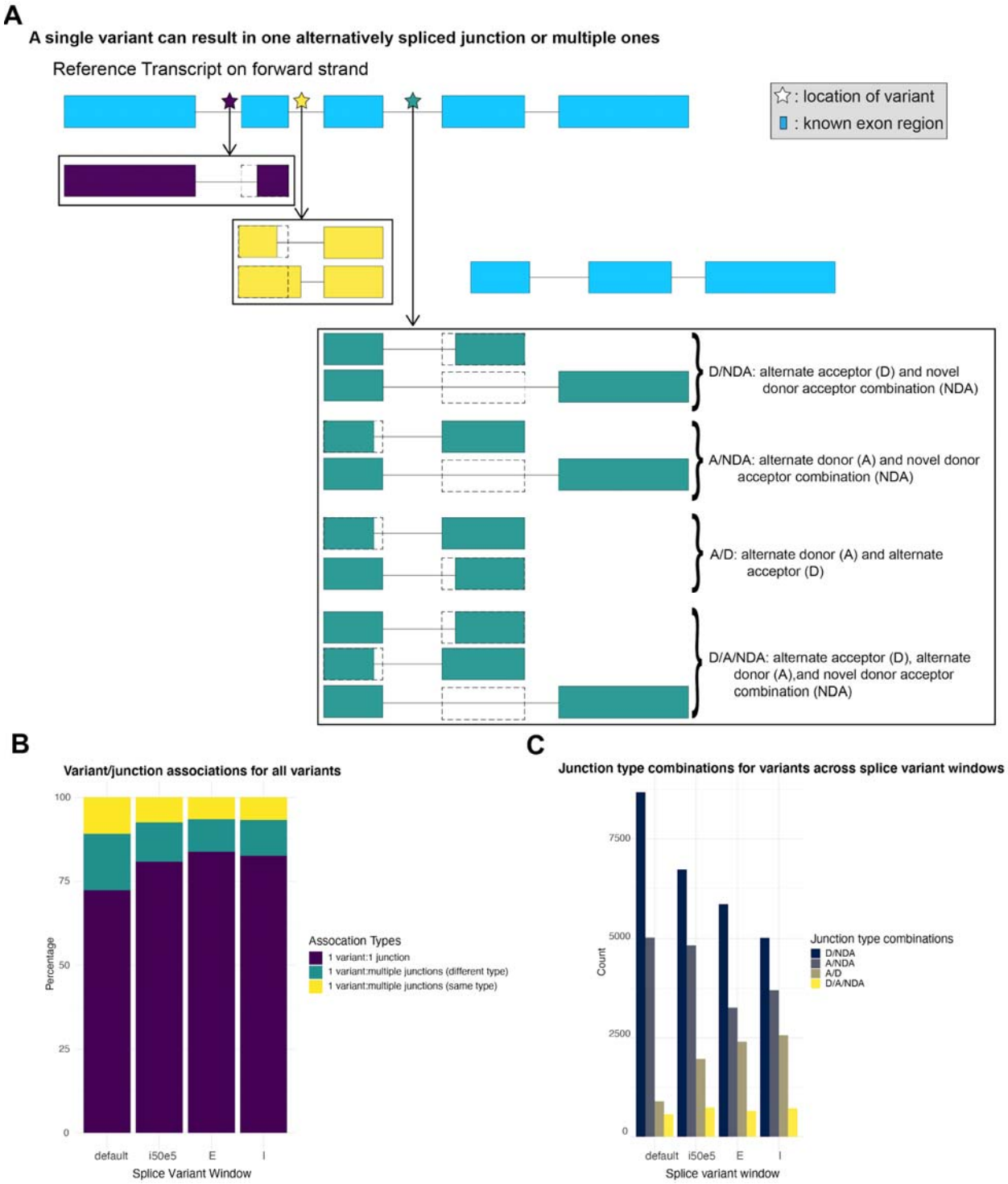


Figure 3. Splice regulatory variants often lead to the expression of multiple alternative junctions.

A) A single variant can result in either one or more than one alternatively spliced junctions. Depicted is a variant resulting in a single novel transcript product (purple), a variant resulting in two novel transcript products that both use alternate donor sites (yellow), and a variant resulting in multiple junctions of different types (teal). **B)** Stacked bar graph visualizing how often a

variant leads to each of the categories mentioned above across the four RegTools variant windows used. This analysis is for all variants that RegTools identified as significant. **C)** Bar chart showing how often each of the described junction combinations occurs when a single variant results in multiple junction types across each of the RegTools splice variant windows used.

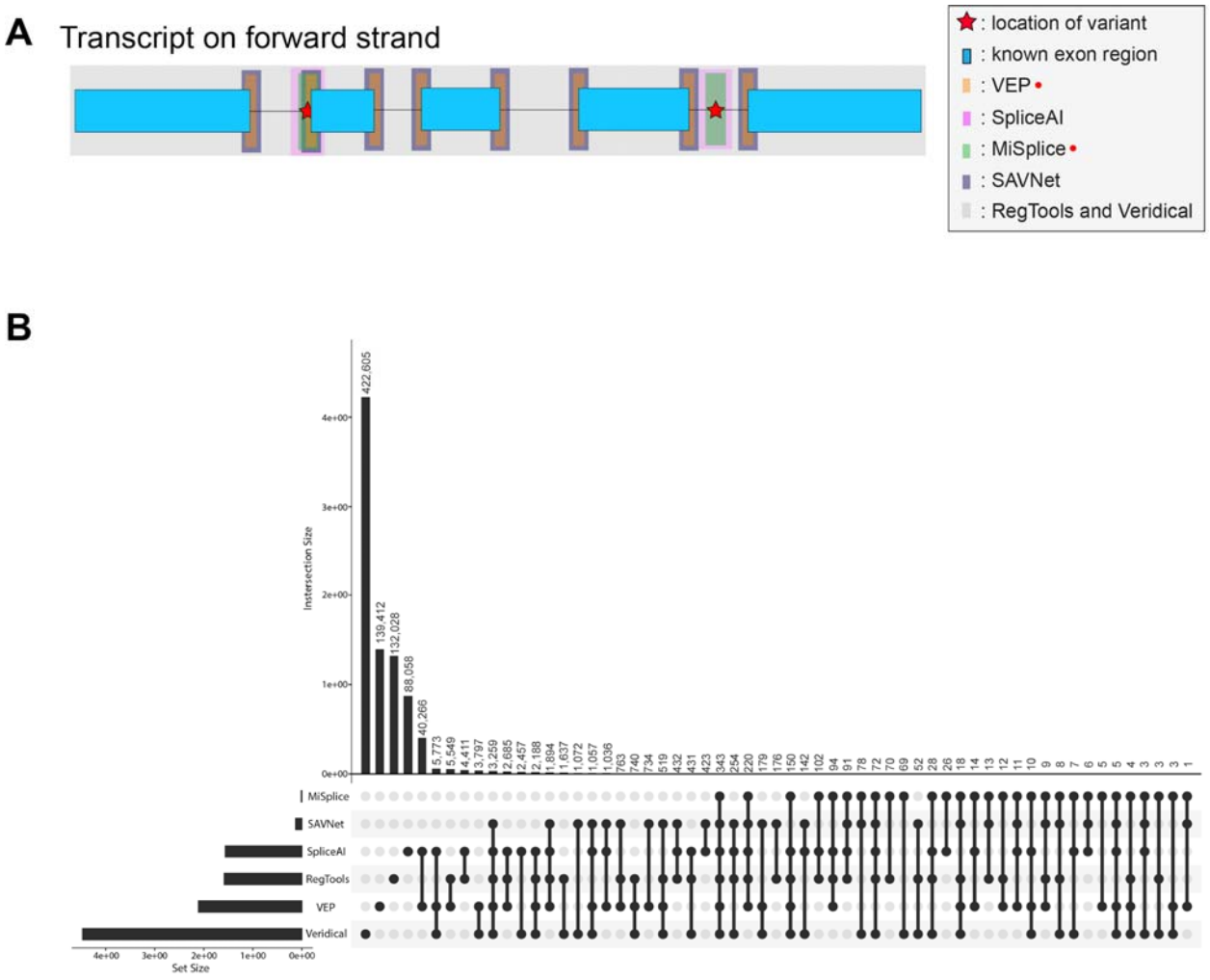


Figure 4. Comparison of RegTools with other tools that identify potential splice altering variants.

A) Conceptual diagram of contrasting approaches used to identify splice regulatory tools/methods. A red dot indicates that the source only considers genomic data for making its calls, as opposed to a combination of genomic and transcriptomic data. **B)** UpSet plot comparing splice altering variants identified by RegTools to those identified by other splice variant predictors and annotators. Each tool and their total number of variant predictions are shown on the left side bar graph. The numbers of variants specific to each tool or shared between different combinations of tools are indicated by the bar graph along the top and the individual or connected dots.

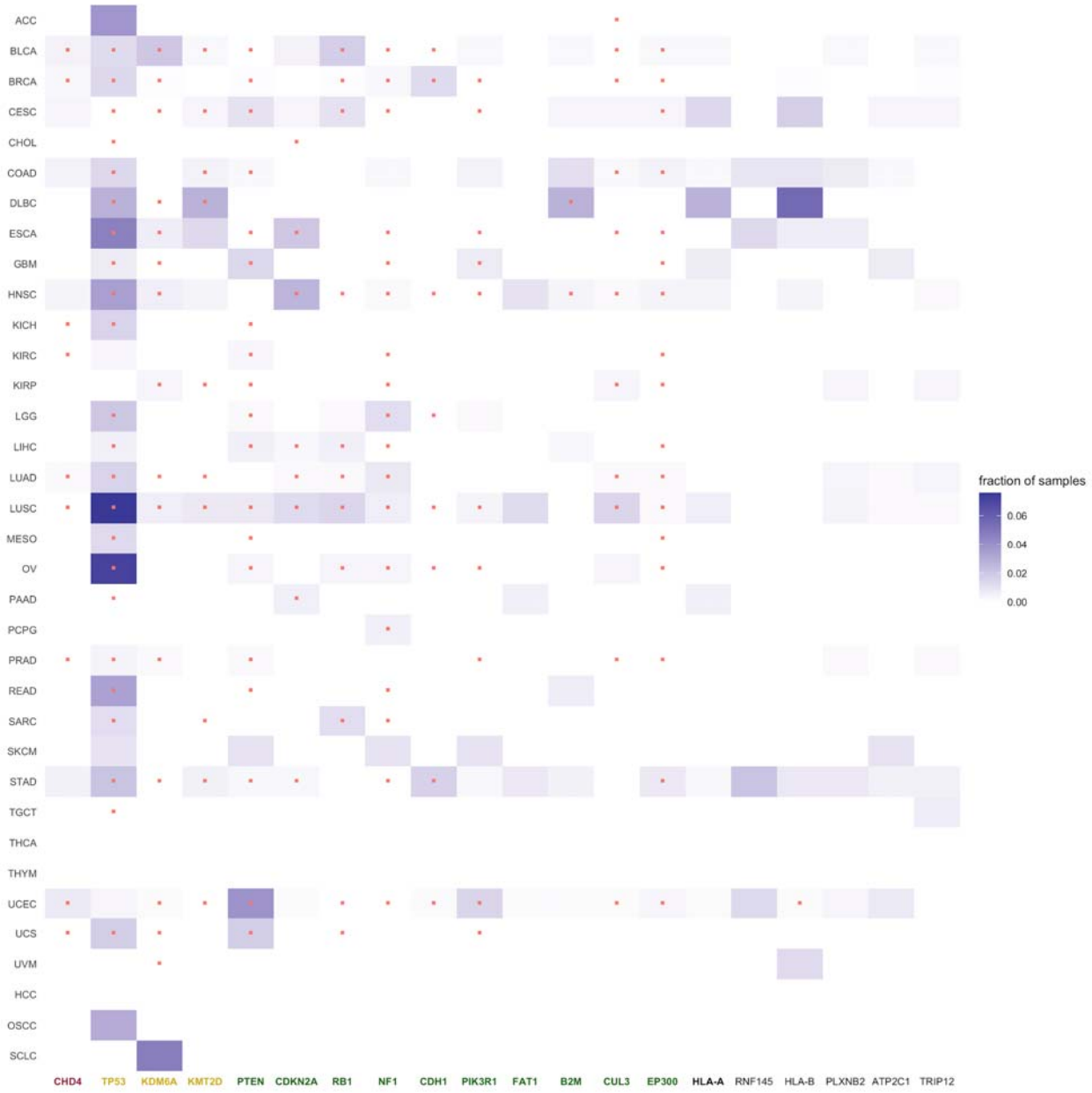


Figure 5. Pan-cancer analysis of cohorts from TCGA and MGI reveals genes recurrently disrupted by variants which cause non-canonical splicing patterns
Results of analysis for recurrently disrupted genes in each cohort. Columns correspond to the 20 most frequently recurring genes, as ranked by fraction of samples. Genes are clustered by whether they were annotated by the CGC as an oncogene (red), an oncogene and tumor suppressor gene (yellow), a tumor suppressor gene (green), or

another type of cancer-relevant gene. Shading corresponds to −log10(p value) and columns represent cancer types. Red marks within cells indicate that the gene was annotated by CHASMplus as a driver within a given TCGA cohort.

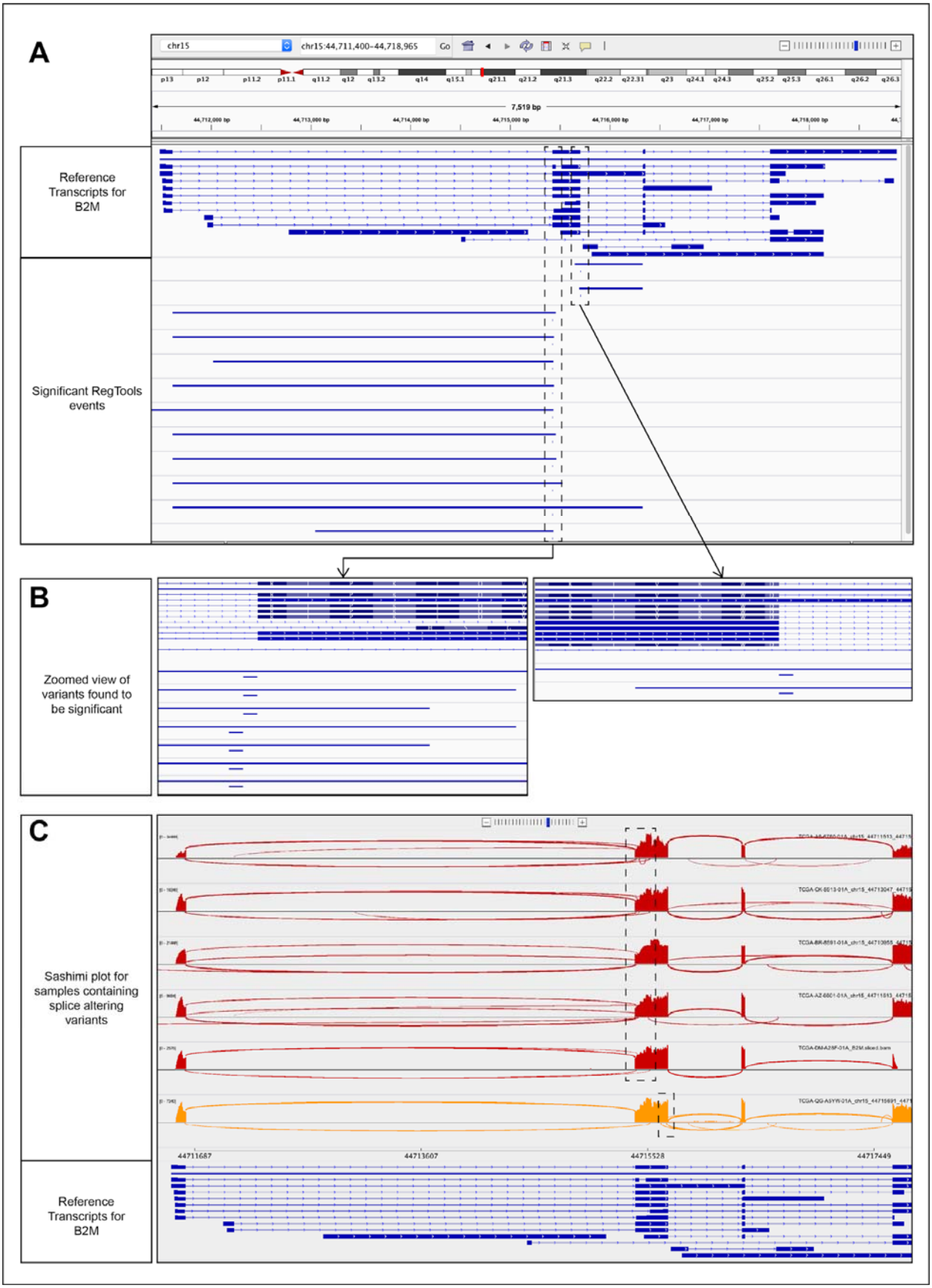
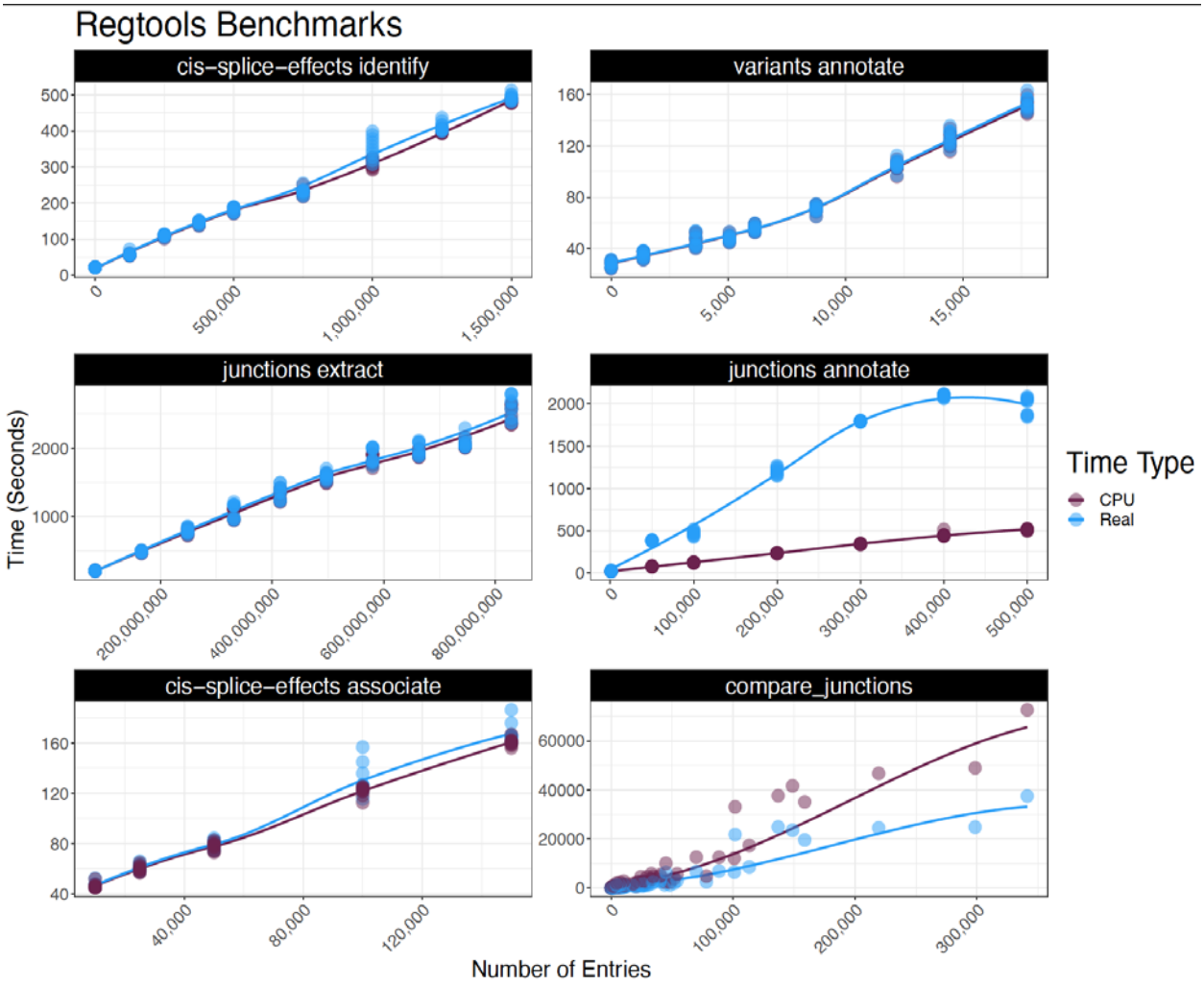


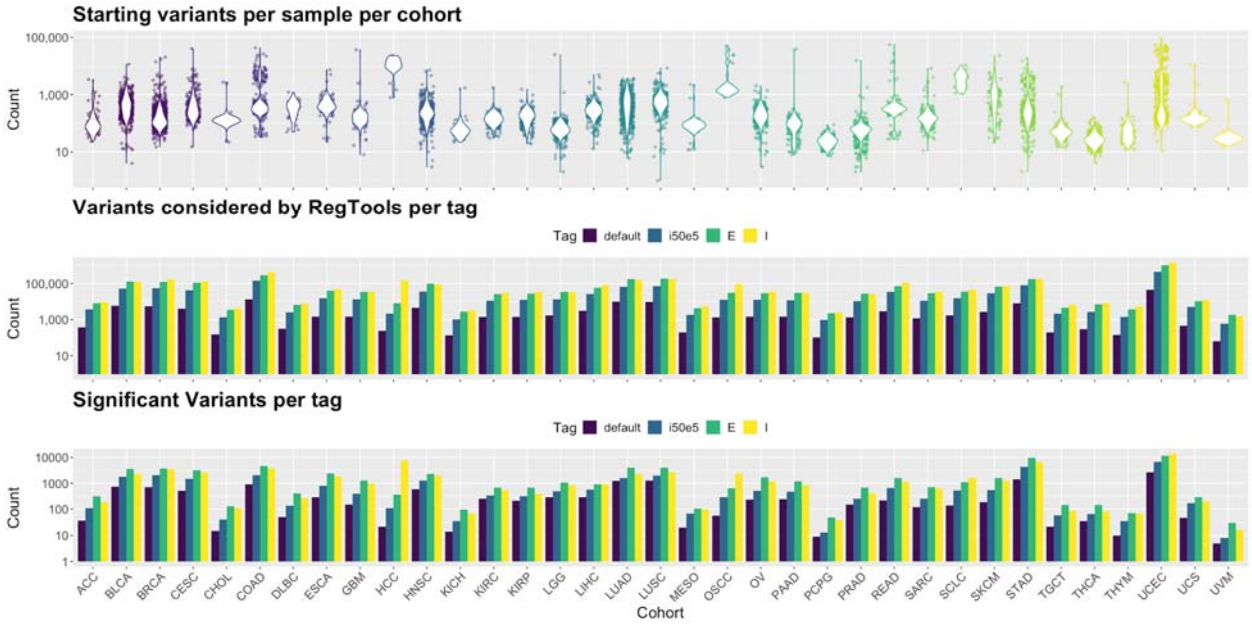
Figure 6. Several SNVs in B2M associated with alternate acceptor and alternate donor usage.

A) IGV snapshot of three intronic variant positions found to be associated with usage of an alternate acceptor and alternative donor site that leads to formation of novel transcript products. This result was found using the default splice variant window parameter (i2e3). **B)** Zoomed in view of the variants identified by RegTools that are associated with alternate acceptor and donor usage. Two of these variant positions flank the acceptor site and one flanks the donor site that are being affected. **C)** Sashimi plot visualizations for samples containing the identified variants that show alternate acceptor usage (red) or alternate donor usage (orange).

Supplemental Figures

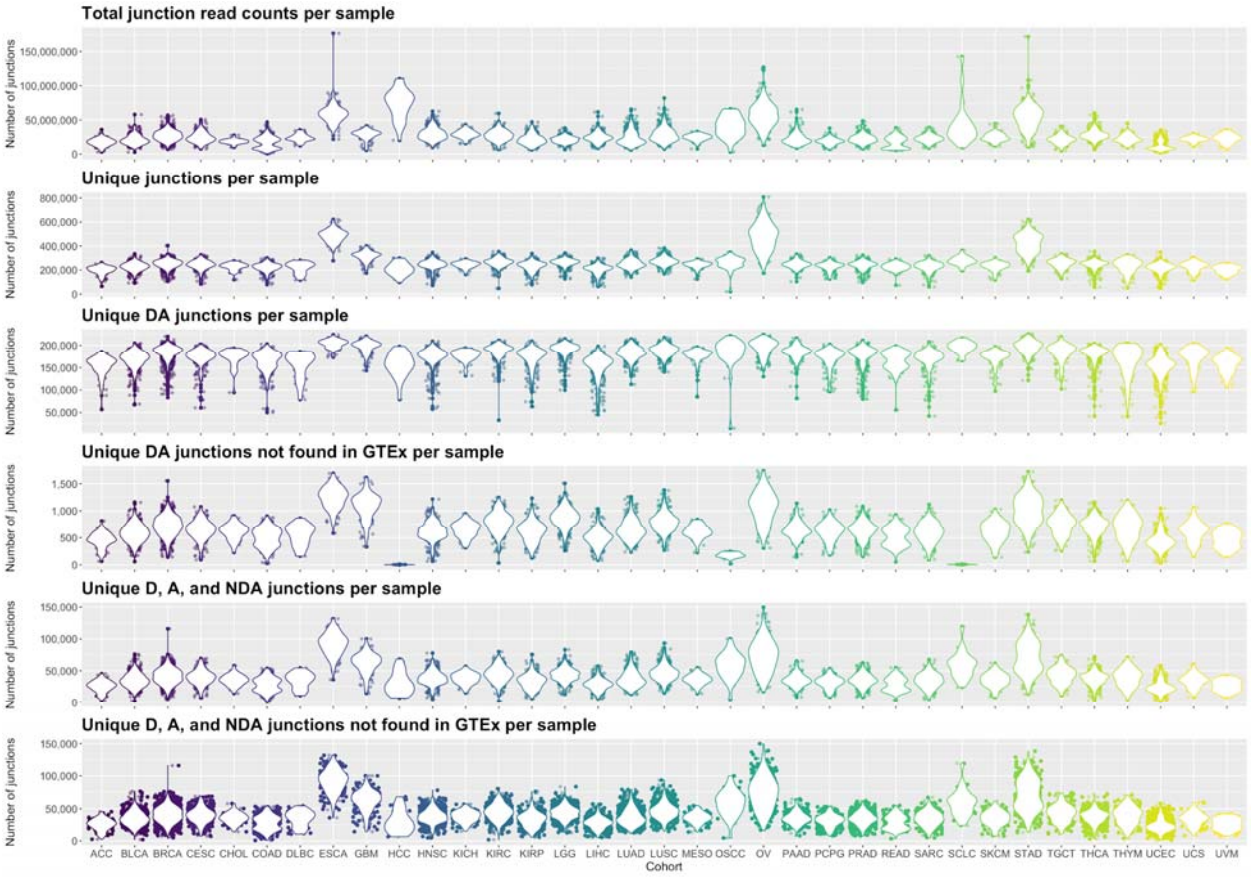


Supplementary Figure 1. Benchmarking of each RegTools command.
The total CPU time (System Time + User Time) and real time are plotted against the number of entries processed for each available RegTools function using 10 total replicates. For the cis-splice-effects identify/cis-splice-effects associate/variants annotate workflows, the number of entries corresponds to the number of somatic variants, whereas the number of entries in the junctions extract/junctions annotate/compare_junctions workflows corresponds to the number of reads processed from a downsampled BAM file, the number of junctions processed, and the number of candidate variant junction pairings processed, respectively. For compare_junctions, candidate variant junction pairings were compared across the number of samples in that cohort, with the largest being 1022 samples that comprise our BRCA cohort. LOESS curves are fitted onto each plot.

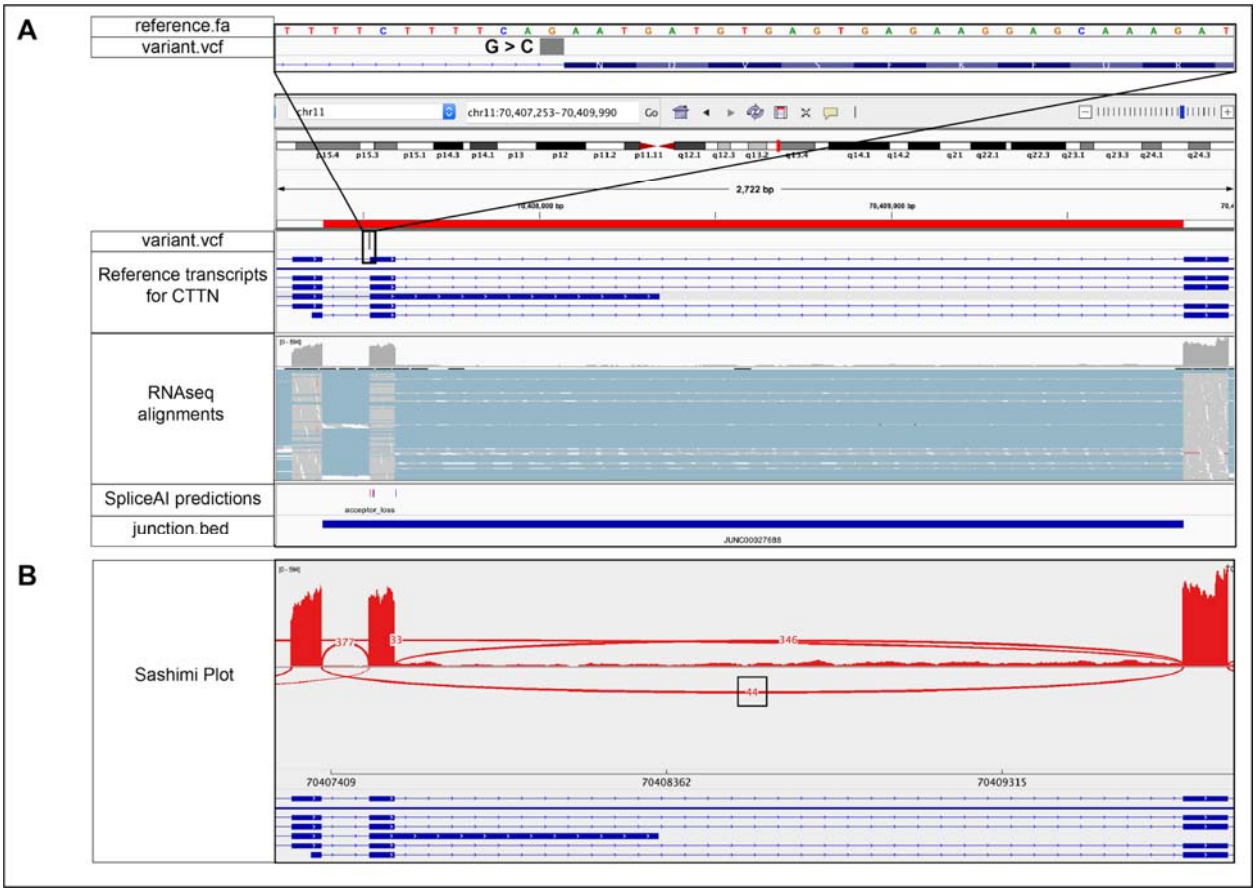


Supplementary Figure 2. Summary of variants analyzed by RegTools in each tumor cohort

Summary of the starting number of high quality variants per sample, the number of initial variants considered for analysis by RegTools for each variant window used per tumor cohort, and the number of significant variants for each variant window used per tumor cohort.

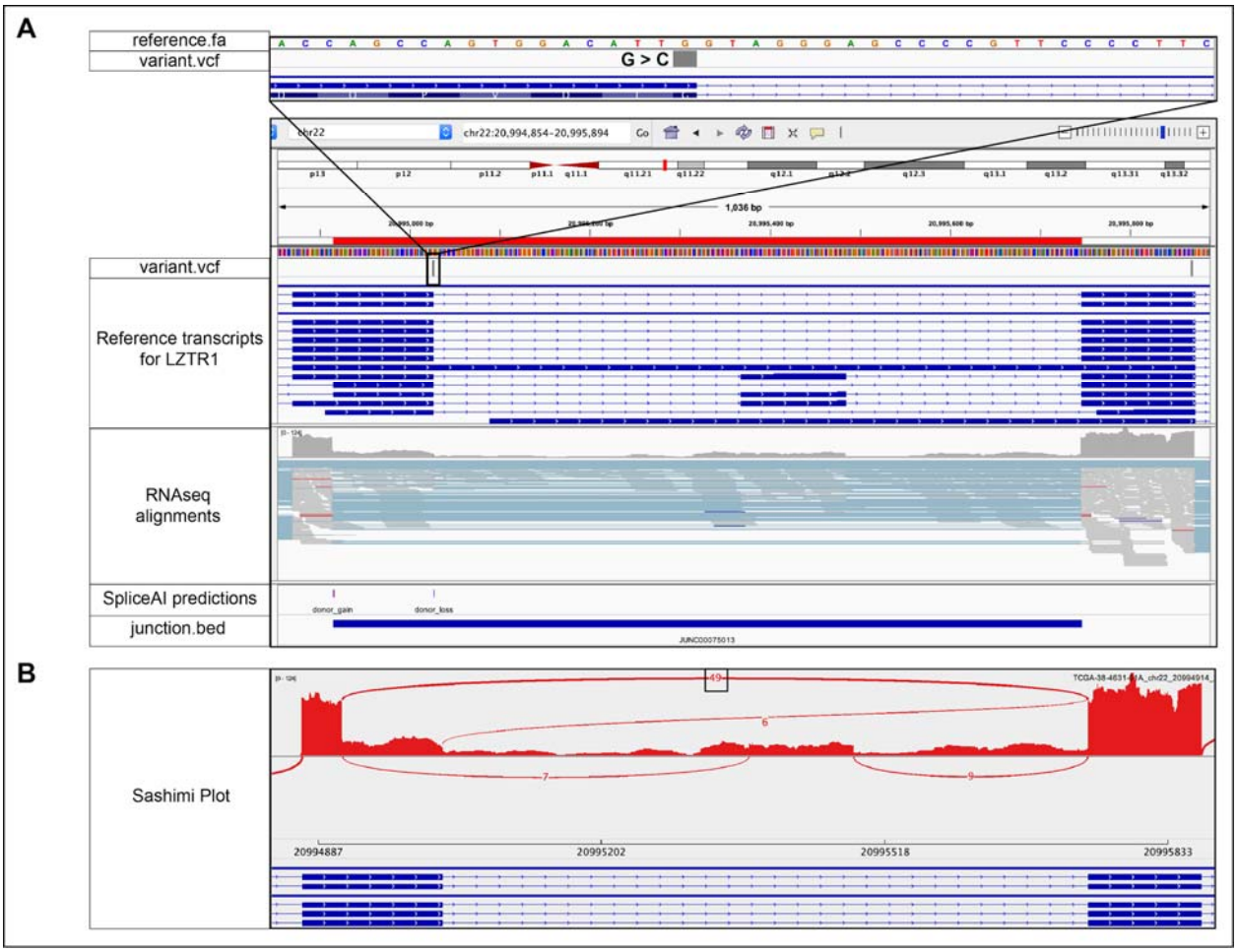


Supplementary Figure 3. Visualization of junctions across cohorts.
Summary of the total junction read counts, unique junctions (all types), unique known (DA) junctions, unique known (DA) junctions not found in GTEx, unique D, A, NDA junctions, and unique D, A, NDA junctions not found in GTEx per sample per cohort.



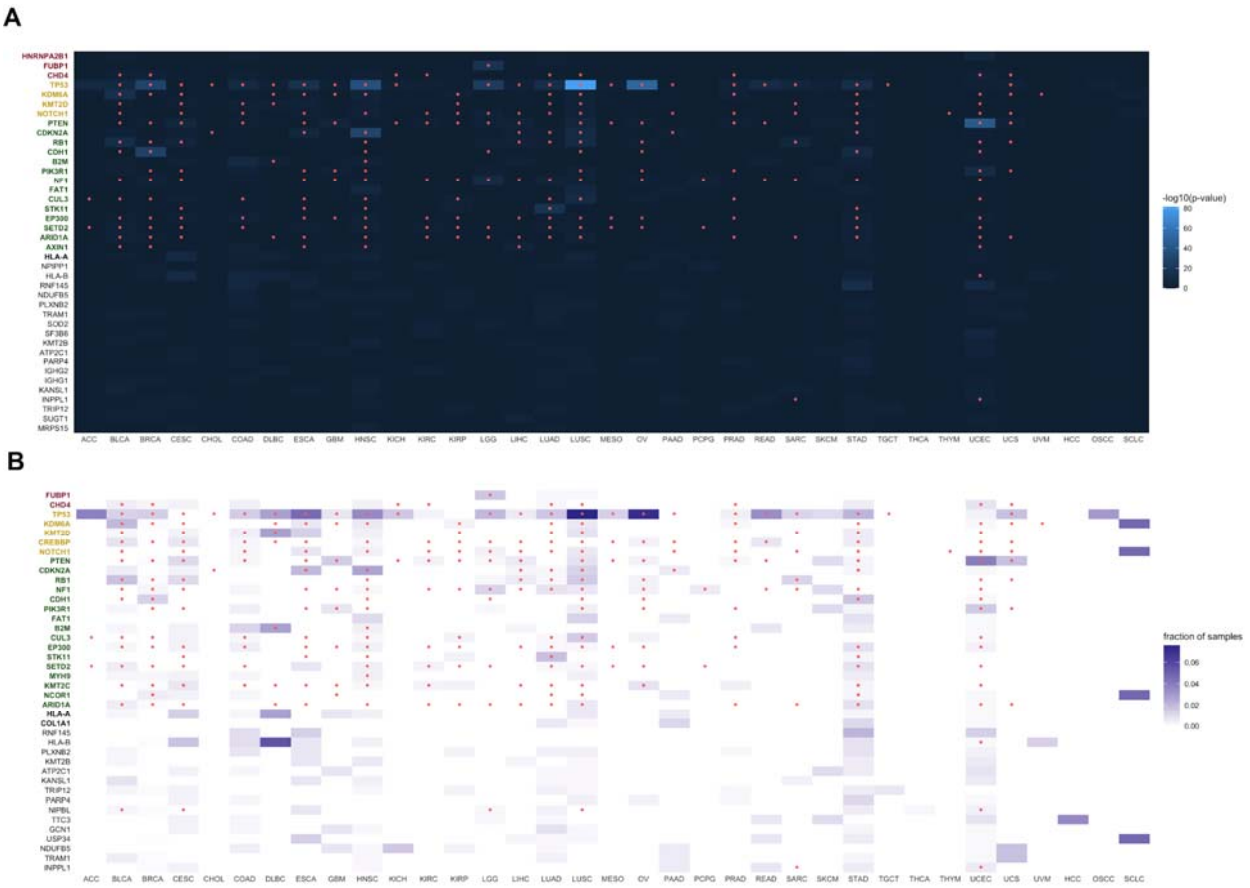
Supplementary Figure 4: Intronic SNV in *CTTN* associated with an exon skipping event.

A) IGV snapshot of a single nucleotide variant (GRCh38, chr11:g.70407517G>C) within an intron of *CTTN* in LUAD sample TCGA-86-6851-01A. This variant is associated with an exon skipping event causing the formation of an NDA junction, JUNC00027688, which has 44 reads of support. The variant was identified by RegTools, VEP, and Veridical but no other tools. This result was found using the default splice variant window parameter (i2e3). **B)** Sashimi plot visualization of the novel junction.



Supplementary Figure 5: Exonic SNV in *LZTR1* associated with alternative donor usage.

A) IGV snapshot of a single nucleotide variant (GRCh38, chr22:g.20995026G>C) within an exon of *LZTR1* in LUAD sample TCGA-38-4631-01A. This variant is associated with the formation of an A junction, JUNC00075013, which has 49 reads of support. The variant was identified by RegTools, VEP, and SpliceAI but no other tools. This result was found using the default splice variant window parameter (i2e3). **B)** Sashimi plot visualization of the novel junction.

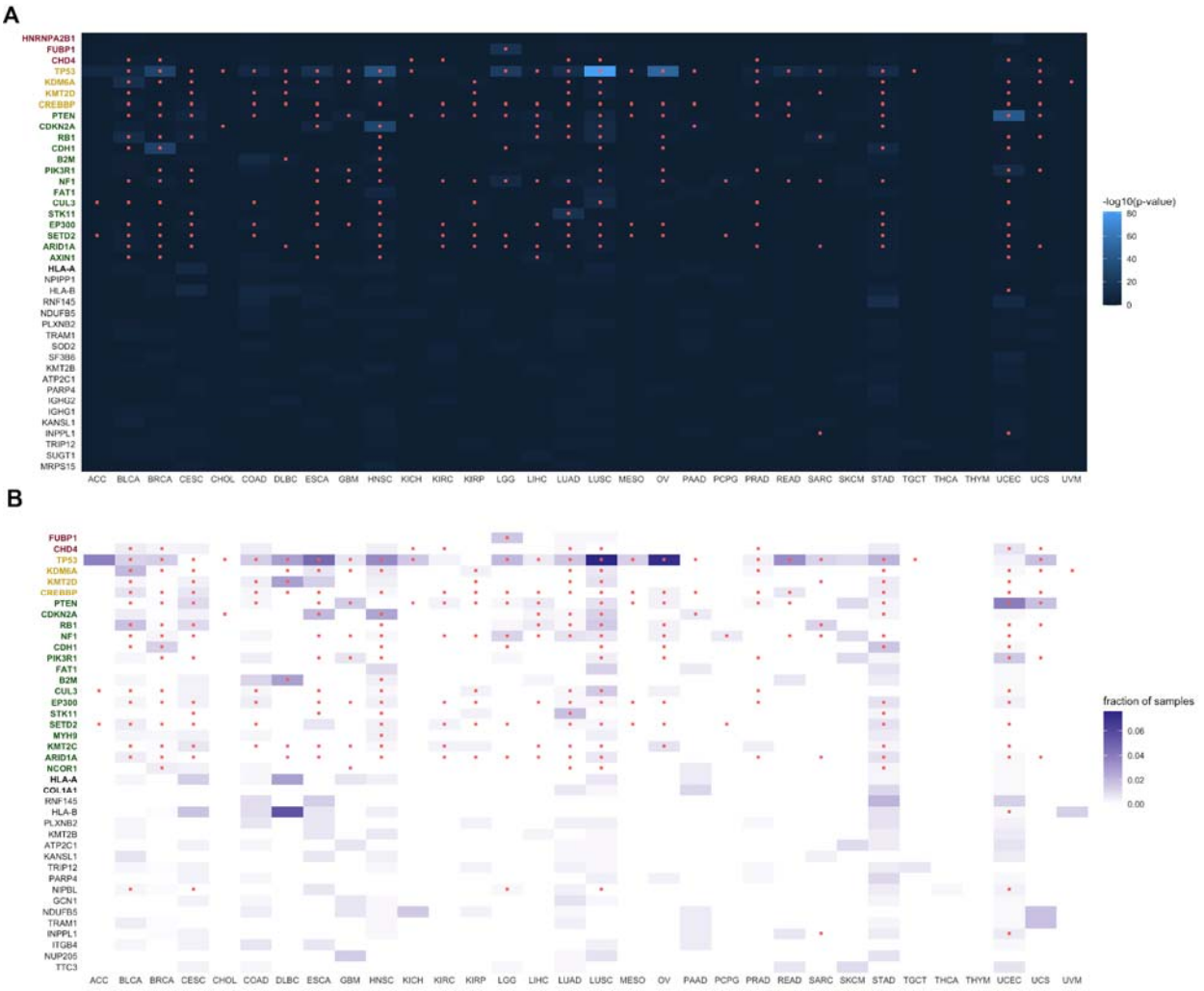


Supplementary Figure 6. Pan-cancer analysis of cohorts from TCGA and MGI reveals genes recurrently disrupted by variants which cause non-canonical splicing patterns

Results of analysis for recurrently disrupted genes in each cohort. **A)** Rows correspond to the 40 most frequently recurring genes, as ranked by binomial p-value. Genes are clustered by whether they were annotated by the CGC as an oncogene (red), an oncogene and tumor suppressor gene (yellow), a tumor suppressor gene (green), or another type of cancer-relevant gene. Shading corresponds to −log10(p value) and columns represent cancer types. Red marks within cells indicate that the gene was annotated by CHASMplus as a driver within a given TCGA cohort. **B)** Rows correspond to the 40 most frequently recurring genes, as ranked by fraction of samples. Shading corresponds to the fraction of samples and columns represent cancer types. Red marks within cells indicate that the gene was annotated by CHASMplus as a driver within a given TCGA cohort.

Supplementary Figure 7. Pan-cancer analysis of cohorts from TCGA and MGI reveals genes recurrently disrupted by variants which promote splicing of particular canonical junctions

Results of analysis for recurrently disrupted genes in each cohort. **A)** Rows correspond to the 40 most frequently recurring genes, as ranked by binomial p-value. Genes are clustered by whether they were annotated by the CGC as an oncogene (red), an oncogene and tumor suppressor gene (yellow), a tumor suppressor gene (green), or another type of cancer-relevant gene. Shading corresponds to −log10(p value) and columns represent cancer types. Red marks within cells indicate that the gene was annotated by CHASMplus as a driver within a given TCGA cohort. **B)** Rows correspond to the 40 most frequently recurring genes, as ranked by fraction of samples. Shading corresponds to the fraction of samples and columns represent cancer types. Red marks within cells indicate that the gene was annotated by CHASMplus as a driver within a given TCGA cohort.



Supplementary Figure 8. TCGA pan-cancer analysis reveals genes recurrently disrupted by variants which cause non-canonical splicing patterns

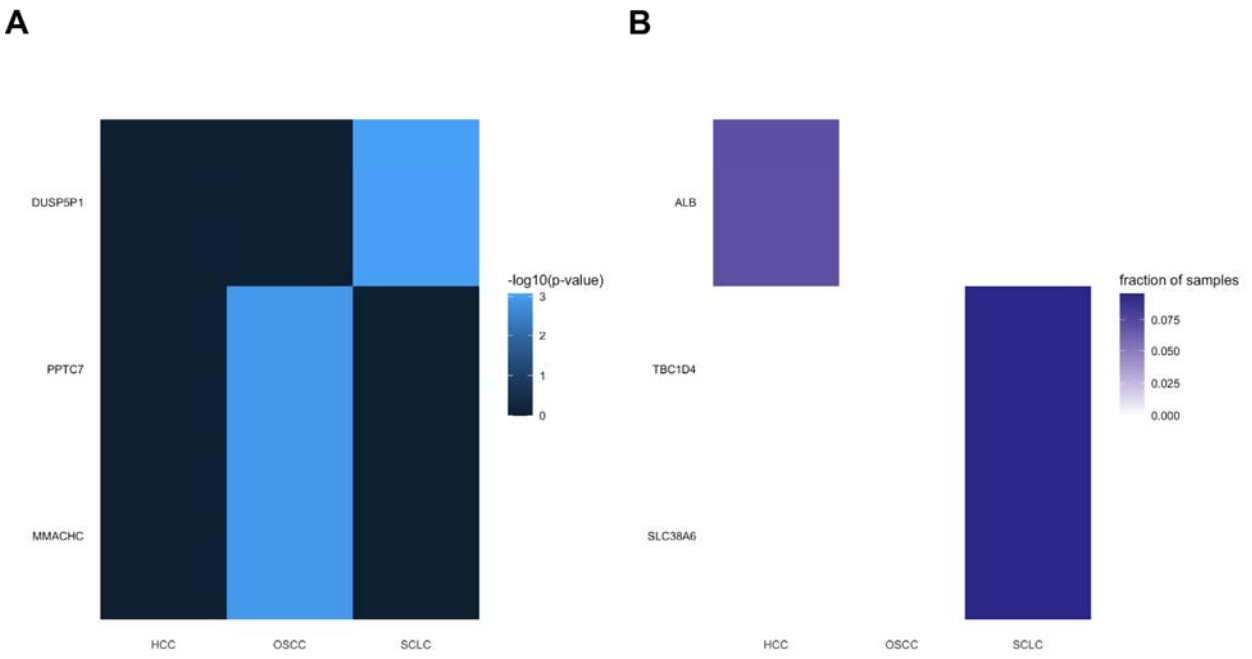
Results of analysis for recurrently disrupted genes in each TCGA cohort. **A)** Rows correspond to the 40 most frequently recurring genes, as ranked by binomial p-value. Genes are clustered by whether they were annotated by the CGC as an oncogene (red), an oncogene and tumor suppressor gene (yellow), a tumor suppressor gene (green), or another type of cancer-relevant gene. Shading corresponds to −log10(p value) and columns represent cancer types. Red marks within cells indicate that the gene was annotated by CHASMplus as a driver within a given TCGA cohort. **B)** Rows correspond to the 40 most frequently recurring genes, as ranked by fraction of samples.

Shading corresponds to the fraction of samples and columns represent cancer types. Red marks within cells indicate that the gene was annotated by CHASMplus as a driver within a given TCGA cohort.

Supplementary Figure 9. TCGA pan-cancer analysis reveals genes recurrently disrupted by variants which promote splicing of particular canonical junctions

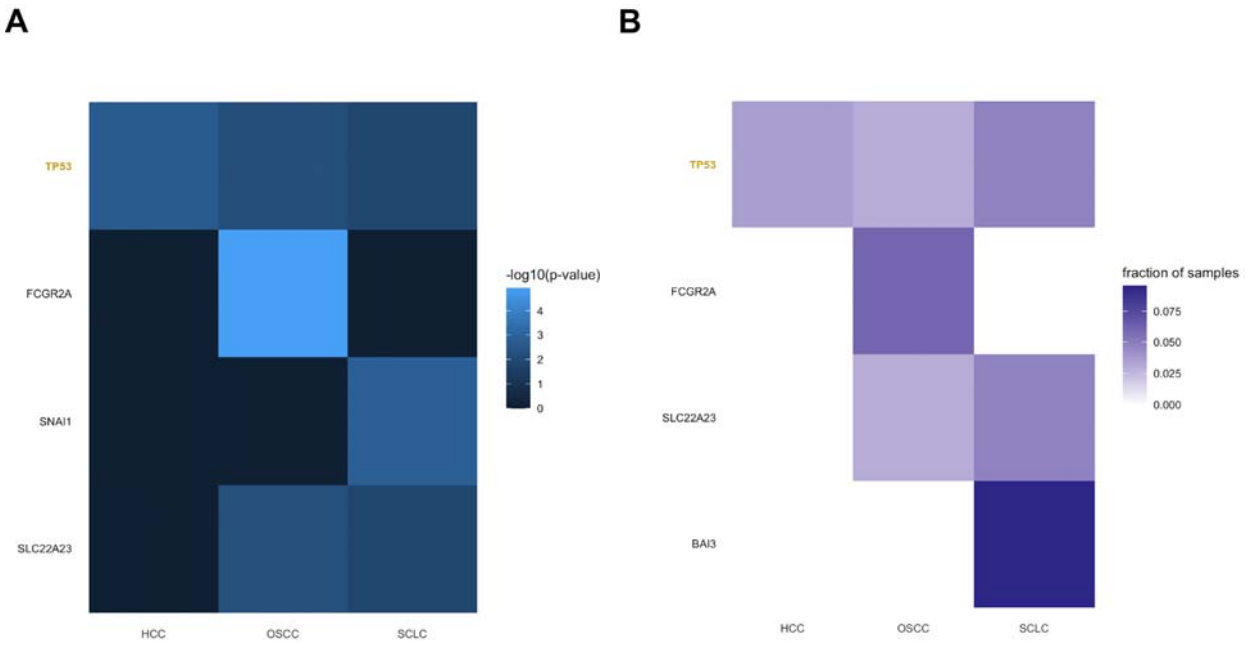
Results of analysis for recurrently disrupted genes in each TCGA cohort. **A)** Rows correspond to the 40 most frequently recurring genes, as ranked by binomial p-value. Genes are clustered by whether they were annotated by the CGC as an oncogene (red), an oncogene and tumor suppressor gene (yellow), a tumor suppressor gene (green), or another type of cancer-relevant gene. Shading corresponds to −log10(p value) and columns represent cancer types. Red marks within cells indicate that the

gene was annotated by CHASMplus as a driver within a given TCGA cohort. **B)** Rows correspond to the 40 most frequently recurring genes, as ranked by fraction of samples. Shading corresponds to the fraction of samples and columns represent cancer types. Red marks within cells indicate that the gene was annotated by CHASMplus as a driver within a given TCGA cohort.



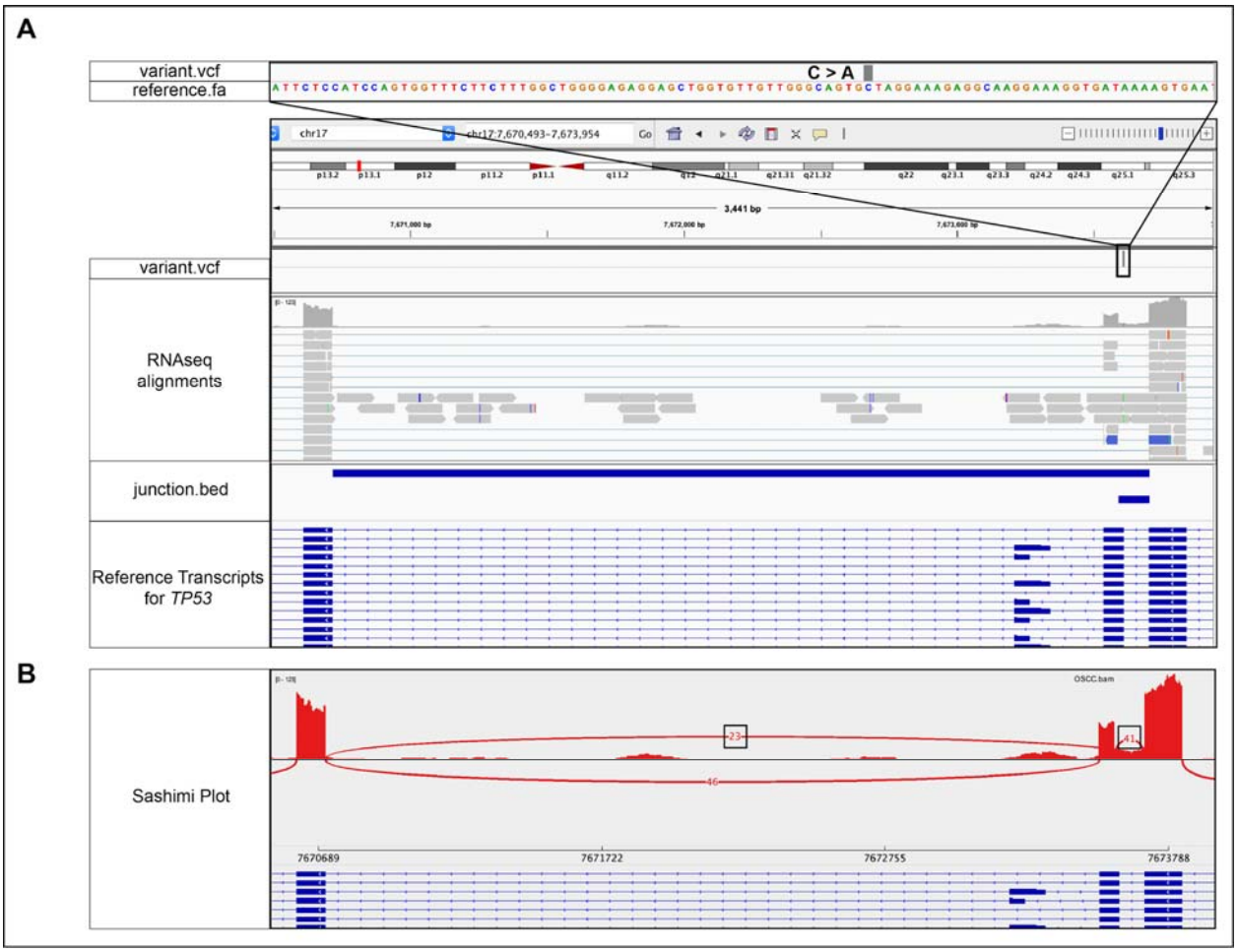
Supplementary Figure 10. Analysis of HCC, OSCC, and SCLC cohorts reveals genes recurrently disrupted by variants which cause non-canonical splicing patterns

Results of analysis for recurrently disrupted genes in each MGI cohort. **A)** Rows correspond to the 3 most frequently recurring genes, as ranked by binomial p-value. Shading corresponds to −log10(p value) and columns represent cancer types. **B)** Rows correspond to the 3 most frequently recurring genes, as ranked by fraction of samples. Shading corresponds to the fraction of samples and columns represent cancer types.



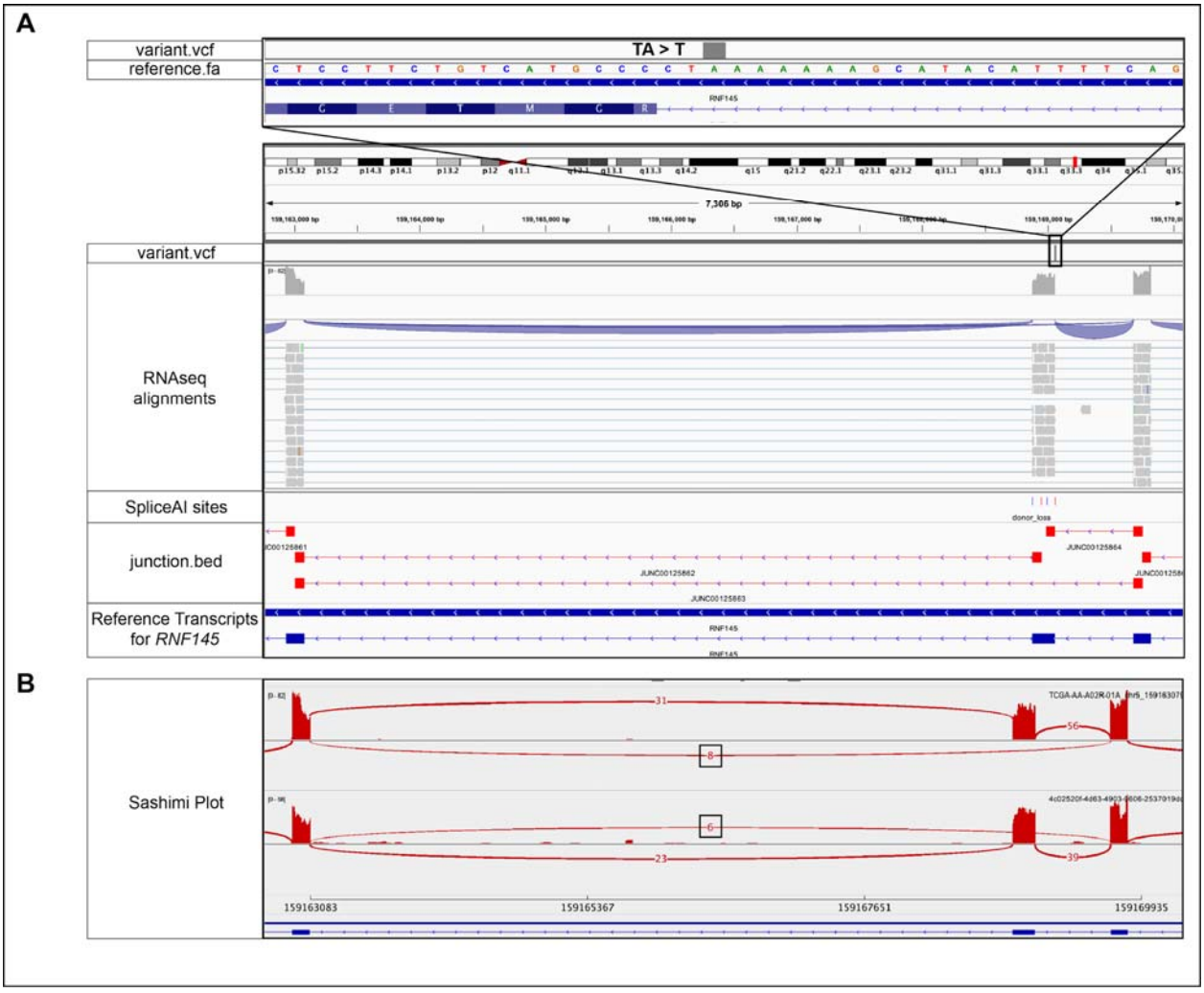
Supplementary Figure 11. Analysis of HCC, OSCC, and SCLC cohorts reveals genes recurrently disrupted by variants which promote splicing of particular canonical junctions

Results of analysis for recurrently disrupted genes in each TCGA cohort. **A)** Rows correspond to the 4 most frequently recurring genes, as ranked by binomial p-value. Shading corresponds to −log10(p value) and columns represent cancer types. **B)** Rows correspond to the 4 most frequently recurring genes, as ranked by fraction of samples. Shading corresponds to the fraction of samples and columns represent cancer types.



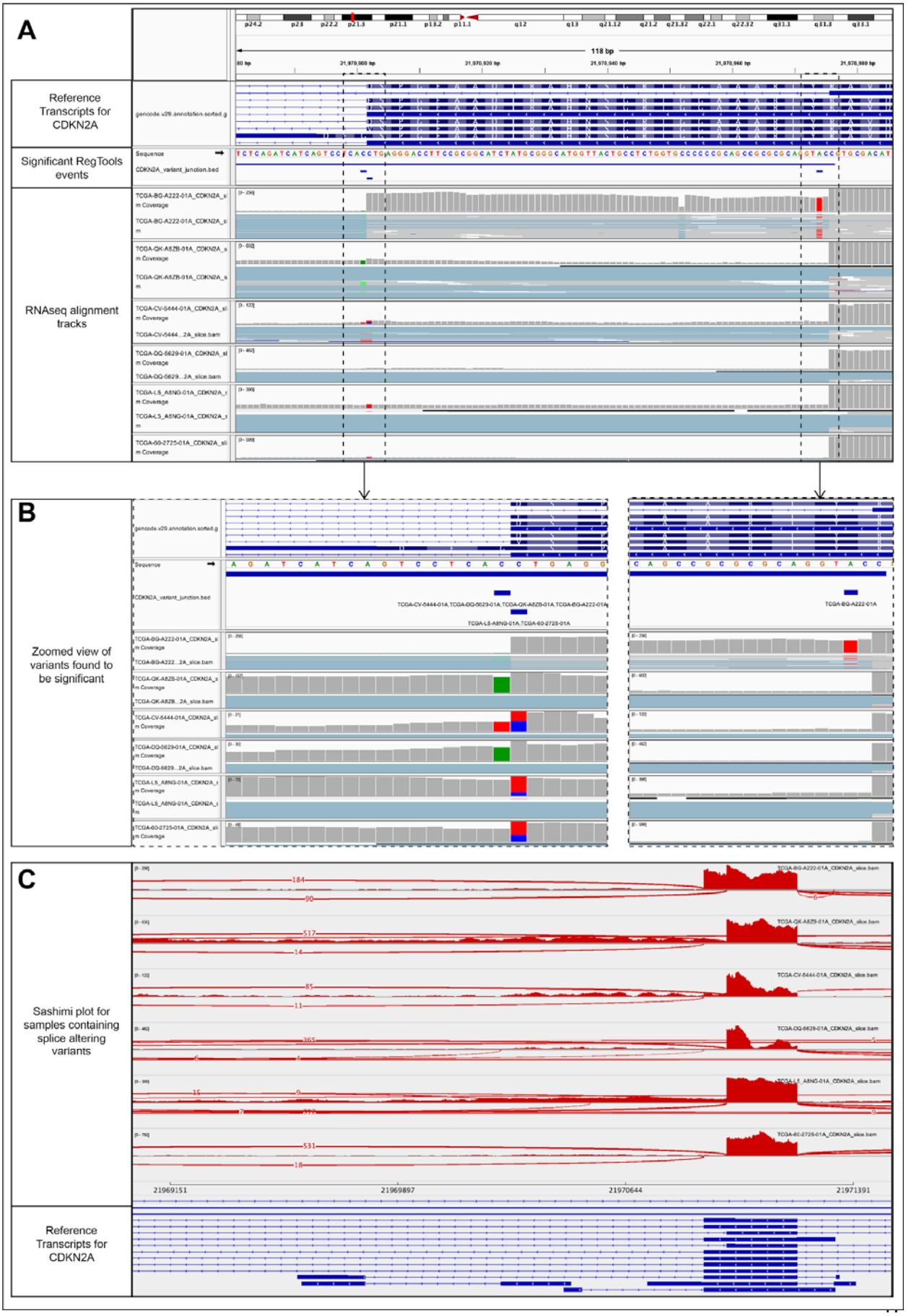
Supplementary Figure 12: Intronic SNV in *TP53* associated with alternative donor usage.

A) IGV snapshot of a single nucleotide variant (GRCh38, chr17:g.7673609C>A) within an intron of *TP53* in an OSCC sample. This variant is associated with an exon skipping event with 23 reads of support and an alternate acceptor site usage with 41 reads of support. This result was found using the default splice variant window parameter (i2e3). **B)** Sashimi plot visualization of the novel junction.



Supplementary Figure 13: Intronic deletion in *RNF145* associated with alternative donor usage.

A) IGV snapshot of a single nucleotide variant (GRCh38, chr5:g.159169058delA) within an intron of *RNF145* in COAD samples. This variant is associated with an exon skipping event with 8 and 6 reads of support for the samples shown. This result was found using the default splice variant window parameter (i2e3). **B)** Sashimi plot visualization of the novel junction.



Supplementary Figure 14: Several SNVs in *CDKN2A* associated with alternate donor usage.

A) IGV snapshot of three variant positions in *CDKN2A* found to be associated with usage of an alternate donor site that leads to formation of an alternate known transcript. This result was found using the default splice variant window parameter (i2e3) for known (DA) junctions. **B)** Zoomed in view of the variants identified by RegTools that are associated with alternate donor usage. Two of these variant positions flank the donor site that is no longer being used. **C)** Sashimi plot visualizations for samples containing the identified variants that show alternate donor usage.