Genome Biology

METHOD Open Access

MMSplice: modular modeling improves the predictions of genetic variant effects on splicing

Jun Cheng[1,2], Thi Yen Duong Nguyen[1], Kamil J. Cygan[3,4], Muhammed Hasan Çelik[1], William G. Fairbrother[3,4], Žiga Avsec[1,2] and Julien Gagneur[1*]

Abstract

Predicting the effects of genetic variants on splicing is highly relevant for human genetics. We describe the framework MMSplice (modular modeling of splicing) with which we built the winning model of the CAGI5 exon skipping prediction challenge. The MMSplice modules are neural networks scoring exon, intron, and splice sites, trained on distinct large-scale genomics datasets. These modules are combined to predict effects of variants on exon skipping, splice site choice, splicing efficiency, and pathogenicity, with matched or higher performance than state-of-the-art. Our models, available in the repository Kipoi, apply to variants including indels directly from VCF files.

Keywords: Splicing, Variant effect, Variant pathogenicity, Deep learning, Modular modeling

Background

Genetic variants altering splicing constitute one of the most important class of genetic determinants of rare [1] and common [2] diseases. However, the accurate prediction of variant effects on splicing remains challenging.

Splicing is the outcome of multiple processes. It is a two-step catalytic process in which a donor site is first attacked by an intronic adenosine to form a branchpoint. In a second step, the acceptor site is cleaved and spliced (i.e., joined) to the 3′ end of the donor site. The sequences of the donor site, of the acceptor site, and of the intronic region surrounding the branchpoint, which are recognized during spliceosome assembly, contribute to splicing regulation [3]. Moreover, many regulatory elements such as exonic splicing enhancers (ESEs) and silencers (ESSs) and intronic splicing enhancers (ISEs) and silencers (ISSs) also play key regulatory roles (reviewed by [4]). In addition to genetic variants at splice consensus sequence, distal elements can also affect splicing and cause disease [5]. Hence, predictive models of splicing need to integrate these various types of sequence elements.

Previous human splice variant interpretation methods can be grouped into two categories.

One category consists of algorithms that score sequence for being bona fide splice regulatory elements including splice sites [6, 7], and exonic and intronic enhancers and silencers [8–13]. Variants can be scored with respect to these regulatory elements by comparing predictions for the reference sequence and for the alternative sequence containing the genetic variant of interest. However, although methods combining several of these scores have been proposed, including Human Splicing Finder [14], MutPred splice [15], and more recently SPiCE [16], the resulting physical and quantitative effect of these variants on splicing remains difficult to assess with these algorithms.

The second category of models aimed at predicting relative amounts of alternative splicing isoforms quantitatively from sequence [17–19]. In this context, a quantitative measure that has retained much attention in the literature is the percent spliced-in (PSI, also denoted Ψ), which quantifies exon skipping. Ψ is defined as the fraction of transcripts that contains a given exon [20]. It can be estimated as the fraction of exon-exon junction reads from an RNA-seq sample supporting inclusion of an exon of interest, over the sum of these reads plus those supporting the exclusion of this exon [20]. Two early models were

*Correspondence: gagneur@in.tum.de
[1] Department of Informatics, Technical University of Munich, Boltzmannstraße, 85748 Garching, Germany
Full list of author information is available at the end of the article

fitted to predict direction of Ψ changes between tissues (exon inclusion, exon skipping, and no change) in mouse [21, 22] from sequence. State-of-the-art models for predicting Ψ from sequence are SPANR [17] and HAL [18] for human, and the model from Jha et al. [23] for mouse. The related quantity Ψ_5 quantifies for a given donor site the fraction of spliced transcripts with a particular alternative 3′ splice site (A3SS). The quantity Ψ_3 has been analogously defined to quantify alternative 5′ splice sites (A5SS) [24]. It should be noted that Ψ_5 is often referred as Ψ for A3SS, and Ψ_3 as Ψ for A5SS (e.g., [25, 26]). However, throughout this manuscript, we are consistently using the notations Ψ_5 and Ψ_3 as defined by Pervouchine et al. [24]. The recently published algorithm COSSMO [19] predicts Ψ_5 from sequence by modeling the competition between alternative acceptor sites for a given donor site and analogously for Ψ_3. COSSMO has shown superior performance over MaxEntScan [7] on predicting the most frequently used splice site among competing ones. Furthermore, splicing efficiency has been proposed to quantify the amount of precursor RNA that undergo splicing (exon-skipped or misspliced transcripts are ignored) at a given splice site by comparing the amount of RNA-seq reads spanning an exon-intron boundary of interest to the corresponding exon-exon junction reads [27]. The latest model to predict variant effects on splicing efficiency is the SMS score, which is based on scores for exonic 7-mers estimated from a recently published saturation mutagenesis assay [28]. However, no model can be applied to all the abovementioned splicing quantities, although they are influenced by common regulatory elements. Furthermore, none of these software handle variant calling format (VCF) files natively, making their integration into genetic diagnostics pipelines cumbersome. Also, these software often do not handle indels (insertions and deletions), although indels are potentially the most deleterious variants.

Here, we trained building block modules separately for the exon, the acceptor site, and the donor site and for intronic sequence close to the donor and close to the acceptor sites. This modular approach allowed leveraging rich datasets from two high-throughput perturbation assays focusing on distinct aspects of splicing: (i) a massively parallel reporter assay (MPRA) with millions of random short sequences in intron and exon sequence [18], and (ii) a high-throughput assay that quantifies the effect of naturally occurring exonic variants on the splicing of their exon [29]. These building block modules could then be combined into distinct models predicting effects of variants on Ψ, Ψ_5, Ψ_3, splicing efficiency, and one model predicting splice variant pathogenicity trained on the database ClinVar [30]. We outperform state-of-the-art models for each task but Ψ_3, on which MMSplice and HAL both are the best. In particular, our model of exon skipping ranked first at the 5th challenge of the Critical Assessment of Genome Interpretation group (CAGI5, https://genomeinterpretation.org/). All our models are available open source in the model zoo Kipoi [31] and can be applied for variant effect prediction directly from VCF files.

Results

Modular modeling strategy

We designed neural networks to score five potentially overlapping splicing-relevant sequence regions: the donor site, the acceptor site, the exon, as well as the 5′ end and the 3′ end of the intron (Fig. 1a). The donor and the acceptor models were trained to predict annotated intron-exon

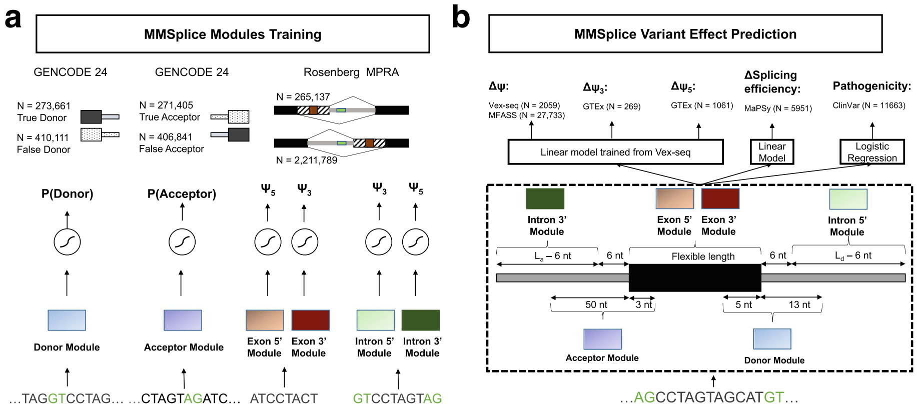


Fig. 1 Individual modules of MMSplice and their combination to predict the effect of genetic variants on various splicing quantities. **a** MMSplice consists of six modules scoring sequences from donor, acceptor, exon, and intron sites. Modules were trained with rich genomics dataset probing the corresponding regulatory regions. **b** Modules from **a** are combined with a linear model to score variant effects on exon skipping ($\Delta\Psi$), alternative donor ($\Delta\Psi_3$), or alternative acceptor site ($\Delta\Psi_5$), splicing efficiency, and they are combined with a logistic regression model to predict variant pathogenicity. L_a and L_d stand for the length of intron sequence taken from the acceptor and donor side respectively

and exon-intron boundaries from GENCODE 24 genome annotation (see the "Methods" section, Fig. 1a, Additional file 1: Figure S1). The exon and intron models were trained from a MPRA that probed the effect of millions of random sequences altering either the exonic 3′ end and the intronic 5′ end for alternative 5′ splicing (A5SS, quantified by Ψ_3), or the exonic 5′ end and the intronic 3′ end for alternative 3′ splicing (A3SS, quantified by Ψ_5) (see the "Methods" section, Fig. 1a, Additional file 1: Figure S2) [18]. For later use, the modules were defined as the corresponding neural network models without the last activation layer. We have two intron modules, the intron 5′ module that scores intron from the donor side and the intron 3′ module that scores intron from the acceptor side. Likewise, we have two exon modules, the exon 5′ module that trained from A3SS and exon 3′ module that trained from A5SS (see the "Methods" section, Additional file 1: Figure S2). To score exonic sequence, only one of the exonic module is applied depending on the alternative splicing quantity. Training data and module architecture are summarized in Table 1. Next, we combined these modules to predict how genetic variants lead to (i) differences in Ψ, (ii) differences in Ψ_3, (iii) differences in Ψ_5, (iv) differences in splicing efficiency,

Table 1 Summary of trained modules and models

MMSplice model	Training data	Architecture	Loss function	Target value	Parameters
Donor module	GENCODE 24, positive: annotated donors, negative: random sequence ("Methods" section)	Four layer neural network with dropout and batch normalization, Additional file 1: Figure S1A	Binary cross entropy	Positive vs. negative	18,049
Acceptor module	GENCODE 24, positive: annotated acceptors, negative: random sequence ("Methods" section)	Two layer conv. neural network with dropout and batch normalization, Additional file 1: Figure S1B	Binary cross entropy	Positive vs. negative	4833
Exon 5′ module	MPRA [18] exonic sequence	One conv. layer shared with the Exon 3′ module, followed with one specific dense layer, Additional file 1: Figure S2	Binary cross entropy	Ψ_5	6145
Exon 3′ module	MPRA [18] exonic sequence	One conv. layer shared with the Exon 5′ module, followed with one specific dense layer, Additional file 1: Figure S2	Binary cross entropy	Ψ_3	6145
Intron 5′ module	MPRA [18] intronic sequence	One conv. layer shared with the Intron 3′ module, followed with one specific dense layer, Additional file 1: Figure S2	Binary cross entropy	Ψ_3	13,825
Intron 3′ module	MPRA [18] intronic sequence	One conv. layer shared with the Intron 5′ module, followed with one specific dense layer, Additional file 1: Figure S2	Binary cross entropy	Ψ_5	13,825
$\Delta\text{logit}(\Psi)$ model	Vex-seq [29]	Linear regression	Huber loss	$\Delta\text{logit}(\Psi)$, Eq. 2	9
Splicing efficiency model (in vivo)	MaPSy ("Methods" section)	Linear regression	Huber loss	Splicing efficiency, Eq. 10	5
Splicing efficiency model (in vitro)	MaPSy ("Methods" section)	Linear regression	Huber loss	Splicing efficiency, Eq. 10	5
Pathogenicity model (w/o phyloP and CADD)	ClinVar [30] [− 10, 10] around donor, [− 40, 10] around acceptor	Logistic regression	Binary cross entropy	Pathogenic vs. benign	14
Pathogenicity model (with phyloP and CADD)	ClinVar [30] [− 10, 10] around donor, [− 40, 10] around acceptor	Logistic regression	Binary cross entropy	Pathogenic vs. benign	18

and (v) to disease or benign phenotypes according to the ClinVar database (Fig. 1b). Specifically, we trained one linear model on top of the modules to predict $\Delta\Psi$. The same linear model was applied to predict $\Delta\Psi_5$ and $\Delta\Psi_3$ by modeling the competition of two alternative exons. Another linear model was trained to predict change of splicing efficiency and a logistic regression model was trained to predict variant pathogenicity from the modules (Fig. 1b).

MMSplice improves the prediction of variant effect on exon skipping

To assess the performance of MMSplice for predicting effects of variants on exon skipping, we first considered the Vex-seq dataset [29]. Vex-seq is a high-throughput reporter assay that compared Ψ for constructs containing a reference sequence to Ψ for matching constructs containing one of 2059 Exome Aggregation Consortium (ExAC [32]) variants. The difference of Ψ for the variant allele to the reference allele is denoted $\Delta\Psi$. These variants consisted of both single nucleotide variants as well as indels from exons and introns (20 nt upstream, 50 nt downstream). The data for the HepG2 cell line was accessed through the Critical Assessment of Genome Interpretation (CAGI) competition [33]. The 957 variants from chromosome 1 to chromosome 8 were provided as training data. The remaining 1054 variants from chromosome 9 to 22 and chromosome X were held out for testing by the CAGI competition organizers and were not available throughout the development of the model. The test data consisted of 572 exonic and 526 intronic variants and included 44 indels.

The Vex-seq experiment is an exon skipping assay, whereas our exon modules were trained for A5SS (Ψ_3) and A3SS (Ψ_5). Because of high redundancy between these two modules, we used the exon 5′ module as it was better at predicting exon skipping exonic variants on Vex-seq training data than the exon 3′ module ($R = 0.52$ v.s $R = 0.25, P = 0.001$, bootstrap, Additional file 1: Figure S3).

We built an MMSplice predictor for $\Delta\Psi$ by training a linear model to combine the modular predictions and interaction terms between modules with overlapping scored regions from the Vex-seq training data (see the "Methods" section, Eq. 2). We compared MMSplice with three state-of-the-art splicing variant scoring models: SPANR [17], HAL [18], and MaxEntScan [7] on the held-out Vex-seq test data ("Methods" section). The methods HAL [18] and SPANR [17] have been reported to be the two best performed existing methods on a recent large-scale perturbation assay probing 27,733 rare variants [34], while MaxEntScan [7] was considered as a baseline reference model. SPANR scores exonic and intronic SNVs up to 300 nt around splice junctions. HAL scores exonic and donor (6 nt to the intron) variants. MaxEntScan scores $[-3, +6]$ nt around the donor and $[-20, +3]$ nt around the acceptor sites. The Vex-seq data was processed the same way for these models ("Methods" section). Unlike the other methods, SPANR does not take custom input sequences and could therefore score single nucleotide variants but not for indels. We evaluated the performance of $\Delta\Psi$ predictions of MMSplice, HAL, and SPANR using root-mean-square errors (RMSE) on test data. MaxEntScan scores sequences but does not predict Ψ. We therefore compared the correlation of differences of MaxEntScan scores to $\Delta\Psi$ and used Pearson correlation on test data as a common metric to compare all these methods.

On the Vex-seq data, MMSplice showed a large improvement over HAL and SPANR. First, MMSplice could score all 1098 variants of the test set whereas HAL could only score 572 (52.1%) and SPANR 966 (88%) of them. Second, the difference in Ψ predicted by MMSplice correlated better when restricted to the respective variants scored by the other methods ($R = 0.68$ for MMSplice v.s. $R = 0.44, 0.26$ for HAL and SPANR respectively, both comparison $P = 0.001$, bootstrap, Fig. 2b–d). A higher performance than other models was also obtained even when we bluntly summed the prediction scores from the five modules without fitting any parameter to the Vex-seq training data ($R = 0.66$ and $R = 0.67$ when using the exon 3′ module in place of the exon 5′ module, Additional file 1: Figure S4). This shows that the superior performance of our model is primarily due to the modules not the combination linear model that was trained from Vex-seq training data. Moreover, MMSplice showed higher accuracy than HAL and SPANR on these data when considering root-mean-square errors (RMSE = 0.1 for MMSplice versus 0.28 for HAL and 0.14 for SPANR, Fig. 2b–d).

We further compared our prediction for donor and acceptor site variants with the popular model MaxEntScan [7]. MMSplice performed better both in donor sequence ($R = 0.87$ for MMSplice versus 0.66 for MaxEntSan5, $P = 0.001$, bootstrap, Additional file 1: Figure S5) and acceptor sequence ($R = 0.81$ for MMSplice versus 0.69 for MaxEntSan3, $P = 0.001$, bootstrap, Additional file 1: Figure S6), when restricted to the subset of variants that MaxEntScan3 could score (42 donor variants and 149 acceptor variants). HAL performed better ($R = 0.71$) than MaxEntScan5 ($R = 0.66$) but worse than MMSplice ($R = 0.87$) on donor variants ($P = 0.001$ for both comparisons, bootstrap, Additional file 1: Figure S5).

Altogether, MMSplice outperformed SPANR, HAL, and MaxEntScan on predicting effects of genetic variants on exon skipping observed on this large-scale perturbation data, by covering more variants and also by providing more accurate predictions. Our model also ranked the first in the 2018 CAGI Vex-seq competition. A joint

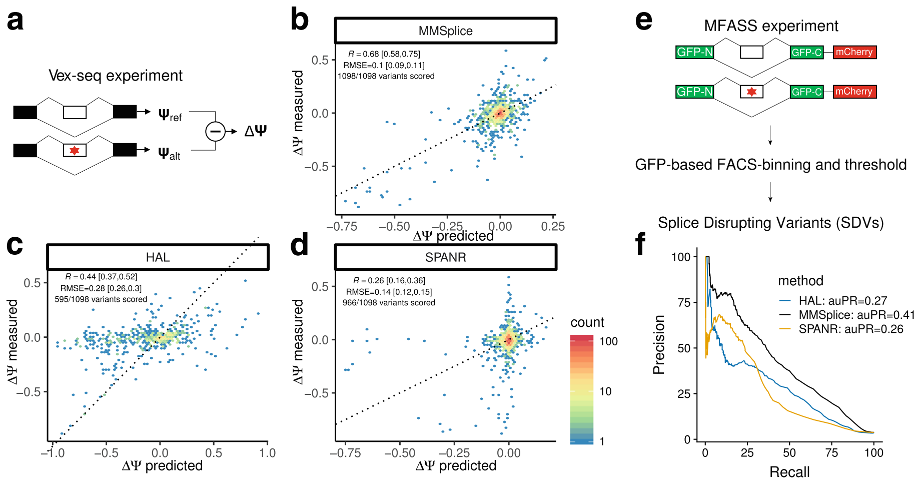


Fig. 2 MMSplice improves the prediction of variant effect on exon skipping. **a** Schema of the Vex-seq experiment [29]. The effect of 2059 ExAC variants (red star) from or adjacent to 110 alternative exons were tested with reporter genes by measuring percent splice-in of the reference sequence (Ψ_{ref}) and of the alternative (Ψ_{alt}) by RNAseq. **b**–**d** Measured (*y*-axis) versus predicted (*x*-axis) Ψ differences between alternative and reference sequence for MMSplice (**b**), HAL [18] (**c**), and SPANR [17] (**d**) on Vex-seq test data. Color scale represents counts in hexagonal bins. The black line marks the $y = x$ diagonal. Each plot is shown with the subset of variants that the considered model can score. Pearson correlations (*R*) and root-mean-square errors (RMSE) were also calculated based on the scored variants. The 95% confidence intervals for these two metrics were calculated with bootstrap ("Methods" section). **(e)** Schema of MFASS experiment [34]. Exon skipping effects of 27,733 ExAC SNVs (red star) spanning or adjacent to 2339 exons were tested by genome integration of designed construct. Splice-disrupting variant (SDV) is defined as a variant that change an exon with original exon inclusion index $\geqslant 0.5$ by at least 0.5. **f** Precision-recall curve of MFASS SDV classification based on model predicted $\Delta\Psi$. Precision-recall curve for all three models was calculated for the sets of variants they can score. MMSplice (black) scored all 27,733 variants, SPANR (yellow) scored 27,663 variants (1,048 SDVs), and HAL (blue) scored 14,353 variants (489 SDVs)

publication with the organizers and challengers is in the planning.

MMSplice classifies rare splice disrupting variants with higher precision and recall

To further compare models on predicting exon skipping level with independent datasets that no model has been trained on, we used the splicing functional assay from Cheung et al. [34]. Cheung et al. found 1050 splice-disrupting variants (SDVs); the majority are extremely rare, after examining 27,733 ExAC single-nucleotide variants (SNV) with Multiplexed Functional Assay of Splicing using Sort-seq (MFASS) (Fig. 2e). The author benchmarked several variant effect prediction methods including conservation-based methods like CADD [35], phastCons [36], and the state-of-the-art splicing variant scoring tools HAL and SPANR. Among all, the two splicing variant scoring methods performed much better than the others, thus MMSplice was compared with those two. MMSplice model with the final combination linear model trained from Vex-seq training data was applied to classify SDVs based on predicted $\Delta\Psi$ solely from sequence. Our model achieved overall higher Area under the precision-recall curve (auPR, MMSplice: 0.41, HAL: 0.27, SPANR: 0.26, $P = 0.001$ for both MMSplice versus HAL and MMSplice versus SPANR, bootstrap) when all models considering only their scored variants (Fig. 2f). In total, MMSplice scored all variants, SPANR scored 99.7% of all variants, while HAL scored only 51.8% of them. When considering exonic variants only, MMSplice (auPR=0.29) performed similar to HAL (auPR = 0.27) ($P = 0.326$, bootstrap, Additional file 1: Figure S7). For intronic variants, MMSplice had an auPR of 0.55 in comparison to 0.43 for SPANR ($P = 0.001$, bootstrap, Additional file 1: Figure S7).

Overall, MMSplice demonstrated a substantital improvement over SPANR for both intronic and exonic variants and showed a similar performance to HAL for classifying exonic SDVs. This result also demonstrates the power of our model to score the effect of rare variants, for which association studies often lack of power.

MMSplice predicts variants associated with competing splice site selection with high accuracy

The MMSplice modular framework allows modeling alternative splicing events other than exon skipping. To demonstrate this and assess the performance of MMSplice on other alternative splicing events, we built MMSplice models to predict association of variants around alternative donors on alternative 5′ splicing (A5SS, Ψ_3) and variants around alternative acceptors on alternative 3′ splicing (A3SS) ("Methods" section) in GTEx. Ψ_5 and Ψ_3 values for homozygous reference variants as well as with heterozygous and homozygous alternative variants were calculated from RNA-seq data of the GTEx consortium [37] ("Methods" section). Here too, our MMSplice models allowed handling indels. One example is the insertion variant rs11382548 (chr11:61165731:C-CA). It is a splice site variant that turns a CG acceptor to an AG acceptor. It showed the largest $\Delta\Psi_5$ among all assessed variants.

We benchmarked MMSplice against MaxEntScan, HAL, and COSSMO. Overall, MMSplice ($R = 0.66$) significantly outperformed COSSMO ($R = 0.5$, $P = 0.016$, bootstrap) and MaxEntScan ($R = 0.46$, $P = 0.001$, bootstrap) and tied with HAL ($R = 0.67$, $P = 0.558$, bootstrap) on predicting $\Delta\Psi_3$ (Fig. 3a–d). On predicting $\Delta\Psi_5$, MMSplice ($R = 0.57$) again significantly outperformed both COSSMO ($R = 0.37$) and MaxEntScan ($R = 0.44$) (all $P = 0.001$, Fig. 3e–g). This conclusion also holds when using RMSE as evaluation metric (Fig. 3). Even though HAL can predict A5SS donor variants well, the model has been trained for predicting A5SS and may not generalize well to other alternative splicing types. It only showed moderate performance when predicting donor variants from Vex-seq skipped exons (Additional file 1: Figure S5). In contrast, MMSplice showed consistent high performance across different types of alternative splicing events.

MMSplice outperformed COSSMO for both donor and acceptor variants even though COSSMO was trained from estimated Ψ_5 and Ψ_3 values from GTEx data. One possible reason is that COSSMO was trained from reference sequence to predict Ψ_5 and Ψ_3, ignoring the genetic variants of the GTEx dataset. In contrast, MMSplice was trained to predict $\Delta\Psi$ from genetic perturbation data (Vex-Seq). Also, COSSMO was trained to predict splice site usage for an arbitrary number of alternative splice sites, while we focused here on the cases with only two alternative splice sites.

Prediction of splicing efficiency

We next used our modular approach to derive a model that predicts splicing efficiency, i.e., the proportion of spliced RNAs among spliced and unspliced RNAs [27]. We have done so in the context of a second CAGI5 challenge (Fig. 4a), whose training dataset is based on a massively parallel splicing assay (MaPSy [27]) and which is described in the "Methods" section. This MaPSy dataset consists of splicing efficiencies, 5761 pairs of matched wild-type and mutated constructs, where each mutated construct differed from its matched wild-type by one exonic non-synonymous single-nucleotide variant ("Methods" section). The assay has been done both with an in vitro splicing assay and in vivo by transfection into HEK293 cells ("Methods" section). A test set of 797 construct pairs was held-out during the development of the model.

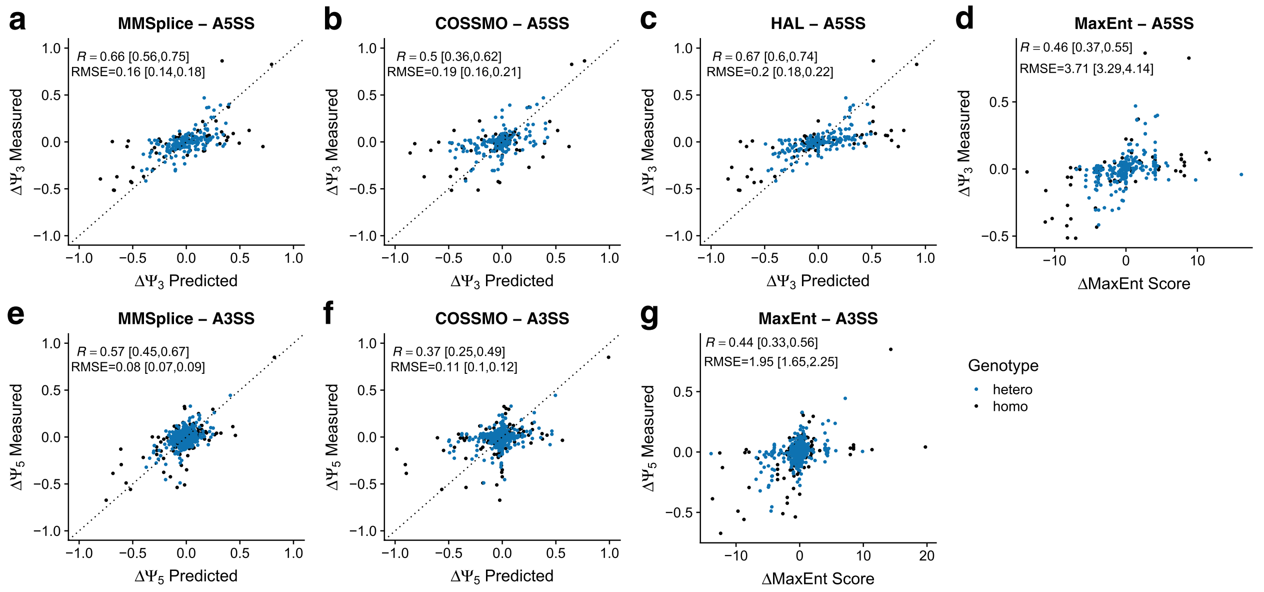


Fig. 3 Evaluation of models predicting $\Delta\Psi_5$ and $\Delta\Psi_3$ on the GTEx dataset. Associated effects (*y*-axis) versus predictions (*x*-axis) for GTEx variants around alternative spliced donors (3 nt in the exon and 6 nt in the intron) and acceptors (3 nt in the exon and 20 nt in the intron) were considered. Ψ_5 (or Ψ_3) of homozygous (black) and heterozygous (blue) alternative variants as well as homozygous reference variants were calculated by taking the mean Ψ_5 (or Ψ_3) across individuals with the same genotype (excluding individuals with multiple variants within 300 nt around splice sites) on brain and skin (not sun exposed) samples. For donor variants, MMSplice (**a**) was benchmarked against COSSMO (**b**), HAL (**c**), and MaxEntScan (**d**). For acceptor variants, MMSplice (**e**) was benchmarked against COSSMO (**f**) and MaxEntScan (**g**). The 95% confidence intervals for Pearson correlation (*R*) and root-mean-square errors (RMSE) were calculated with bootstrap ("Methods" section). The dotted line marks the $y = x$ diagonal

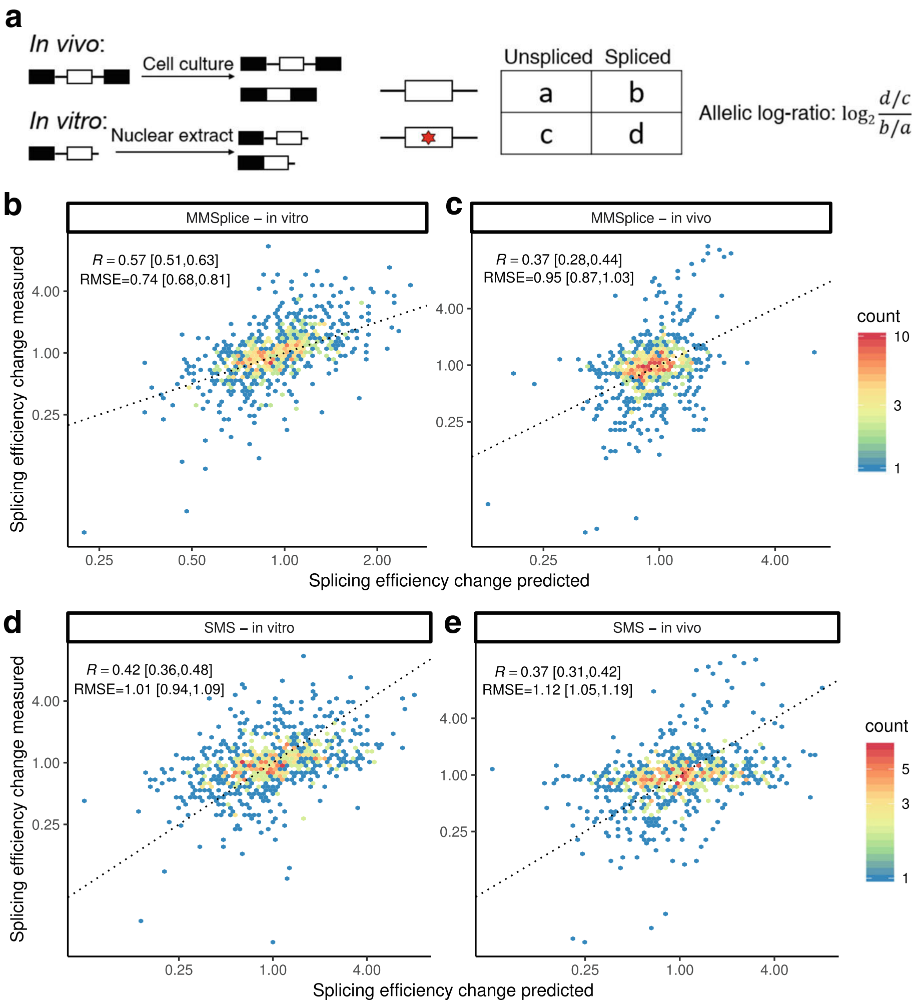


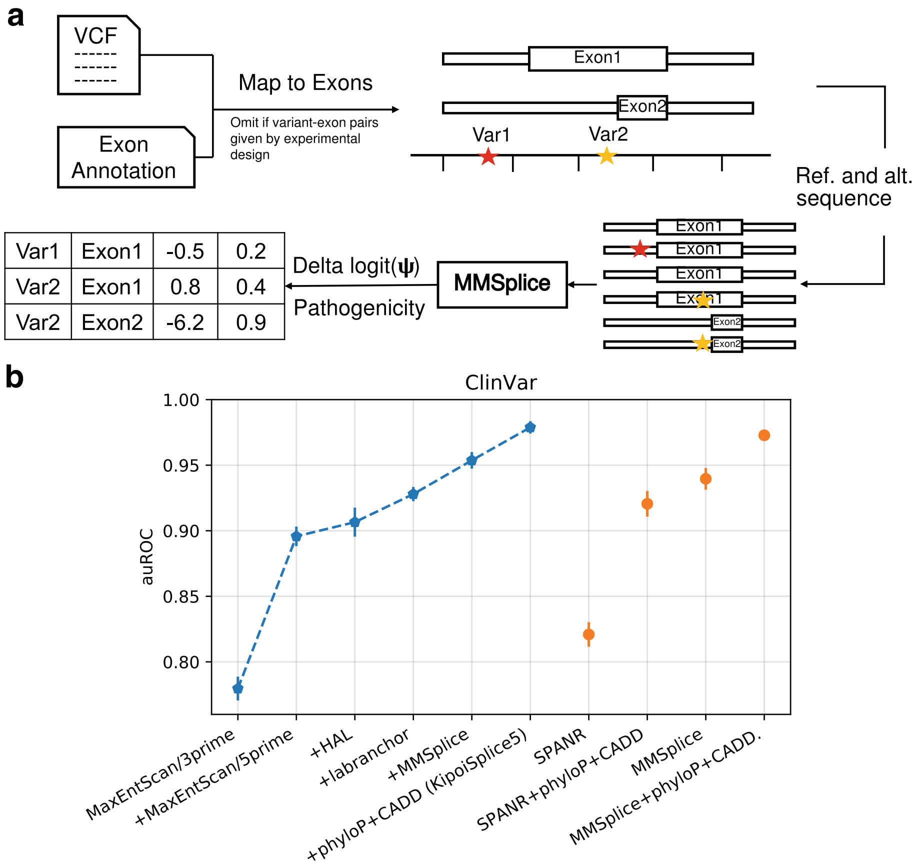
Fig. 4 Splicing efficiency prediction. **a** MaPSy experiment ("Methods" section). Effect of 5761 published disease-causing exonic mutations on splicing efficiency is measured both in vivo and in vitro. Changes of splicing efficiency were quantified by allelic log-ratio. **b**–**e** Measured (*y*-axis) versus predicted (*x*-axis) allelic ratio for 797 variants in the test set for MMSplice (**b**, **c**) and the SMS score [28] (**d**, **e**). The dotted line marks the $y = x$ diagonal. The 95% confidence intervals for Pearson correlation (*R*) and root-mean-square errors (RMSE) were calculated with bootstrap ("Methods" section)

We trained a linear model on top of the modular predictions with MaPSy training data to predict differential splicing efficiency reported by the MaPSy data ("Methods" section). This linear model was trained the same way as for Vex-seq except that the response was the allelic log-ratio (Fig. 4a and "Methods" section) instead of $\Delta\text{logit}(\Psi)$. One model was trained for the in vivo data and another model was trained for the in vitro data. Our MMSplice model for differential splicing efficiencies predicted the effect of those non-synonymous mutations on the held-out test set reasonably well in vitro ($R = 0.57$, 4a) and well in vivo ($R = 0.37$, 4c). Also, our MMSplice model for differential splicing efficiencies outperformed the SMS score algorithm [28] on in vitro data ($P = 0.001$, bootstrap, 4d) and reached similar performance on the in vivo data ($P = 0.524$, bootstrap, 4e). MMSplice significantly outperformed SMS scores in both conditions when evaluated with RMSE (0.74 and 0.95 for MMSplice versus 1.01 and 1.12 for SMS scores, $P = 0.001$ for both comparison, bootstrap). Several reasons may have led to the worse performance in vivo. One possible reason is that the in vivo assay may involve RNA degradation factors, which also regulate level of spliced RNA species by regulating RNA stability. Another possible reason is that the folding of RNAs in vivo may be more complex than in vitro, which in turn affects splicing [38], making the prediction in vitro more difficult.

MMSplice can contribute to improved predictions of splice site variant pathogenicity

Predicting variant pathogenicity is a central task of genetic diagnosis. However, large amount of variants are annotated as variant of uncertain significance (VUS). A good splice variant effect prediction model can help interpreting VUSs. To evaluate the potential of MMSplice to contribute in predicting variant pathogenicity, we considered the ClinVar variants (version 20180429, [30]) that lie between 40 nt 5′ and 10 nt 3′ of an acceptor site or 10 nt either side of a donor site of a protein coding gene (Ensembl GRCh37 v75 annotation, "Methods" section) as potentially affecting splicing. Among these variants, we aimed at discriminating between the 6310 variants classified as pathogenic and the 4405 variants classified as benign. To this end, we built an MMSplice model that implements a logistic regression on top of the MMSplice modules ("Methods" section). Variants can potentially be in the vicinity of multiple exons. MMSplice handles this many-to-many relationship (Fig. 5a). Conveniently, MMSplice can be applied to a variant file in the standard format VCF [39] and a genome annotation file in the standard GTF format. Moreover, MMSplice is available as a Variant Effect Predictor Plugin (VEP [40]).

This MMSplice model was benchmarked against SPANR [17] and the ensemble of three other models: MaxEntScan [7], HAL [18], and the branch point predictor LaBranchoR [41]. We also compared our MMSplice model and competing models with phyloP and CADD scores as additional features (Additional file 1: Supplementary Methods). Model performances were benchmarked under 10-fold cross-validation (Fig. 5b). Globally on all the 10,715 considered variants, MMSplice alone (auROC = 0.940) outperformed SPANR (auROC = 0.821, $P = 0.001$, bootstrap) and the ensemble model combining MaxEntScan, HAL, and LaBranchoR (auROC = 0.928) ($P = 0.001$, bootstrap). Adding MMSplice to the ensemble model further improved the auROC to 0.954 ($P = 0.001$, bootstrap). Moreover, MMSplice with phyloP and CADD features (auROC = 0.973) achieved a performance close to the best ensemble model kipoiSplice5 that included MMSplice (auROC = 0.979, $P = 0.003$, bootstrap, Fig. 5),



Var1	Exon1	-0.5	0.2
Var2	Exon1	0.8	0.4
Var2	Exon2	-6.2	0.9

Fig. 5 Predictions on ClinVar variants. **a** Variants are first mapped to potentially affected exons. Variants in the exon or in the intron, within L_a nt of the acceptor site or within L_d nt from the donor site are considered to affect splicing of the exon. Afterwards, reference and alternative sequences are retrieved and subjected to MMSplice for prediction. MMSplice gives a prediction for each variant-exon pair. **b** Model comparison on classifying pathogenicity of ClinVar splice variants. Models were trained and evaluated in 10-fold cross-validation. Error bars indicate one standard deviation calculated across folds. The six leftmost models (blue) are incrementally added to the ensemble model: "+phyloP+CADD " uses all five previous models as well as phyloP and CADD scores. Performance of MMSplice and SPANR alone as well as their performance with phyloP and CADD scores are on the right (orange)

indicating that MMSplice alone captured most of the sequence information captured by all other models.

We were then interested in delineating the added value of MMSplice per gene region. To this end, we grouped the variants based on their position yielding to (1) 832 exonic variants from the acceptor site region, (2) 1902 exonic variants from the donor site region, (3) 3575 intronic variants from the donor site region, and (4) 4393 intronic variants from the acceptor site region. On exonic variants, we further benchmarked against MutPred Splice [15] which predicts pathogenicity of exonic variants. Among the models that do not integrate phyloP and CADD features, MMSplice was the best in the acceptor site region (auROC = 0.602 for the exonic variants and auROC = 0.970 for the intronic variants, Additional file 1: Figure S8A,D). On the donor site region, MMSplice and the ensemble of MaxEntScan, HAL, and LaBranchoR were both the best models (auROC = 0.651 for the exonic variants and auROC = 0.977 for the intronic variants, Additional file 1: Figure S8B,C). MMSplice performed better than MutPred Splice on both exonic regions (MMSplice: auROC = 0.602, 0.651, MutPred: auROC = 0.594, 0.642, Additional file 1: Figure S8A,B), even though MutPred integrates conservation features [15]. Furthermore, the ensemble model that included MMSplice with phyloP and CADD features had a similar performance than the best ensemble model in all four regions (Additional file 1: Figure S9, auROC = 0.893, 0.917, 0.981, 0.982 versus auROC = 0.894, 0.919, 0.988, 0.985). Notably, phyloP and CADD had good performance on exonic variants (auROC = 0.874, 0.869), but close to random in the evaluated intronic variants (auROC = 0.505, 0.483). In contrast, all other splicing models without phyloP and CADD were performing better at intronic variants but much worse at exonic variants, likely because many pathogenic exonic variants do not affect splicing but have a functional impact on the protein.

Recently, SPiCE [16] has been proposed as a method to predict the probability of a splice site variant affecting splicing. SPiCE is a logistic regression model trained from 142 manually collected and experimentally tested variants. We thus benchmarked against SPiCE with 12,625 ClinVar variants (2312 indels) that SPiCE was able to score (it failed to score variants from sex chromosomes, "Methods" section). MMSplice (auROC = 0.911) outperformed SPiCE (auROC = 0.756, P = 0.001, bootstrap). Moreover, this higher performance of the MMSplice model also held when we fine-tuned the logistic regression model of SPiCE on the ClinVar training dataset (auROC = 0.760, $P = 0.001$, bootstrap, Additional file 1: Figure S10).

Altogether, these results show that MMSplice not only improves the predictions of the effects of variants on biophysical splicing quantities, but also helped improving variant pathogenicity predictions.

Discussion

We have introduced MMSplice, a modular framework to predict the effects of genetic variants on splicing quantities. We did so by training individual modules scoring exon, intron, and splice sites. Models built by integrating these modules showed improved performance against state-of-the-art models on predicting the effects of genetic variants on Ψ, Ψ_3, Ψ_5, splicing efficiency, and pathogenicity. The MMSplice software is open source and can be directly applied on VCF files and handles single nucleotide variants and indels. Like other recent models [17–19], MMSplice score variants beyond the narrow region close to splice sites that is for now suggested by clinical guidelines [42]. We also implemented a VEP [40] plugin that wraps the python implementation. These features should facilitate the integration of MMSplice into bioinformatics pipelines at use in genetic diagnostic centers and may help in improving the discovery of pathogenic variants.

MMSplice leverages the modularity of neural networks and deep learning frameworks. MMSplice is implemented using the deep learning python library Keras [43]. All MMSplice modules and models are shared in the model repository Kipoi [31], which should allow other computational biologists to improve individual modules or to flexibly include modules into their own models. We hope this modular approach will help the community to coordinate efforts and continuously and effectively built better variant effect prediction models for splicing.

Variations across the reference genome or across natural genetic variations in the population may be limited by evolutionary confounding factors, limiting the model's ability to make predictions about rare genetic variants. Experimental perturbation assays are useful because they circumvent these confounding factors. Here, we have leveraged a massively parallel reporter assay [18] to build individual modules. Also, models predicting Ψ and splicing efficiencies were trained on large-scale perturbation datasets (Vex-seq [29] and MaPSy). We note however that MMSplice was not entirely fitted on perturbation assays: The donor site and the acceptor site modules have been trained on the GENCODE annotation, which is observational. Our models outperformed models based on the reference genome and natural variations and was only matched by models based on perturbation assays (HAL for $\Delta\Psi_3$ and the SMS score for in vivo splicing efficiency changes). Nonetheless, one should remain cautious about how predictive rules learned from specific perturbation assays generalize to more general contexts. For instance, the Rosenberg MPRA dataset probed only two 25-nt-long sequences for a very specific construct. Hence, it is important to validate models on further independent perturbation data.

Our models have some limitations. First, splicing is known to be tissue-specific [44, 45], while our models are not. Nevertheless, our models can serve as a good foundation to train tissue-specific models. Second, RNA stability also plays a role in determining the ratio of different isoforms [29]. Models predicting RNA stability from sequence, as we recently developed for the *Saccharomyces cerevisiae* genome [46] could be integrated as further modules. Third, our exon and intron modules are developed from minigene studies, and the performance evaluation on predicting $\Delta\Psi$ and splicing efficiency changes are also done with minigene experiment data. However, chromatin states are known to have a significant role in splicing regulation [47]. Hence, variant effect prediction for endogenous genes could possibly benefit from models taking chromatin states into account. Fourth, our exon and intron modules have only one convolutional layer, which is not enough to learn complex interaction effects of splicing regulatory elements [48]. We have explored using multiple convolutional layers, but the performance on the Vex-seq training data was similar (data not shown). We therefore chose the simpler architecture. The limitation may come from the training data, as the perturbation assay we are training from has 2.5 million random sequences of 25 nucleotides. This library is maybe not deep enough to probe motif interactions, relative distances, and orientations. Non-random libraries that probe the grammar of discovered motifs could be designed in the future and help studying motif interactions. Fifth, MMSplice can technically score variants arbitrarily deep into introns. However, as the training data of MMSplice did not cover deep intronic variants, we suggest to only consider up to 100 nt into introns, as we did here. Further models, such as SPANR which is able to score variants up to 300 nt into the intron, would need to be developed to cover deep intronic variants.

Like former splicing predictors [17–19, 21–23], the goal of MMSplice is to predict quantitatively physical measures of splicing and not variant pathogenicity. Whether affecting splicing at given locus leads to disease heavily depends on the function of the gene and of the splice isoforms. Moreover, existing pathogenicity annotations, such as from the ClinVar database, are probably biased toward tools such as MaxEntScan that are popular and have been in use for a long time. Nonetheless, our results indicate that MMSplice predictions could be potent predictive features for pathogenic variant scores such as S-CAP [49] or CADD [35].

Methods

Donor and acceptor modules

The donor and the acceptor modules were trained using the same approach. A classifier was trained to classify positive donor sites (annotated) against negative ones (random, see below) and the same for the acceptor sites. The classifiers predicted scores can be interpreted as predicted strength of the splice sites.

Donor and acceptor module training data

For the positive set, we took all annotated splice junctions based on the GENCODE annotation version 24 (GRCh38.p5). For the donor module, a sequence window with 5 nt in the exon and 13 nt in the intron around the donor sites was selected. For the acceptor module, the region around the acceptor sites spanning from 50 nt in the intron to 3 nt in the exon was selected in order to cover most branch points. In total, there were 273,661 unique annotated donor sites and 271,405 unique annotated acceptor sites. This set of splice sites was considered as the positive set. In particular, not only sites with the canonical splicing dinucleotides GT and AG for donor and acceptor sites, respectively, were selected, but also sites with non-canonical splicing dinucleotides were included as positive splice sites.

The negative set consisted of genomic sequences selected within the genes that contributed to positive splice sites, in order to approximately match the sequence context of the positive set. Negative splice sites were selected randomly around but not overlapping the positive splice sites. To increase the robustness of the classifiers, around 50% of the negative splice sites were selected to have the canonical splicing dinucleotides. In total, 410,111 negative donor sites and 406,841 negative acceptor sites were selected. During model training, we split 80% of the data for training and 20% of the data for validation. The best performing model on the validation set was used for variant effect prediction.

Donor and acceptor module architecture

Neural network models were trained to score splice sites from one-hot-encoded input sequence. The donor model was a multilayer perceptron with two hidden layers with Rectified Linear Unit (ReLU) activations and a sigmoid output (Additional file 1: Figure S1A). The hidden layers were trained with a dropout rate [50] of 0.2 and batch normalization [51]. We chose a multilayer perceptron over a convolutional neural network because of the short input sequence of the donor model. The acceptor model was a convolutional neural network with two consecutive convolution layers, with 32 15 × 1 convolution followed by 32 1 × 1 convolution (Additional file 1: Figure S1B). The second convolutional layer was trained with a dropout rate of 0.2 and batch normalization. For these models, we found the number of layers and the number of neurons in each layer by hyperparameter optimization.

Exon module

Exon module training data

The exonic random sequences from the MPRA experiment by Rosenberg et al. [18] were used to train the exon scoring module. This MPRA experiment contains two libraries, one for alternative 5′ splicing and one for alternative 3′ splicing. The alternative 5′ splicing library has 265,137 random constructs while the alternative 3′ splicing library has 2,211,789. Each random construct has a 25-nt random sequence in the alternative exon and a 25-nt random sequence in the adjacent intron. Ψ_5 and Ψ_3 of different isoforms were quantified by RNA-Seq for each random construct [18]. Here, 80% of the data was used for model training and the remaining were used for validation. The best performing model on the validation set was used for variant effect prediction.

Exon module architecture

Rosenberg et al. [18] showed that the effects of splicing-related features in alternative exons are strongly correlated with each other across the two MPRA libraries, reflecting that similar exonic regulatory elements are involved for both donor and acceptor splicing. We thus decided to train exon scoring module from the two MPRA libraries by sharing low-level convolution layers (128 15 × 1 filters, Additional file 1: Figure S2). The inputs of the network were one-hot-encoded 25-nt random sequences. The output labels were Ψ_5, respectively Ψ_3, for the alternative exon. After training, the exon modules for each library were separated by transferring the corresponding weights to two separated modules with convolution layer with ReLU non-linearity followed by a global average pooling and a fully connected layer. We have used a global pooling after the convolution layer allowing to take exons of any length as input. This ended up with two exon scoring modules, one for alternative 5′ end (exon 5′ module) and one for alternative 3′ end (exon 3′ module).

Intron module

Intron modules were trained in the same way as the exon modules (Additional file 1: Figure S2) by using intronic random sequences from the MPRA experiment as inputs, except that we used 256 15 × 1 convolution filters, because intronic splicing regulatory elements from the donor side and the acceptor side are less similar [18]. This ended up with a module to score intron on the donor side (intron 5′ module) and a module to score intron on the acceptor side (intron 3′ module).

Training procedure for the modules

All neural network models for the six modules were trained with binary cross-entropy loss (Eq. 1) and Adam optimizer [52]. We implemented and trained these models with the deep learning python library Keras [43]. Bayesian optimization implemented in hyperopt package [53] was used for hyper-parameter optimization together with the kopt package (github.com/avsecz/kopt). Every trial, a different hyper-parameter combination is proposed by the Bayesian optimizer, with which a model is trained on the training set, its performance is monitored by the validation loss. The model that had the smallest validation loss was selected.

$$\text{Loss}_i = -(\psi_i \log \hat{\psi}_i + (1 - \psi_i) \log(1 - \hat{\psi}_i)) \qquad (1)$$

Variant effect prediction models

Variant processing

Variants are considered to affect the splicing of an exon if it is exonic or if it is intronic and at a distance less than L_a from an acceptor site or less than L_d from a donor site. The distances L_a and L_d were set to 100 nt in this study but can be flexibly set for MMSplice. MMSplice provides code to generate reference and alternative sequences from a variant-exon pair by substituting variants into the reference genome. Variant-exon pairs can be directly provided to MMSplice. This is the case for the perturbation assay data Vex-seq, MFASS, and MaPSy. MMSplice can also generate variant-exon pairs from given VCF files (Fig. 5a). For insertions, and for deletions that are not overlapping a splice site, the alternative sequence is obtained by inserting or deleting sequence correspondingly. For deletions overlapping a splice site, the alternative sequence is obtained by deleting the sequence and the new splice site is defined as the boundaries of the deletion. In all cases, the returned alternative sequence always have the same structure as the reference sequence, with an exon of flexible length flanked by L_a and L_d intronic nucleotides. Each variant is processed independently from the other variants, i.e., each mutated sequence contains only one variant (Fig. 5a). If a variant can affect multiple target (i.e., sites or exons), the MMSplice models return predictions for every possible target (Fig. 5a).

Variant effect prediction for Ψ

Strand information of all Vex-seq assayed exons were first determined by overlapping them with Ensembl GRCh37 annotation release 75. Reference sequences were extracted by taking the whole exon and 100 nt flanking intronic sequence. Variant sequences were retrieved as described in the "Variant processing" in the "Methods" section, whereby variant-exon pairs were provided by the experimental design.

We modeled the differential effect on Ψ in the logistic scale with the following linear model:

$$\begin{aligned}\Delta\text{logit}(\Psi) &= \text{logit}(\Psi_{\text{alt}}) - \text{logit}(\Psi_{\text{ref}})\\ &= \beta_0 + \beta_1 \Delta S_{3' \text{ intron}}\\ &+ \beta_2 \Delta S_{\text{acceptor}} + \beta_3 \Delta S_{\text{exon}}\\ &+ \beta_4 \Delta S_{\text{donor}} + \beta_5 \Delta S_{5' \text{ intron}}\\ &+ \beta_6 \mathbb{1}(\text{Exon overlap splice site modules}) \Delta S_{\text{exon}}\\ &+ \beta_7 \mathbb{1}(5' \text{ intron overlap donor module}) \Delta S_{5' \text{ intron}}\\ &+ \beta_8 \mathbb{1}(3' \text{ intron overlap acceptor module}) \Delta S_{3' \text{ intron}}\\ &+ \epsilon \end{aligned} \tag{2}$$

where

$$\Delta S = S_{\text{alt}} - S_{\text{ref}} \tag{3}$$

for all five modules, $\mathbb{1}(\cdot)$ is the indicator function, ϵ is the error term, the suffix *alt* denotes the alternate allele, and the suffix *ref* denotes the reference allele. This model has nine parameters: one intercept, one coefficient for each of the five modules, and interaction terms for regions that were scored by two modules (Fig. 1). The latter interaction terms were useful to not double count the effect of variants scored by multiple modules. These nine parameters were the only parameters that were trained from the Vex-seq data. The parameters of the modules stayed fixed. To fit this linear model, we used Huber loss [54] instead of ordinary least squares loss to make the fitting more robust to outliers.

The model predicts $\Delta\text{logit}\Psi$ for the variant. We transform this to $\Delta\Psi$ with a given reference Ψ as follows:

$$\begin{aligned}\hat{\Psi}_{\text{alt}} &= \sigma(\Delta\text{logit}\Psi + \text{logit}(\Psi_{\text{ref}}))\\ \Delta\hat{\Psi} &= \hat{\Psi}_{\text{alt}} - \Psi_{\text{ref}}\end{aligned} \tag{4}$$

where

$$\sigma(x) = \frac{1}{1+e^{-x}} \tag{5}$$

$$\text{logit}(x) = \log \frac{x}{1-x} \tag{6}$$

To prevent infinite values in cases $\Psi_{\text{ref}} = 0$ or $\Psi_{\text{ref}} = 1$, Ψ_{ref} values were clipped to the interval $[10^{-5}, 1-10^{-5}]$.

HAL model is provided by the authors. A scaling factor required by HAL was trained on the Vex-seq training data using code provided by the authors [18]. The SPANR precomputed scores (which are called SPIDEX), were obtained from http://www.openbioinformatics.org/annovar/spidex_download_form.php.

Performance on the MFASS dataset

MMSplice was applied the same way as for Vex-seq, except that module combining weights were learned from the Vex-seq training data, with MFASS data kept entirely unseen. SDVs were classified based on the predicted $\Delta\Psi$ for a variant. Area under the precision-recall curve (auPR) were calculated with `trapz` function from R package `pracma`.

Variant effect prediction for Ψ_3 and Ψ_5

The Genotype-Tissue Expression (GTEx) [37] RNAseq data (V6) was used to extract variant effect on Ψ_3 and Ψ_5. Variants $[-3, +6]$ nt around alternative donors of alternative 5′ splicing events and variants $[-20, +3]$ nt around alternative acceptors for alternative 5′ splicing events were considered. The skin (not sun exposed) samples and the brain samples with matched whole genome sequence data available were processed. Ψ_5 and Ψ_3 were calculated with MISO [20] for each sample. Altogether, 1057 brain samples and 211 skin samples could be successfully processed with MISO. Ψ_3 and Ψ_5 for homozygous reference variant, heterozygous variants, and homozygous alternative variants were calculated by taking the average across samples with the same genotype, excluding samples from individuals with more than one variants within 300 nt around the competing splice sites.

We predicted differences in Ψ_5 as follows. We considered only donor sites with two alternative acceptor sites. We extracted the relevant sequences for the corresponding two alternative exons and apply the model of Eq. (2) which was fitted on Vex-seq training data. This returned a $\Delta\text{logit}(\Psi)$ for each alternative exon, denoted ΔS_1 and ΔS_2, from which we calculate the predicted alternative Ψ_5 as follows:

$$\Psi_{5_{\text{alt}}} = \sigma\left(\Delta\text{logit}(\Psi_5) + \text{logit}\left(\Psi_{5_{\text{ref}}}\right)\right) \tag{7}$$

where we model the $\Delta\text{logit}(\Psi_5)$ considering the influence of variant on both alternative exon as follows (derivations provided in supplements):

$$\Delta\text{logit}(\Psi_5) = \Delta S_1 - \Delta S_2 \tag{8}$$

The above computation applies to individual alleles. To handle heterozygous variants, we assumed expression from both alleles are equal. This led to the following predictions for homozygous and heterozygous variants:

$$\begin{aligned}\Delta\Psi_{5_{\text{homo}}} &= \Psi_{5_{\text{alt}}} - \Psi_{5_{\text{ref}}}\\ \Delta\Psi_{5_{\text{hetero}}} &= \left(\Psi_{5_{\text{ref}}} + \Psi_{5_{\text{alt}}}\right)/2 - \Psi_{5_{\text{ref}}}\end{aligned} \tag{9}$$

Analagous calculations were made to predict differences in Ψ_3.

Pre-trained COSSMO model [19] was obtained from the author website (http://cossmo.genes.toronto.edu/). The predicted $\Delta\Psi_5$ (or $\Delta\Psi_3$) values of COSSMO were calculated by taking the difference between the predicted Ψ_5 (or Ψ_3) from alternative sequence processed by MMSplice and reference sequence.

Splicing efficiency dataset (MaPSy data)

The splicing efficiency assay was performed for 5,761 disease causing exonic nonsynonymous variants both in vivo in HEK293 cells and in vitro in HeLa-S3 nuclear extract as previously described [27]. Here, the exons were derived from human exons and were reduced in size to be shorter than 100 nt long by small deletions applied to both the reference and the alternative version of the sequence. This way, the wild-type and the mutated alleles differed from each other by a single point mutation and the wild-type allele differed from a human exon by the small deletions. The deletions were centered at the midpoint between the variant and the furthest exon boundary. The sequences of each substrate are listed in Additional file 2: Table S1 and also described further on the CAGI website (https://genomeinterpretation.org/content/MaPSy).

Overall, 4964 of the variants were in the training set and 797 were in the test set. The amount of spliced transcripts and unspliced transcripts for each construct with reference allele or alternative allele were determined by RNA-Seq. The effect of mutation on splicing efficiency for a specific reporter sequence was quantified by the allelic log-ratio, which is defined as:

$$\log_2\left(\frac{m_o/m_i}{w_o/w_i}\right) \tag{10}$$

where m_o is the mutant spliced RNA read count, m_i is the mutant input (unspliced) RNA read count, w_o is the wild-type spliced RNA read count, and w_i is the wild-type input RNA read count. Transcripts with exon-skipped or misspliced are ignored.

Variant effect prediction for splicing efficiency (MaPSy data)

We fitted a model to predict differential splicing efficiency on the training set with a linear regression with a Huber loss as defined by Eq. 2, except that the response variable is the allelic log-ratio (Eq. 10) instead of $\Delta\text{logit}(\Psi)$. We used the exon 5′ module for the splicing efficiency model. Performance on MaPSy data was reported on the held-out test set.

SMS scores was applied to wild-type and mutant sequence by summing up all 7-mer scores as described by Ke et al. [28]. The predicted allelic log-ratio is the SMS score difference between mutant and wild-type sequence.

Variant pathogenicity prediction

Processed ClinVar variants (version 20180429 for GRCh37) around splice sites were obtained from Avsec et al. [31]. Specifically, single-nucleotide variants [− 40, 10] nt around the splicing acceptor or [− 10, 10] nt around the splice donor of a protein-coding genes (Ensembl GRCh37 v75 annotation) were selected. Variants causing a premature stop codon were discarded. After the filtering, the 6310 pathogenic variants constituted the positive set and the 4405 benign variants constituted the negative set. The CADD [35] scores and the phyloP [55] scores were obtained through VEP [40]. MMSplice $\Delta Score$ predictions of the five modules as well as indicator variables of the overlapping region were assembled with a logistic regression model to classify pathogenicity. Performance was assessed by 10-fold cross-validation (Additional file 1: Supplementary Methods).

To compare MMSplice with SPiCE [16], we restricted to the regions that SPiCE scores, i.e., [− 12, 2] nt around the acceptor or [− 3, 8] nt around the donor of protein-coding genes. Variants causing a premature stop codon were discarded. SPiCE was trained to predict the probability of a variant to affect splicing (manually defined by experimental observations). To apply it for pathogenicity prediction, the logistic regression model of SPiCE was refitted with ClinVar pathogenicity as response variable. MMSplice model was applied as described above without conservation features. Models were compared under 10-fold cross-validation.

Bootstrapping for P value and confidence interval estimation

Significance levels when comparing the performance of two models were estimated with the basic bootstrap [56]. Denoting t_1 the performance metric (Pearson correlation, auPRC, or auROC) of MMSplice and t_2 the performance metric of a competing model, we considered the difference $d = t_1 - t_2$. We sampled with replacement the test data $B = 999$ times and each time i computed the bootstrapped metric difference d_i^*. The one-sided P value was approximated as [56].

$$P = \frac{1 + \#\{d_i^* \leq 0; i = 1 ... B\}}{B + 1} \tag{11}$$

We estimated confidence intervals of Pearson correlations and root-mean-square values, using the percentile bootstrap approach. Specifically, we generated 1000 bootstrap datasets of the same size by sampling with replacement. Noting the value of either of the statistics of interest as θ^*, the reported 95% confidence interval is $\left(\theta^*_{0.025}, \theta^*_{0.975}\right)$, where $\theta^*_{0.025}$ and $\left(\theta^*_{0.975}\right)$ are the 2.5 and the 97.5 percentiles, respectively.

Additional files

Additional file 1: Supplementary methods and figures. (PDF 674 kb)
Additional file 2: Table S1: MaPSy splicing efficiency data. (CSV 3 kb)

Acknowledgements
We thank the Critical Assessment of Genome Interpretation (CAGI) organizers, especially Steven E. Brenner, John Moult and Gaia Andreoletti for organizing the CAGI competition. We thank Scott I. Adamson, Brenton R. Graveley for formatting and providing Vex-seq experiment data through CAGI.

Funding
JC was supported by the Competence Network for Technical, Scientific High Performance Computing in Bavaria KONWIHR. ZA and JC were supported by a Deutsche Forschungsgemeinschaft fellowship through the Graduate School of Quantitative Biosciences Munich. This work was supported by NVIDIA hardware grant providing a Titan X GPU card.

Availability of data and materials

MMSplice and its VEP plugin are available at https://github.com/gagneurlab/MMSplice [57] and Kipoi: https://kipoi.org/models/MMSplice [58] under the MIT License. An archival version of MMSplice is available at zenodo: https://doi.org/10.5281/zenodo.2555955 [59]. The analysis code is available at https://github.com/gagneurlab/MMSplice_paper [60].
Vex-seq data is available at https://github.com/scottiadamson/Vex-seq [61]. MFASS data is available at https://github.com/KosuriLab/MFASS [62]. GTEx data is available through Genotypes and Phenotypes (dbGaP) under accession number (phs000424.v6.p1). ClinVar data is available at ftp://ftp.ncbi.nlm.nih.gov/pub/clinvar/vcf_GRCh37/archive_2.0/2018/clinvar_20180429.vcf.gz [63]. MaPSy data is available as Additional files 2 of this manuscript.

Authors' contributions

JC and JG designed the model, with the help of ZA. JC implemented the software and analysed data. TYDN and ZA contributed to developing the modules. JC and JG wrote the manuscript, with the help of ZA, KJC, and WGF. KJC and WGF generated the MaPSy data. MHC wrote the VEP plugin. All authors read and approved the final manuscript.

Ethics approval and consent to participate

Not applicable.

Consent for publication

Not applicable.

Competing interests

The authors declare that they have no competing interests.

Publisher's Note

Springer Nature remains neutral with regard to jurisdictional claims in published maps and institutional affiliations.

Author details

[1]Department of Informatics, Technical University of Munich, Boltzmannstraße, 85748 Garching, Germany. [2]Graduate School of Quantitative Biosciences (QBM), Ludwig-Maximilians-Universität München, München, Germany. [3]Center for Computational Molecular Biology, Brown University, Providence, Rhode Island, USA. [4]Department of Molecular Biology, Cell Biology and Biochemistry, Brown University, Providence, Rhode Island, USA.

Received: 2 October 2018 Accepted: 12 February 2019
Published online: 01 March 2019

References

1. López-Bigas N, Audit B, Ouzounis C, Parra G, Guigó R. Are splicing mutations the most frequent cause of hereditary disease? FEBS Lett. 2005;579(9):1900–3. https://doi.org/10.1016/j.febslet.2005.02.047.
2. Li YI, van de Geijn B, Raj A, Knowles DA, Petti AA, Golan D, Gilad Y, Pritchard JK. RNA splicing is a primary link between genetic variation and disease. Science. 2016;352(6285):600–4. https://doi.org/10.1126/science.aad9417.
3. Wahl MC, Will CL, Lührmann R. The spliceosome: design principles of a dynamic RNP machine. Cell. 2009;136(4):701–18.
4. Wang Z, Burge CB. Splicing regulation: from a parts list of regulatory elements to an integrated splicing code. Rna. 2008;14(5):802–13.
5. Scotti MM, Swanson MS. RNA mis-splicing in disease. Nat Rev Genet. 2015;17(1):19–32. https://doi.org/10.1038/nrg.2015.3.
6. Reese MG, Eeckman FH, Kulp D, Haussler D. Improved splice site detection in Genie. J Comput Biol. 1997;4(3):311–23.
7. Yeo G, Burge CB. Maximum entropy modeling of short sequence motifs with applications to RNA splicing signals. J Comput Biol J Comput Mol Cell Biol. 2004;11(2-3):377–94. https://doi.org/10.1089/1066527041410418.
8. Fairbrother WG, Yeh RF, Sharp PA, Burge CB. Predictive identification of exonic splicing enhancers in human genes. Science. 2002;297(5583):1007–13.
9. Fairbrother WG, Yeo GW, Yeh R, Goldstein P, Mawson M, Sharp PA, Burge CB. RESCUE-ESE identifies candidate exonic splicing enhancers in vertebrate exons. Nucleic Acids Res. 2004;32(Web Server issue):187–90. https://doi.org/10.1093/nar/gkh393.
10. Zhang XHF, Chasin LA. Computational definition of sequence motifs governing constitutive exon splicing. Genes Dev. 2004;18(11):1241–50. https://doi.org/10.1101/gad.1195304.
11. Zhang XH-F, Kangsamaksin T, Chao MS, Banerjee JK, Chasin LA. Exon inclusion is dependent on predictable exonic splicing enhancers. Mol Cell Biol. 2005;25(16):7323–32.
12. Wang Z, Xiao X, Van Nostrand E, Burge CB. General and specific functions of exonic splicing silencers in splicing control. Mol Cell. 2006;23(1):61–70.
13. Ke S, Shang S, Kalachikov SM, Morozova I, Yu L, Russo JJ, Ju J, Chasin LA. Quantitative evaluation of all hexamers as exonic splicing elements. Genome Res. 2011;21(8):1360–74. https://doi.org/10.1101/gr.119628.110.
14. Desmet FO, Hamroun D, Lalande M, Collod-Béroud G, Claustres M, Béroud C. Human Splicing Finder: an online bioinformatics tool to predict splicing signals. Nucleic Acids Res. 2009;37(9):67. https://doi.org/10.1093/nar/gkp215.
15. Mort M, Sterne-Weiler T, Li B, Ball EV, Cooper DN, Radivojac P, Sanford JR, Mooney SD. MutPred Splice: machine learning-based prediction of exonic variants that disrupt splicing. Genome Biol. 2014;15(1):19. https://doi.org/10.1186/gb-2014-15-1-r19.
16. Leman R, Gaildrat P, Gac GL, Ka C, Fichou Y, Audrezet M-P, Caux-Moncoutier V, Caputo SM, Boutry-Kryza N, Léone M, et al. Novel diagnostic tool for prediction of variant spliceogenicity derived from a set of 395 combined in silico/in vitro studies: an international collaborative effort. Nucleic Acids Res. 2018;46(15):7913–23.
17. Xiong HY, Alipanahi B, Lee LJ, Bretschneider H, Merico D, Yuen RKC, Hua Y, Gueroussov S, Najafabadi HS, Hughes TR, Morris Q, Barash Y, Krainer aR, Jojic N, Scherer SW, Blencowe BJ, Frey BJ. The human splicing code reveals new insights into the genetic determinants of disease. Science (80-). 2015;347(6218):1254806. https://doi.org/10.1126/science.1254806.
18. Rosenberg AB, Patwardhan RP, Shendure J, Seelig G. Learning the sequence determinants of alternative splicing from millions of random sequences. Cell. 2015;163(3):698–711. https://doi.org/10.1016/j.cell.2015.09.054.
19. Bretschneider H, Gandhi S, Deshwar AG, Zuberi K, Frey BJ. COSSMO: predicting competitive alternative splice site selection using deep learning. Bioinformatics (Oxford, England). 2018;34(13):429–37. https://doi.org/10.1093/bioinformatics/bty244.
20. Katz Y, Wang ET, Airoldi EM, Burge CB. Analysis and design of RNA sequencing experiments for identifying isoform regulation. Nat Methods. 2010;7(12):1009–15. https://doi.org/10.1038/nmeth.1528. 9605103.
21. Barash Y, Calarco JA, Gao W, Pan Q, Wang X, Shai O, Blencowe BJ, Frey BJ. Deciphering the splicing code. Nature. 2010;465(7294):53–9. https://doi.org/10.1038/nature09000.
22. Xiong HY, Barash Y, Frey BJ. Bayesian prediction of tissue-regulated splicing using RNA sequence and cellular context. Bioinformatics. 2011;27(18):2554–62. https://doi.org/10.1093/bioinformatics/btr444.
23. Jha A, Gazzara MR, Barash Y. Integrative deep models for alternative splicing. Bioinformatics. 2017;33(14):274–82. https://doi.org/10.1093/bioinformatics/btx268.
24. Pervouchine DD, Knowles DG, Guigó R. Intron-centric estimation of alternative splicing from rna-seq data. Bioinformatics. 2012;29(2):273–4.
25. Park E, Pan Z, Zhang Z, Lin L, Xing Y. The expanding landscape of alternative splicing variation in human populations. Am J Hum Genet. 2018;102(1):11–26. https://doi.org/10.1016/j.ajhg.2017.11.002.
26. Vaquero-Garcia J, Barrera A, Gazzara MR, González-Vallinas J, Lahens NF, Hogenesch JB, Lynch KW, Barash Y. A new view of transcriptome complexity and regulation through the lens of local splicing variations. eLife. 2016;5:11752. https://doi.org/10.7554/eLife.11752. arXiv:1011.1669v3.
27. Soemedi R, Cygan KJ, Rhine CL, Wang J, Bulacan C, Yang J, Bayrak-Toydemir P, McDonald J, Fairbrother WG. Pathogenic variants that alter protein code often disrupt splicing. Nat Genet. 2017;49(6):848–55. https://doi.org/10.1038/ng.3837.
28. Ke S, Anquetil V, Zamalloa JR, Maity A, Yang A, Arias MA, Kalachikov S, Russo JJ, Ju J, Chasin LA. Saturation mutagenesis reveals manifold determinants of exon definition. Genome Res. 2018;28(1):11–24.
29. Adamson SI, Zhan L, Graveley BR. Vex-seq: high-throughput identification of the impact of genetic variation on pre-mRNA splicing efficiency. Genome Biol. 2018;19(1):71.
30. Landrum MJ, Lee JM, Riley GR, Jang W, Rubinstein WS, Church DM, Maglott DR. Clinvar: public archive of relationships among sequence variation and human phenotype. Nucleic Acids Res. 2013;42(D1):980–5.

31. Avsec Z, Kreuzhuber R, Israeli J, Xu N, Cheng J, Shrikumar A, Banerjee A, Kim DS, Urban L, Kundaje A, Stegle O, Gagneur J. Kipoi: accelerating the community exchange and reuse of predictive models for genomics. bioRxiv. 2018. https://doi.org/10.1101/375345. https://www.biorxiv.org/content/early/2018/07/24/375345.full.pdf.
32. Lek M, Karczewski KJ, Minikel EV, Samocha KE, Banks E, Fennell T, O'Donnell-Luria AH, Ware JS, Hill AJ, Cummings BB, Tukiainen T, Birnbaum DP, Kosmicki JA, Duncan LE, Estrada K, Zhao F, Zou J, Pierce-Hoffman E, Berghout J, Cooper DN, Deflaux N, DePristo M, Do R, Flannick J, Fromer M, Gauthier L, Goldstein J, Gupta N, Howrigan D, Kiezun A, Kurki MI, Moonshine AL, Natarajan P, Orozco L, Peloso GM, Poplin R, Rivas MA, Ruano-Rubio V, Rose SA, Ruderfer DM, Shakir K, Stenson PD, Stevens C, Thomas BP, Tiao G, Tusie-Luna MT, Weisburd B, Won HH, Yu D, Altshuler DM, Ardissino D, Boehnke M, Danesh J, Donnelly S, Elosua R, Florez JC, Gabriel SB, Getz G, Glatt SJ, Hultman CM, Kathiresan S, Laakso M, McCarroll S, McCarthy MI, McGovern D, McPherson R, Neale BM, Palotie A, Purcell SM, Saleheen D, Scharf JM, Sklar P, Sullivan PF, Tuomilehto J, Tsuang MT, Watkins HC, Wilson JG, Daly MJ, MacArthur DG. Analysis of protein-coding genetic variation in 60,706 humans. Nature. 2016;536(7616):285–91. https://doi.org/10.1038/nature19057. 030338.
33. Hoskins RA, Repo S, Barsky D, Andreoletti G, Moult J, Brenner SE. Reports from CAGI: the critical assessment of genome interpretation. Hum Mutat. 2017;38(9):1039–41.
34. Cheung R, Insigne KD, Yao D, Burghard CP, Wang J, Hsiao Y-HE, Jones EM, Goodman DB, Xiao X, Kosuri S. A multiplexed assay for exon recognition reveals that an unappreciated fraction of rare genetic variants cause large-effect splicing disruptions. Mol Cell. 2019;73(1):183–94.
35. Kircher M, Witten DM, Jain P, O'roak BJ, Cooper GM, Shendure J. A general framework for estimating the relative pathogenicity of human genetic variants. Nat Genet. 2014;46(3):310.
36. Siepel A, Bejerano G, Pedersen JS, Hinrichs AS, Hou M, Rosenbloom K, Clawson H, Spieth J, Hillier LW, Richards S, et al. Evolutionarily conserved elements in vertebrate, insect, worm, and yeast genomes. Genome Res. 2005;15(8):1034–50.
37. Lonsdale J, Thomas J, Salvatore M, Phillips R, Lo E, Shad S, Hasz R, Walters G, Garcia F, Young N, et al. The genotype-tissue expression (gtex) project. Nat Genet. 2013;45(6):580.
38. Warf MB, Berglund JA. Role of RNA structure in regulating pre-mRNA splicing. Trends Biochem Sci. 2010;35(3):169–78. https://doi.org/10.1016/j.tibs.2009.10.004.
39. Danecek P, Auton A, Abecasis G, Albers CA, Banks E, DePristo MA, Handsaker RE, Lunter G, Marth GT, Sherry ST, McVean G, Durbin R. The variant call format and VCFtools. Bioinformatics. 2011;27(15):2156–8. https://doi.org/10.1093/bioinformatics/btr330. NIHMS150003.
40. McLaren W, Gil L, Hunt SE, Riat HS, Ritchie GRS, Thormann A, Flicek P, Cunningham F. The Ensembl variant effect predictor. Genome Biol. 2016;17(1):. https://doi.org/10.1186/s13059-016-0974-4.
41. Paggi JM, Bejerano G. A sequence-based, deep learning model accurately predicts RNA splicing branchpoints. RNA. 2018;24(12):1647–58. https://doi.org/10.1261/rna.066290.118.
42. Richards S, Aziz N, Bale S, Bick D, Das S, Gastier-Foster J, Grody WW, Hegde M, Lyon E, Spector E, Voelkerding K, Rehm HL. Standards and guidelines for the interpretation of sequence variants: a joint consensus recommendation of the American College of Medical Genetics and Genomics and the Association for Molecular Pathology. Genet Med. 2015;17(5):405–24. https://doi.org/10.1038/gim.2015.30. 15334406.
43. Chollet F, et al. Keras. 2015. https://keras.io, version: 2.2.4.
44. Consortium G, et al. The Genotype-Tissue Expression (GTEx) pilot analysis: multitissue gene regulation in humans. Science. 2015;348(6235):648–0.
45. Yeo G, Holste D, Kreiman G, Burge CB. Variation in alternative splicing across human tissues. Genome Biol. 2004;5(10):74. https://doi.org/10.1186/gb-2004-5-10-r74.
46. Cheng J, Maier KC, Avsec Ž, Rus P, Gagneur J. Cis -regulatory elements explain most of the mRNA stability variation across genes in yeast. RNA. 2017;23(11):1648–59. https://doi.org/10.1261/rna.062224.117.
47. Kolasinska-Zwierz P, Down T, Latorre I, Liu T, Liu XS, Ahringer J. Differential chromatin marking of introns and expressed exons by H3K36me3. Nat Genet. 2009;41(3):376–81. https://doi.org/10.1038/ng.322.
48. Han K, Yeo G, An P, Burge CB, Grabowski PJ. A combinatorial code for splicing silencing: UAGG and GGGG motifs. PLoS Biol. 2005;3(5):0843–60. https://doi.org/10.1371/journal.pbio.0030158.
49. Jagadeesh KA, Paggi JM, Ye JS, Stenson PD, Cooper DN, Bernstein JA, Bejerano G. S-CAP extends clinical-grade pathogenicity prediction to genetic variants that affect RNA splicing. bioRxiv. 2018343749. https://doi.org/10.1101/343749.
50. Srivastava N, Hinton G, Krizhevsky A, Sutskever I, Salakhutdinov R. Dropout: a simple way to prevent neural networks from overfitting. J Mach Learn Res. 2014;15(1):1929–58.
51. Ioffe S, Szegedy C. Batch Normalization: accelerating deep network training by reducing internal covariate shift. arXiv. 2015. 1502.03167.
52. Kingma D, Ba J. Adam: a method for stochastic optimization. 2014. arXiv preprint arXiv:1412.6980.
53. Bergstra J, Komer B, Eliasmith C, Yamins D, Cox DD. Hyperopt: a python library for model selection and hyperparameter optimization. Comput Sci Discov. 2015;8(1):014008.
54. Huber PJ. Robust estimation of a location parameter. Ann Math Stat. 1964;35(1):73–101. https://doi.org/10.1214/aoms/1177703732. arXiv:1111.1308v3.
55. Pollard KS, Hubisz MJ, Rosenbloom KR, Siepel A. Detection of nonneutral substitution rates on mammalian phylogenies. Genome Res. 2010;20(1):110–21. https://doi.org/10.1101/gr.097857.109.
56. Davison AC, Hinkley DV. Bootstrap methods and their applications, vol. 1. Cambridge University Press; 1997.
57. Cheng J, Çelik MH. MMSplice : modular modeling improves the predictions of genetic variant effects on splicing. GitHub. https://github.com/gagneurlab/MMSplice.
58. Cheng J, Çelik MH, Avsec Z. MMSplice : modular modeling improves the predictions of genetic variant effects on splicing. GitHub. https://github.com/kipoi/models/tree/master/MMSplice.
59. Cheng J. MMSplice : modular modeling improves the predictions of genetic variant effects on splicing. Zenodo. https://doi.org/10.5281/zenodo.2555955.
60. Cheng J. MMSplice : modular modeling improves the predictions of genetic variant effects on splicing. GitHub. https://github.com/gagneurlab/MMSplice_paper.
61. Adamson SI. Vex-seq: high-throughput identification of the impact of genetic variation on pre-mRNA splicing efficiency. GitHub. https://github.com/scottiadamson/Vex-seq. Accessed 16 Feb 2018.
62. Insigne KD. A multiplexed assay for exon recognition reveals that an unappreciated fraction of rare genetic variants cause large-effect splicing disruptions. GitHub. https://github.com/KosuriLab/MFASS. Accessed 15 Mar 2018.
63. Landrum MJ, Lee JM, Riley GR, Jang W, Rubinstein WS, Church DM, Maglott DR. ClinVar: public archive of relationships among sequence variation and human phenotype. ClinVar. ftp://ftp.ncbi.nlm.nih.gov/pub/clinvar/vcf_GRCh37/archive_2.0/2018/clinvar_20180429.vcf.gz. Accessed May 2018.