Evaluating Predictions of Splice Site Usage Probabilities

Jannis Baum
Hasso Plattner Institute, Digital Engineering Faculty
University of Potsdam
Potsdam, Germany
Email: Jannis.Baum@student.hpi.uni-potsdam.de

***Abstract*—Alternative RNA splicing is a post-transcriptional process in eukaryotic cells essential for physiological function that can also contribute to or cause disease. While splicing can be observed experimentally with RNA sequencing, this comes with limitations that in-silico tools aim to overcome by predicting it from the DNA sequence. These tools are typically used to classify effects of single variants on nearby splice sites, and previous work on evaluating the tools has focused on this use case. State of the art deep learning-based tools are however trained to predict per-base splice site usage probabilities (SSUPs) for a given sequence. In this work, I extend the evaluation of two popular deep learning-based tools, SpliceAI and Pangolin, by directly analyzing their predicted SSUPs. For this, I introduce a pipeline to compute per-base SSUPs from genetic variants with the two tools, and to evaluate and compare them to RNA-seq results based on visualizations and a suitable error metric. I execute the pipeline on the example of 100 breast cancer patients and the gene BRCA1. The results show that both prediction tools are in general accordance with the RNA-seq data, and that the choice of the more suitable tool should consider how important the avoidance of underestimation is for the given use case. With this work, I contribute to the in-depth evaluation of two popular deep learning-based splice site prediction tools, and offer a new perspective on what to consider when selecting a tool for a use case.**

***Index Terms*—RNA splicing, splice site prediction, model analysis**

I. Introduction

RNA splicing is a post-transcriptional process in eukaryotic cells where introns, the non-coding regions, are removed from pre-mRNA, and exons, the coding sequences, are joined together. This process is part of the generation of mature mRNA, which is then translated into proteins. Additionally, RNA can undergo alternative splicing, where different combinations of splice sites, called splice junctions, are used to join different exons together, forming different transcripts. This allows a single gene to be expressed as multiple functional protein isoforms, but can also result in dysfunctional forms and contribute to disease [1].

Experimentally, the usage of splice sites in cells can be observed by sequencing mRNA (RNA-seq) [2] but this comes with several limitations. First, the cells in question have to be available for RNA-seq, which is not the case if a biopsy of the respective tissue is not feasible [3] or if the goal is to test a hypothesized variant without having cells with the corresponding genome. Second, RNA-seq may fail to acquire sufficient data if the investigated gene has low expression levels or if alternative splicing leads to unstable RNA, making it difficult to detect [4]. Third, the typical short read lengths of RNA-seq pose a significant challenge to reliably identifying detected isoforms [2].

In an effort to overcome these limitations, various in-silico tools have been developed that predict splice sites and their usage levels based on the DNA sequence. These tools generally aim to predict the effects of a single given variant on splice sites within a given distance, and are evaluated with classification metrics that compare the predicted effects to effects that were experimentally observed with RNA-seq [4].

While some tools restrict their functionality to these classification outputs for single variants, the two tools that have been found to have highest performance [4], [5], SpliceAI [3] and Pangolin [5], both deep learning-based models, internally predict splice site usage probabilities (SSUPs) for every single base in the given input DNA sequence. With this work, I extend the evaluation of SpliceAI and Pangolin by directly analyzing their predicted per-base SSUPs, rather than just their classification outputs. By comparing these predicted probabilities against experimental RNA-seq data, this approach provides a more granular assessment of the tools in terms of their performance and offers deeper insights into their applicability. In this report, I apply and evaluate the presented approach on the BRCA1 gene in breast cancer patients, but the approach is generally applicable and can be easily adapted to other genes and use cases. The implementation of the approach is publicly available through a GitHub repository [6].

In the following, I will introduce related work on splice site prediction tools, existing evaluation of them, and applicable data sources in section II. I will then explain my methods in section III, present the results in section IV, and discuss and interpret them in section V. Finally, I will conclude the report in section VI while also looking into future work on the topic.

II. Related Work

Previous work related to this report comes (1) from the area of splice site prediction, and (2) from medical research about splicing and its effects as well as collection of suitable data. For the latter, I specifically focus on splicing in relation to cancer.

A. Splice site prediction

Splice site prediction has evolved significantly since its first appearance in the 1980s, when the first models relied on scoring sequences based on their alignment to canonical sequences associated with splicing. These early models laid the foundation but were limited by their simplistic assumptions [7].

In the 1990s and early 2000s, machine learning techniques began to replace these simpler models, offering more accurate predictions by capturing more complex patterns in genetic data [7]. More recently, with the advancement of sequencing techniques, data availability, and computational resources, deep learning models have further advanced the field, improving predictive performance by capturing more complex patterns and dependencies that earlier models could not [8].

While deep learning-based tools such as SpliceAI [3] and Pangolin [5] offer the highest prediction performance based on classification metrics [4], traditional (non-deep learning) approaches stay relevant due to their higher interpretability that can be crucial to clinical use cases [4]. One example of such a tool based on traditional machine learning that focuses on being interpretable is SQUIRLS [9].

Contrary to the deep learning-based tools [3], [5], SQIRLS and other relevant traditional machine learning models do not compute SSUPs for each base of the input sequence [9]–[11]. Since the goal of this work is to evaluate these predicted per-base probabilities, I focus on SpliceAI and Pangolin.

Both SpliceAI and Pangolin have a similar architecture based on a Convolutional Neural Network (CNN) [3], [5]. The main differences of these tools are (1) that SpliceAI predicts distinct probabilities for donor sites (5' end of the intron) and acceptor sites (3' end of the intron) [3], while Pangolin predicts one combined probability value [5], and (2) that SpliceAI is a single model used for all types of tissue [3], while Pangolin consists of four tissue-specific models for heart, liver, brain and testis [5].

B. Splicing in cancer

Alternative splicing has diverse effects on the human body, ranging from the production of multiple protein isoforms essential for normal physiological function to contributing to various diseases [1], [2]. Since abnormal splicing can lead to the development of tumors, one group of diseases with a major influence by alternative splicing is cancer [12]. In this work I evaluate the selected tools on patients with breast cancer, the leading cause of cancer-related deaths in women [13]. Specifically, I focus on the gene BRCA1 which has long been established to be related to breast cancer and is known to have different disease-relevant isoforms that arise from alternative splicing [13].

The data source selected for this work is The Cancer Genome Atlas, and specifically the Breast Invasive Carcinoma Collection (TCGA-BRCA) [14]. This dataset includes matched data on somatic mutations of the tumor tissue from whole genome sequencing (WGS) as well as RNA-seq data of the tumor tissue for 100 patients with a diagnosis of breast cancer [14]. The data from TCGA-BRCA was not used in the training of SpliceAI [3] or Pangolin [5].

III. Methods

The process of comparing predicted SSUPs to experimental observations from RNA-seq begins with reconstructing the tumor genomes to serve as input sequences for the prediction tools. After obtaining the SSUP predictions, these have to be realigned to the reference genome for consistency with the RNA-seq data and comparability across individuals. Finally, I estimate the experimentally observed SSUPs from the matched tumor RNA-seq data. The result are per-base predicted and experimentally observed SSUPs for each individual in the dataset.

A. Reconstructing the tumor genome

The input data to reconstruct the tumor genome are somatic mutations in Variant Call Format (VCF) [14]. From this data and the corresponding reference genome, I take the section of interest, in this case the BRCA1 gene. Starting with the reference sequence, I iterate over all variants in the section of interest, and substitute the $[1, n]$ bases in the reference sequence with the $[1, m]$ bases of the tumor's variant at the given reference index.

Since there are cases where $n \neq m$, the produced tumor sequences likely have unique index shifts compared to the reference. To later be able to compare the predictions to the reference-aligned RNA-seq data and across different individuals, I keep track of a mapping for which reference indices the bases of the tumor sequence correspond to.

The implementation of the tumor genome reconstruction can be found in the file `script/sequences.py` in the repository.

B. Predicting splice sites

For both SpliceAI and Pangolin, the outer 5000 bases on each side of the input sequence are treated as padding, and SSUPs are not predicted for these regions. To address this, I reconstruct a sequence that includes the BRCA1 gene along with the additional 5000 bases on each side. This padded sequence is one-hot encoded and then input into the models.

For Pangolin, I use the model that is specific for testis tissue. I selected this model based on the correlation of BRCA1 junction expression between breast tissue and the tissue types the Pangolin models were trained on. The expression values were extracted from the GTEx Data Portal API [15]. More detailed information on, and visualization of the correlation values, as well as the selection of the testis-specific model can be found in the repository in `tissue-selection.ipynb`.

SpliceAI outputs separate predictions for donor and acceptor probabilities. Since a splice site can only be either donor or acceptor [1], I take the per-base maximum of the predicted values and treat it as the SSUP.

The implementation of the predictors can be found in the file `script/predictions.py` in the repository.

C. Realigning predictions to the reference genome

The predicted SSUPs align with the genome sequences of the individual tumors. To achieve consistency with the reference-aligned RNA-seq data and comparability across different individuals, the results have to be realigned to the reference index. This is done based on the index mapping tracked in the realignment step in subsection III-A. For deleted bases, a predicted SSUP of 0 is assumed to have values for all reference indices.

The realignment implementation can be found in the file `evaluation.ipynb`.

D. Estimating experimental splice site usage probabilities

The TCGA-BRCA dataset includes files for splice junction quantification of 100 individuals [14] produced by the transcript alignment software STAR2 in the mRNA analysis pipeline of the National Cancer Institute [16]. These splice junction quantification files define information on the junctions that were observed in RNA reads during the sequencing. For each junction, the reference indices of the splice sites, as well as the absolute read count of the junction are given [17].

Directly computing SSUPs from RNA-seq data would require knowing the exact number of transcriptions of the given gene, as well as the exact counts for how many times each unique transcript of that gene occurred. However, due to the short read lengths of RNA-seq, many reads consist only of exonic regions and do not cross any splice junctions, meaning the read count of the given gene is always much higher than the read counts of the junctions. To address this, I instead estimate the SSUPs only based on the junction read counts, by taking the ratio of the count for a given junction and the count of the junction with the maximal read count in the gene. The predicted SSUPs are normalized analogously by the maximum predicted value to improve comparability.

The implementation of the splice junction analysis and SSUP estimation can be found in the file `evaluation.ipynb`.

IV. Results

To visualize and analyze the predicted and estimated experimental SSUPs, I present them organized by location on the gene and as a scatter plot in subsection IV-A, and then compute an error metric between the predicted and experimental values for direct numerical comparison in subsection IV-B.

A. Per-base predictions

Figure 1 shows the experimentally observed and predicted SSUPs by SpliceAI and Pangolin for each base of BRCA1, aggregated as the mean values across all 100 individuals. The exonic regions of the reference genome are highlighted as translucent red at the bottom of the plots. Less translucent shades of red indicate that the region corresponds to multiple possible exons that arise due to alternative splice junctions.

Figure 2 shows scatter plots of the experimentally observed SSUPs matched against the SSUPs predicted by SpliceAI and Pangolin, respectively. Each dot on the plots corresponds to the mean SSUP across the 100 individuals for a single base on BRCA1.

B. Asymmetric root mean squared error

To compute numerical goodness of fit of the predictions of SpliceAI and Pangolin, I compute an adjusted version of the regression metric root mean squared error across all individuals and bases. The adjustment is an additional scalar penalty α for underestimation of the SSUP, making the metric asymmetric. The adjusted root mean squared error RMSE_α across n matched experimental values Y and predicted values $\hat{Y}$ is given as

$$\text{RMSE}_\alpha = \sqrt{\frac{\sum_{i=1}^{n} w_i^\alpha \left(Y_i - \hat{Y}_i\right)^2}{\sum_{i=1}^{n} w_i^\alpha}},$$

where w_i^α denotes the underestimation penalty for the values at i with respect to α as

$$w_i^\alpha = \begin{cases} \alpha & \text{if } \hat{Y}_i < Y_i \\ 1 & \text{else} \end{cases}.$$

The values of this metric for different values of the underestimation penalty factor α and for SpliceAI and Pangolin are shown in Figure 3. Additionally, for interpretatability of the metric, two baselines are included, one baseline that always predicts and SSUP of zero, and one that always predicts 1.

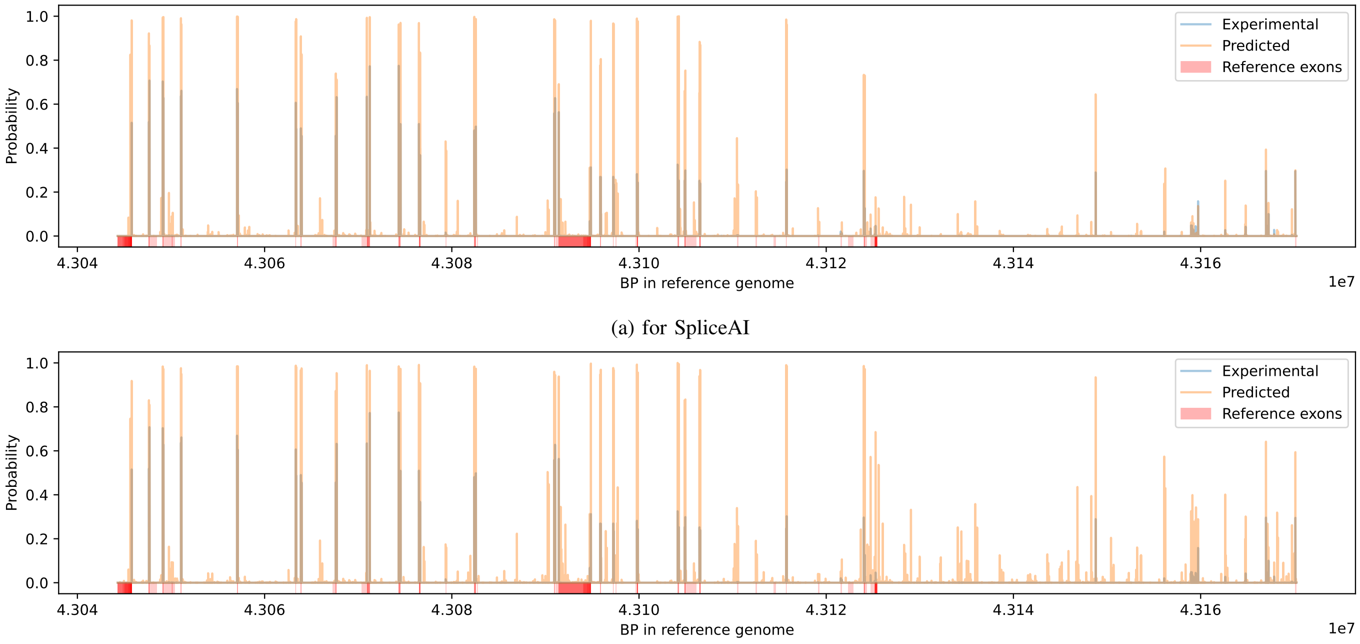
V. Discussion

I will discuss the results of the per-base predictions of SSUPs, followed by the asymmetric root mean squared error in dependence of the underestimation penalty factor α. Finally, I will discuss limitations of this work.

A. Per-base predictions

Figure 1 suggests overall accordance between the SSUPs from experimental observation and the predictions from both SpliceAI as well as Pangolin. Many of the common splice sites are at the borders of the reference exons as it would be expected. In addition to splice sites at the borders of exons, the RNA-seq data of the cancer patients also shows some splicing at locations where no reference exons are, for example around base pair (BP) $4.315\text{e}7$. This could be an alternative splice site attributed to the cancer of the patients. Despite this splice site not being part of the reference, both SpliceAI as well as Pangolin predicted it in the patients.

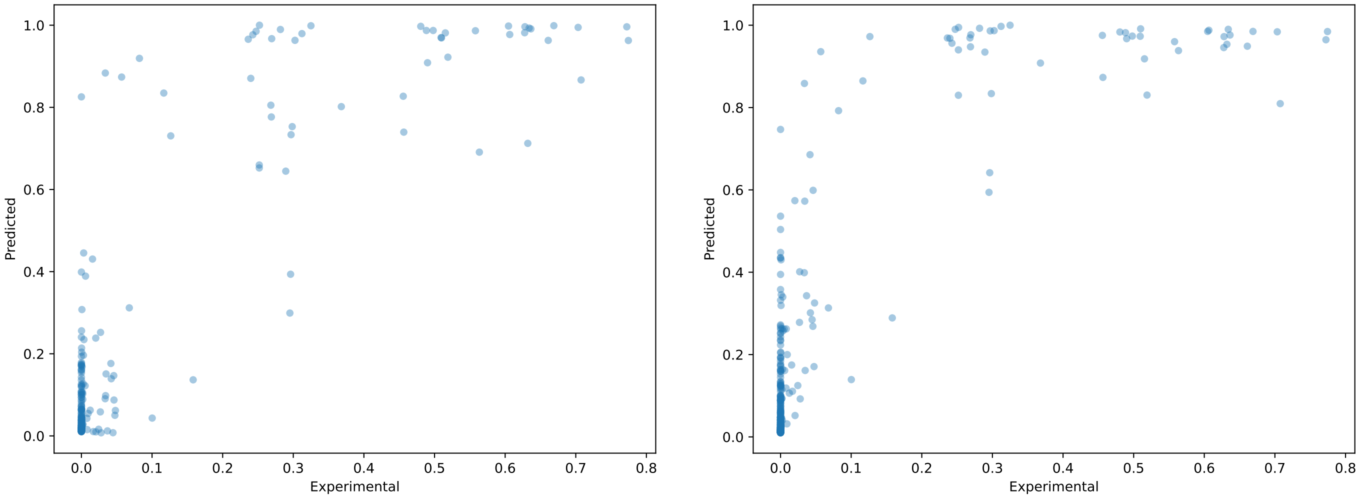
While the accordance between the experimental data and predictions could be assumed based on Figure 1, only Figure 2 confirms that the results do align exactly, and are not, for example, shifted by a small number of bases. In addition, Figure 2 reveals an upper-left triangle pattern for both prediction tools. This upper-left triangle is likely a result of favoring overestimation and of the classification-based metrics used in training of the tools. Since the predictions are thresholded to produce a binary outcome, overestimation above the threshold is not penalized. One clear difference visible on these plots is that there are bases where SpliceAI predicted SSUPs very close to zero for some bases that were observed to be splice sites in the experimental data. Pangolin, on the other hand,



(a) for SpliceAI

(b) for Pangolin

Fig. 1. Experimentally observed and predicted splice site usage probabilities by location on BRCA1, aggregated as mean across individuals



(a) for SpliceAI

(b) for Pangolin

Fig. 2. Scatter plots of experimentally observed vs. predicted splice site usage probabilities on BRCA1, aggregated as mean across individuals

always predicted non-zero SSUPs where splice sites were observed experimentally.

B. Asymmetric root mean squared error

The asymmetric root mean squared error metric is motivated by the asymmetric significance of errors in clinical practice. The cost of overestimating, i.e. incorrectly predicting bases to be splice sites, is the loss of the resources that are required to refute the prediction. Underestimating, i.e. missing a real splice site, however can mean missing and ignoring a disease-contributing factor, which can negatively affect the patients' outcomes. For this reason, I investigated how the numerical error develops while increasing α, the additional penalty factor for underestimation, in Figure 3.

At $\alpha = 1$ where underestimation and overestimation are penalized equally, the baseline of always predicting an SSUP of zero performs best in terms of this metric. This is to be expected, (1) due to the fact that most bases are not splice sites and therefore have an experimental SSUP of zero, and (2) due to the prediction tools favoring overestimation as seen in the upper-left triangles in Figure 2. As α increases, the prediction tools quickly outperform the baseline of zero, with SpliceAI

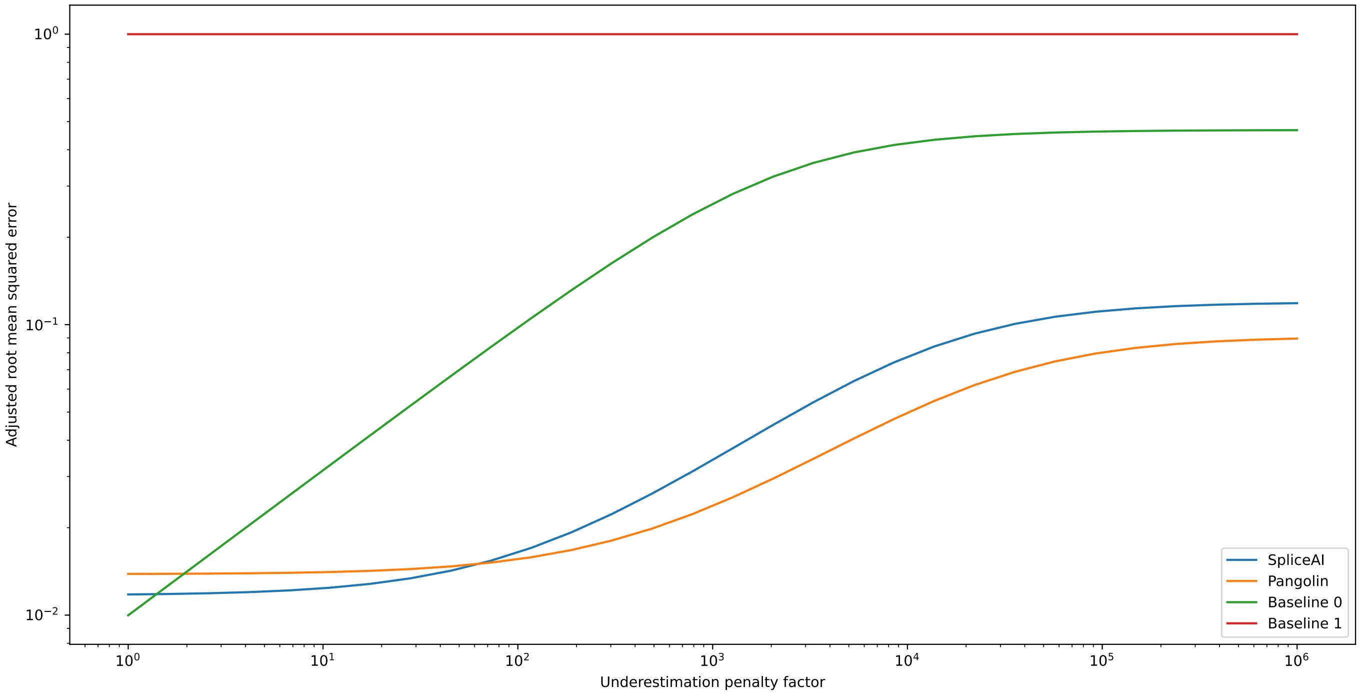


Fig. 3. Asymmetric root mean squared error between experimentally observed vs. predicted splice site usage probabilities with additional scaled penalty of underestimation.

having a better error value from around $\alpha = 1.5$, and Pangolin from around $\alpha = 2$. This means that if underestimation is considered to be at least 1.5 or 2 times as relevant as overestimation, the respective tools have better error values.

The intersection of the error values for SpliceAI and Pangolin is around $\alpha = 65$. If underestimation is considered to be less than 65 times as relevant as overestimation, SpliceAI performs better than Pangolin in terms of this metric, while Pangolin outperforms SpliceAI for factors bigger than 65. This behavior aligns with the scatter plots seen in Figure 2, where Pangolin overestimates more than SpliceAI, and, contrary to SpliceAI, never predicts an SSUP of zero for a base that was observed to be a splice site.

While the error values of SpliceAI, Pangolin and the baseline of zero change up until around $\alpha = 1e5$ and then stabilize, the baseline of always predicting an SSUP of 1 has a constant nearly maximal error. The poor performance of always predicting 1 is attributed to the relative rarity of splice sites. Most experimentally observed SSUPs are zero, making the prediction of 1 maximally incorrect.

The results of this error metric and Figure 3 offer insights on which tool might be the better choice for splice site prediction in BRCA1 on breast cancer patients depending on how important the avoidance of underestimation is considered to be for the given use case.

C. *Limitations*

The evaluation in this work assumes the results of RNA-seq to be the optimal goal for prediction tools. However, this assumption is not necessarily correct. RNA-seq data is the result of multiple stochastic processes, including the selection of transcriptions that were present in the sequenced cells at the moment of sequencing, the alignment of the RNA reads to the reference genome, and the recognition of splice junctions [18]. In addition, the SSUPs considered as experimentally observed are also only estimates on top of these stochastic processes as discussed in subsection III-D. While data that is subject to these limitations is the current best option for evaluation, the theoretical optimal goal for prediction tools would overcome these limitations.

Similarly as with the RNA-seq data, this evaluation assumes correctness of the tumor genome sequences used as input for the prediction tools. However, the somatic variants of the tumors used to reconstruct the genomes are also results of stochastic processes that are subject to errors. Furthermore, even though these variants are somatic variants, i.e. variants compared to the respective individual's healthy tissue, the reference genome was used as a baseline for reconstruction. This means that variants that were already present in the healthy tissue were ignored. While this can introduce errors, I expect them to be minor due to significant mutations such as those that are splice-altering being unlikely in healthy tissue.

VI. Conclusion

In this work, I introduced a pipeline to compute per-base splice site usage probabilities from genetic variants with SpliceAI and Pangolin, and to evaluate and compare them to RNA-seq results based on visualizations and a suitable error metric. I executed this pipeline for 100 breast cancer patients and the gene BRCA1. I discussed the results, concluding that both prediction tools are in general accordance with the RNA-seq data, and that the choice of the more suitable tool should consider how important the avoidance of underestimation is for the given use case. With this work, I contributed to the in-depth evaluation of two popular deep learning-based splice site prediction tools, and offered a new perspective on what to consider when selecting a tool for a use case.

A. Future work

In the continued work of this project, the implemented pipeline's generalizability should be taken advantage of to investigate more genes and different datasets. This is particularly valuable if the results of this bigger investigation are analyzed by a specialist of the underlying biology.

In addition to widening the considered data, a case study of a single or a few individuals will provide further in-depth insights into the characteristics of the prediction tools. In this context, it is especially insightful to investigate the tools based on patterns of errors and of high accuracy. A case study also presents the opportunity to revisit non-deep learning tools and explore how they compare to the deep learning-based predictors.

Finally, the reconstruction of the tumor genomes could be adjusted to consider the individual genomes of the healthy tissue as baselines. This should improve the correctness of the assumed tumor sequences and also allows for verification or refutation of my assumption that only minor errors are introduced by reconstructing based on the reference genome.

References

[1] M. A. Clark, J. Choi, and M. Douglas, "Biology 2e," in *Biology 2e*, Open Textbook Library, pp. 395–442, OpenStax, 2018.

[2] W. Jiang and L. Chen, "Alternative splicing: Human disease and quantitative analysis from high-throughput sequencing," *Computational and Structural Biotechnology Journal*, vol. 19, pp. 183–195, Jan. 2021. Publisher: Elsevier.

[3] K. Jaganathan, S. Kyriazopoulou Panagiotopoulou, J. F. McRae, S. F. Darbandi, D. Knowles, Y. I. Li, J. A. Kosmicki, J. Arbelaez, W. Cui, G. B. Schwartz, E. D. Chow, E. Kanterakis, H. Gao, A. Kia, S. Batzoglou, S. J. Sanders, and K. K.-H. Farh, "Predicting Splicing from Primary Sequence with Deep Learning," *Cell*, vol. 176, pp. 535–548.e24, Jan. 2019.

[4] C. Smith and J. O. Kitzman, "Benchmarking splice variant prediction algorithms using massively parallel splicing assays," *Genome Biology*, vol. 24, p. 294, Dec. 2023.

[5] T. Zeng and Y. I. Li, "Predicting RNA splicing from DNA sequence using Pangolin," *Genome Biology*, vol. 23, p. 103, Apr. 2022.

[6] J. Baum, "jannis-baum/hpi-hoai," 2024.

[7] X. Jian, E. Boerwinkle, and X. Liu, "In silico tools for splicing defect prediction: a survey from the viewpoint of end users," *Genetics in Medicine*, vol. 16, pp. 497–503, July 2014. Publisher: Nature Publishing Group.

[8] G. Eraslan, Z. Avsec, J. Gagneur, and F. J. Theis, "Deep learning: new computational modelling techniques for genomics," *Nature Reviews Genetics*, vol. 20, pp. 389–403, July 2019.

[9] D. Danis, J. O. Jacobsen, L. C. Carmody, M. A. Gargano, J. A. McMurry, A. Hegde, M. A. Haendel, G. Valentini, D. Smedley, and P. N. Robinson, "Interpretable prioritization of splice variants in diagnostic next-generation sequencing," *The American Journal of Human Genetics*, vol. 108, pp. 1564–1577, Sept. 2021.

[10] K. A. Jagadeesh, J. M. Paggi, J. S. Ye, P. D. Stenson, D. N. Cooper, J. A. Bernstein, and G. Bejerano, "S-CAP extends pathogenicity prediction to genetic variants that affect RNA splicing," *Nature Genetics*, vol. 51, pp. 755–763, Apr. 2019.

[11] A. Rosenberg, R. Patwardhan, J. Shendure, and G. Seelig, "Learning the Sequence Determinants of Alternative Splicing from Millions of Random Sequences," *Cell*, vol. 163, pp. 698–711, Oct. 2015.

[12] Y. Zhang, J. Qian, C. Gu, and Y. Yang, "Alternative splicing and cancer: a systematic review," *Signal Transduction and Targeted Therapy*, vol. 6, pp. 1–14, Feb. 2021. Publisher: Nature Publishing Group.

[13] N. Martínez-Montiel, M. Anaya-Ruiz, M. Pérez-Santos, and R. Martínez-Contreras, "Alternative Splicing in Breast Cancer and the Potential Development of Therapeutic Tools," *Genes*, vol. 8, p. 217, Oct. 2017.

[14] W. Lingle, B. J. Erickson, M. L. Zuley, R. Jarosz, E. Bonaccio, J. Filippini, J. M. Net, L. Levi, E. A. Morris, G. G. Figler, P. Elnajjar, S. Kirk, Y. Lee, M. Giger, and N. Gruszauskas, "The Cancer Genome Atlas Breast Invasive Carcinoma Collection," 2016.

[15] The GTEx Consortium, "GTEx Portal API," 2024.

[16] NCI Genomic Data Commons, "Bioinformatics Pipeline: mRNA Analysis," 2024.

[17] A. Dobin, "STAR Manual," Jan. 2024. Version 2.7.11b.

[18] C. M. Koch, S. F. Chiu, M. Akbarpour, A. Bharat, K. M. Ridge, E. T. Bartom, and D. R. Winter, "A Beginner's Guide to Analysis of RNA Sequencing Data," *American Journal of Respiratory Cell and Molecular Biology*, vol. 59, pp. 145–157, Aug. 2018.