A Bayesian model for unsupervised detection of RNA splicing based subtypes in cancers

David Wang,[1,2] Mathieu Quesnel-Vallieres,[1,3] Paul Jewell,[1] Moein Elzubeir,[1] Kristen Lynch,[1,3] Andrei Thomas-Tikhonenko,[4,5] Yoseph Barash[1,6*]

[1]Department of Genetics, Perelman School of Medicine, University of Pennsylvania
[2]Graduate Group in Genomics and Computational Biology, Perelman School of Medicine, University of Pennsylvania
[3]Department of Biochemistry and Biophysics, Perelman School of Medicine, University of Pennsylvania
[4]Department of Pathology and Laboratory Medicine, Perelman School of Medicine, University of Pennsylvania
[5]Division of Cancer Pathobiology, Children's Hospital of Philadelphia
[6]Department of Computer and Information Sciences, School of Engineering, University of Pennsylvania

*To whom correspondence should be addressed; E-mail: yosephb@upenn.edu.

Identification of cancer sub-types is a pivotal step for developing personalized treatment. Specifically, sub-typing based on changes in RNA splicing has been motivated by several recent studies. We thus develop CHESSBOARD, an unsupervised algorithm tailored for RNA splicing data that captures "tiles" in the data, defined by a subset of unique splicing changes in a subset of patients. CHESSBOARD allows for a flexible number of tiles, accounts for uncertainty of splicing quantification, and is able to model missing values as additional signals. We first apply CHESSBOARD to synthetic data to assess its domain specific modeling advantages, followed by analysis of several leukemia datasets. We show detected tiles are reproducible in independent studies, investigate their possible regulatory drivers and probe their relation to known AML mutations. Finally, we demonstrate the potential clinical utility

of CHESSBOARD by supplementing mutation based diagnostic assays with discovered splicing profiles to improve drug response correlation.

Introduction

Analysis of RNA sequencing (RNA-seq) data from large patient cohorts is commonly used to reveal transcriptomic variations that are associated with complex disease. Within the framework of machine learning, such analysis can be framed as either supervised or unsupervised learning tasks. In a supervised setting, the objective is usually to identify transcriptomic variations that act as predictive markers for disease state or are strongly correlated with clinically significant variables[1;2;3]. Unsupervised analysis typically involves identifying latent substructures in the data which can be used to learn more about disease etiology, such as cancer subtypes[4;5]. One approach to quantify changes in the transcriptome is at the level of alternative splicing (AS). AS is the process by which different segments of pre-mRNA can be removed while others are joined, or spliced, together to form mature mRNA. AS is regulated by intricate interactions between hundreds of RNA binding proteins (RBPs) and is thus highly susceptible to disease-causing disruption, especially in cancer[6;7;8]. Given the strong association between splice variants and disease, we propose an unsupervised learning algorithm for identifying substructures in a matrix of RNA splicing measurements from cancer patients.

The focus on identifying substructures in RNA splicing cancer data is motivated by several additional observations. First, in cancers such as acute myeloid leukemia (AML), the mutation burden is particularly low such that analyzing genetic mutations alone is insufficient for explaining disruption of key oncogenic pathways[9]. Instead, several works have pointed to splicing aberrations which can severely perturb the function of regulatory proteins involved in apoptosis and cancer suppression[10;11]. Second, many cancers have been shown to be heterogeneous, with patients exhibiting high variability in splicing measurements. While some of this variability is

likely due to confounders such as batch effects, recent studies have shown that this variability can result from mutations which seldom appear in a large fraction of the patients[12]. Rather, they are observed in small subsets of patients with both cis acting effects due to mutations at splice sites and trans acting effects due to mutations in splicing regulatory machinery[13]. These observations motivate the derivation of a dedicated method for identifying "tile" substructures in the RNA splicing data matrix, i.e. subsets of patients that exhibit distinct splicing changes in a subset of genes.

Splicing variations derived from RNA-seq are commonly defined at the level of whole transcripts or at the level of AS "events". Transcript based approaches rely on estimating the abundance of whole isoforms (e.g. RSEM[14], SALMON[15], Kalisto[16]) or relative isoform usage (e.g. MISO[17], BANDITS[18]). In contrast, methods such as MAJIQ[19], SUPPA2[20], rMATS[21], quantify splicing of events, or local splicing variations (LSVs), such as cassette exons. These local splice variations measure splicing as the ratio of RNA segment (e.g. exon) inclusion (defined as percent spliced in or $\Psi \in [0, 1]$) for isoforms that contain the segment vs. isoforms that do not. While quantifying whole isoforms is clearly appealing, we focus here on event based splicing quantification. Some advantages of using AS events include the fact they do not require a-priori assumptions about the underlying isoforms, can handle un-annotated (denovo) isoforms which is crucial in cancer analysis, and can be directly validated with orthogonal methods (e.g. RT-PCR).

Despite the above advantages, there are several modeling challenges associated with unsupervised tile identification using event based quantifications. First, splicing measurements are inherently different from gene expression measurements which are modeled as Gaussian (TPM) or negative binomial (read counts) distributions in many previous works[22;23;24;25;26]. In contrast, Ψ is bounded in the $[0, 1]$ interval, with inclusion levels commonly exhibiting a U shape distribution, favoring either high or low exon inclusion values. Furthermore, Ψ quantifications

typically involve only a small subset of reads that span splice junctions, thus accounting for uncertainty of Ψ estimates is important. Finally, when identifying substructures in cancer RNA splicing measurements it is important to address the inherent heterogeneous nature of the data and natural variations between individuals. Specifically, global patterns across rows (Ψ for specific AS events) or columns (patients) are unlikely. Instead, only a small subset of LSVs may be perturbed in a subset of samples.

Another important challenge we address here, which has implications beyond the analysis of RNA splicing data, is the modeling of missing values. Genomics data often contains missing values that result from technical limitations in sequencing technologies and are assumed to be missing completely at random (MCAR). Under this model, the missingness rate does not depend on observed or unobserved values and can be imputed or ignored[27]. However, in RNA-seq data, the missingness rate is inversely proportional to the sequencing depth where higher read coverage results in a lower probability of missingness. Furthermore, splicing quantifications, unlike expression measurements, cannot be meaningfully imputed since a missing Ψ quantification can be denoted by a 0 or 1 representing alternate junction usage. Thus, naive imputation (e.g. mean) can lead to unlikely intermediate values. This necessitates an alternate model in which values are MNAR (missing not at random). Under this model, the missingness rate depends on observations in the data matrix and external factors such as coverage. In cancer data specifically, values can also be systematically missing due to genetic mutations which could result in a specific junction not being observed (e.g. mutations near splice sites) and should be modeled as a secondary signal.

Here we address the above modeling challenges, by developing CHESSBOARD (Characterizing Heterogeneity of Expression and Splicing by Search for Blocks of Abnormalities and Outliers in RNA Datasets). CHESSBOARD is a Bayesian tile finding algorithm tailored for splicing data with missing values and includes a suite of data processing and visualization tools

(Fig. 1). The input consists of a matrix of junction spanning reads counts to account for uncertainty in splicing quantifications (Fig. 1a). The algorithmic task is to identify splicing patterns in the form of tiles in this matrix (Fig. 1b). This is achieved by first employing model based pre-filtering to remove irrelevant LSVs and reduce and data size (Fig. 1c, left). Next, CHESSBOARD's non-parametric Bayesian tiles model is fit to the data using efficient blocked Gibbs sampling (Fig. 1c center). Finally, posterior summary statistics can be visualized to perform downstream analysis (Fig. 1c right).

We first apply CHESSBOARD to synthetic datasets to show it outperforms several baseline methods and validate the effectiveness of our modeling approach. Next, we show that CHESSBOARD recovers tiles characterized by splicing aberrations which are reproducible in multiple AML patient cohorts. Finally, we show that tiles we discover are correlated with drug responses, pointing to translational potential of our findings. We also develop GAMBIT (Graphical Annotated Map for Basic Inspection of Tiles), a web-based visualization tool which allows users to visually explore the discovered tile structures and Bayesian output. Both CHESSBOARD and GAMBIT are available as open source tools to facilitate reproducible workflow and analysis.

Results

CHESSBOARD robustly models alternative splicing and missing values to discover tile structures in large heterogeneous datasets

To address the challenges of analyzing heterogeneous alternative splicing datasets, CHESSBOARD directly models properties of the data that arise from biological and technical processes. Briefly, the model's input data matrix X contains the number of junction spanning reads x_{ij} mapped to the representative (i.e. most variable) splice junction of LSV j in sample i and the total number of reads mapped to the LSV, denoted η_{ij} (more input details in Supplementary Note 1.2). Under CHESSBOARD's model, naturally occurring splicing variations in each LSV are

captured by a (learned) mixture of a Beta-binomial distribution over each x_{ij} and a binomial distribution (defined by missingness rate θ_{j0}) for having a missing value. This mixture distribution over observed and missing values captures the background. In specific patient subsets however, additional variation or signal in underlying Ψ (captured by a separate Beta-binomial distribution over observed values) or an elevated missingness rate (captured by a separate θ_{j1}) may be observed. Thus, the first part of the CHESSBOARD pipeline is to filter out non-informative splicing events which can be captured well by the background distribution. This is achieved using a parametric bootstrap Kolmogorov-Smirnov test (Supplementary Note 2.1). Then, for the remaining LSVs in the data matrix, CHESSBOARD aims to find latent "tiles" in which multiple LSVs deviate from the background in the same subset of samples. In practice, this means that every sample i belonging to tile k has its (unknown) group indicator variable set $c_i = k$ and every LSVs j belonging to this tile has a matching (also unknown) indicator variable $r_{jk} = 1$. A specific CHESSBOARD model is represented by a learned tile configuration and distribution parameters for all the background and signal groups. Under this Bayesian formulation, every such model can be assigned a posterior probability, and the CHESSBOARD algorithm uses an efficient blocked Gibbs sampling procedure to sample from the posterior distribution over possible models given the observed data matrix X. See Methods for a detailed description of the CHESSBOARD model.

In this section we demonstrate the utility of the CHESSBOARD model formulation described above. First, we show that CHESSBOARD accounts for uncertainty in splicing measurements due to low sequencing coverage. Specifically, CHESSBOARD includes a beta binomial distribution which attributes higher variance to LSVs with low coverage, capturing increased uncertainty in their underlying Ψ. To assess whether this model is advantageous for estimating variability in splicing data, we simulate Ψ values from a Beta distribution modeling low exon inclusion ($Beta(10, 90)$) and generate reads for each Ψ at various coverage levels from

a binomial distribution (Fig. 2a). We compute the empirical variance using MLE (maximum likelihood estimation) under the CHESSBOARD model and two alternative models: a Beta model which also functions on a domain of [0,1] analogous to Ψ values and a Gaussian model which represents a generic approach. This analysis shows the error in the variance estimation is lowest for the Beta Binomial model and all 3 models converge at about 50 reads. However, in real life datasets the majority of quantifiable LSVs have read coverage significantly lower than 50. For example, in the beatAML and TARGET datasets used in this study (see Fig. 2a as red and blue histograms), 38% and 88% respectively have coverage below this level, indicating a substantial portion of the data benefits from CHESSBOARD's modeling.

CHESBOARD's model further alleviates the effect of coverage dependent uncertainty in heterogeneous data by sharing information across samples. Specifically, CHESSBOARD uses empirical Bayesian shrinkage to learn group specific priors, taking advantage of samples with higher coverage assigned to the same cluster to improve estimates for samples with lower coverage (Methods). To demonstrate the advantages of CHESSBOARD's modeling approach we generated Ψ values from $Beta(10, 90)$ and read counts from each Ψ at varying levels of coverage as before. The results shown in Fig. 2b demonstrate that indeed there is lower error in Ψ estimates in samples with lower read counts when estimates are shrunken to the group mean compared to computing $\hat{\Psi}$ without prior information. Furthermore, we show that shrinkage significantly increases correlation of $\hat{\Psi}$ and the true value of Ψ, especially in samples with low coverage. As denoted in Fig. 2c, $\hat{\Psi}$ for darker data points representing low coverage samples is closer to the ground truth with group shrinkage (right) compared to individual quantification (left). Together these experiments show that CHESSBOARD's generative Beta-Binomial model acting on read counts can substantially improve analysis of splicing data by accounting for uncertainty in the RNA sequencing measurements.

Next, we turned to assess CHESSBOARD's missing values modeling. To account for miss-

ing values, CHESSBOARD uses a MNAR model where missing values are treated as a secondary signal when the missingness rate of an LSV is much higher than expected under the null missingness rate associated with sequencing limitations. We first replace each unquantifiable splicing event with a missingness indicator. We then estimate priors for the missingness rates using an Empirical Bayes procedure (Methods). During CHESSBOARDs model fit, we obtain posterior estimates for both the background and signal missingness rate, where the latter can account for other factors such as unobserved values due to mutations (Methods). We show that the MNAR missing value model is effective in identifying tiles containing missing value signals. For comparison, we implemented an alternative version of CHESSBOARD that uses the MCAR model assumption where missing values are integrated out. Both CHESSBOARD and CHESSBOARD-MCAR were then applied to a simulated homogeneous data matrix in which the read counts for each LSV (row) were drawn from a background distribution with parameters estimated from beatAML and values were missing at a fixed dropout rate of 10%. To these we added a single tile of varying size with a missingness rate of 60%. We then assessed the algorithm's ability to recover this tile by information gain which measures the purity of the clusters (Supplementary Note 2.2). Fig. 2d shows that the MCAR model was unable to identify this tile (information gain close to 0) regardless of tile size, as it relies solely on the observed Ψ values to identify tiles. The MNAR model is able to effectively recover the tile, but as expected the information gain decreases as the tile size decreases. When the size of the tile increases to 40 LSVs, the information gain reaches a maximum.

After assessing the CHESSBOARD modeling components individually, we turned to assess its ability to recover tiles. For this, we generated synthetic splicing data modeled based on statistics collected from the BeatAML dataset (Supplementary Note 2.3). For comparison, we also ran two commonly used algorithms for biclustering as baselines: single-link hierarchical biclustering (HBC) and spectral co-clustering (SCC)[28]. Since these algorithms lack some of

CHESSBOARD's features, both were given data with scalar Ψ and no missing values to fit their input definition. We also ran both with the correct number of clusters given as input. Since CHESSBOARD learns the number of clusters using an infinite mixture modeling approach with a Chinese Restaurant Process (CRP) prior (Methods), we first evaluated the behavior of this feature under various parameters (Supplementary Note 2.4). Then the ability of the algorithms to recover tiles was evaluated using a tile precision (τ_{pr}) and recall (τ_{rc}) statistic adapted from the recovery and relevance score[25] (Supplementary Note 2.2). Intuitively, these scores identify the tile in the test set that maximizes precision or recall with respect to each of the reference tiles, then average the precision or recall across the tiles. In addition, we evaluated sample group clustering using adjusted rand index (ARI) (Supplementary Note 2.2). The results in Fig. 2e show that all algorithms were able to recover sample groups well (ARI > 0.9). This result is to be expected given the strong group signal (number of changing LSV, magnitude of change) in the original data (see BeatAML analysis below) and the fact the baseline algorithms were initialized with the exact cluster number. However, CHESSBOARD significantly outperformed the baseline algorithms in recovering the exact tiles, achieving τ_{pr} = 1.00, τ_{rc} = 0.98 compared to τ_{pr} = 0.33, τ_{rc} = 0.68 for HBC and τ_{pr} = 0.78, τ_{rc} = 0.55 for SCC.

CHESSBOARD Discovers reproducible tiles in AML data which correlate with cancer associated regulators

Having established strong performance of the CHESSBOARD model on synthetic data, we applied it to several primary leukemia sample datasets to discover tiles that correspond to cancer associated regulators. We ran the standard CHESSBOARD pipeline (Supplementary Note 3.1) on the beatAML[12] dataset (samples = 477, LSVs = 2299) (Supplementary Data 1). The algorithm detected a single large tile consisting of 217 samples and 1910 LSVs (Fig. 3a). Confidence in the predicted tile structure was high with most probabilities of sample assignment to

the tile cluster and LSV assignment to the signal distribution being close to 1 (Supplementary Fig. 4a). To confirm whether this tile constitutes a real biological signal, we first assessed its reproducibility in Penn HTSC, an independent in-house dataset consisting of 77 adult AML samples. We trained the CHESSBOARD model on a random subset of the beatAML cohort (samples = 400) and used this as a predictive model to predict the tile assignments of the held out beatAML samples (samples = 77) and Penn HTSC (samples = 77) samples (Supplementary Note 3.2). We used MOCCASIN[29] to account for confounding factors between the datasets. The prediction yielded a similar tile structure in the Penn HTSC dataset (Fig. 3b). Furthermore, the $median(\Delta\Psi)$ (change in Ψ) of LSVs belonging to the tile between the 2 groups in each dataset are highly correlated (r = 0.779) suggesting that the splicing perturbations captured by the tile are similar in both datasets (Fig. 3c). Sample likelihoods were also comparable between the held out and external data indicating that the model has similar confidence in the tile structure predictions (Supplementary Fig. 4b).

Having established the reproducibility of the AML splicing tile in two independent cohorts, we then turned to investigate potential mechanisms for formation of this tile. First, we tested whether the identified tile was enriched for differentially spliced junctions that are co-regulated by RNA Binding Proteins (RBPs). Intersecting the tile's differentially spliced junctions with those observed as differentially spliced in ENCODE's RBP knockdown experiments implicated 17 RBPs, all of which were either differentially expressed or spliced between the signal and background patient groups (Supplementary Note 3.3). Put together, all 106 RBPs considered in the analysis affected approximately 11.75% of the junctions in the signal tile. Notably, two RBPs with the most significantly enriched DS junction overlap include *SRSF1* (2.48%) and *U2AF2* (1.54%), both of which have known roles in promoting expression of antiapoptotic isoforms of oncogenes in several hematopoietic maglicanies[30]. Another candidate splicing regulator which appeared to be differentially expressed and spliced between the signal and back-

ground groups is *HNRNPC* (2.44%), which has been implicated in AML in a recent study[31]. Next, we analyzed eCLIP data for each RBP to test whether there was evidence for direct RBP binding around the tile's differentially spliced junctions. We observed high binding rates for *SRSF1* and *U2AF2* ($>$ 4% of tile junctions). However, the binding rate was lower compared to spliceosome components including *AQR*, *SF3B4*, *PRPF8* and *EFTUD2* which are known to bind spuriously to constitutive splice sites. Surprisingly, almost no binding was observed for *HNRNPC*. For *SRSF1* specifically, there was also significant enrichment of CLIP binding to junctions that were also DS suggesting direct splicing regulation by *SRSF1* (Fig. 3d).

Interestingly, *SRSF1* itself undergoes alternative splicing whereby one isoform includes exon 4 for the production of the full protein while the other skips exon 4, resulting in a transcript that contains a premature termination codon that is targeted for nonsense-mediated decay[32]. We thus assessed whether variation in *SRSF1* exon 4 splicing between the 2 clusters corresponds to splicing variations in its known targets. Observed differences in *SRSF1* splicing between the signal and background occurred almost exclusively at exon 4 (Supplementary Fig. 4c). The background cluster had a higher rate of inclusion for exon 4 (Ψ = 0.759) compared to the signal cluster (Ψ = 0.490) and higher expression of the functional transcript (log2FC = 1.16). This suggests there is higher expression of the productive isoform in the background cluster due to lack of NMD-induced degradation. Over expression of *SRSF1* has been associated with aberrant splicing of several apoptotic factors in cancer[32;33]. We analyzed several cancer-associated genes with experimentally verified splice variations that are affected by *SRSF1* overexpression. Notably, *BIN1* exon 12a inclusion is upregulated in the background and is associated with antiapoptotic processes[33]. Furthermore, exon 3-6 inclusion in *CASP9* is upregulated in the background and is associated with proapoptotic processes[34] (Supplementary Fig. 4c).

CHESSBOARD offers a gene ranking method that implicates mTORC signaling in identified differentially spliced gene set

In order to more broadly assess whether the AML identified tiles correspond to known biological functions, we performed a gene ontology analysis of biological processes with the genes harboring LSVs in an extended tile containing all DS LSVs between the two clusters. The analysis shows that genes with differential splicing in the tile are enriched for roles related to general functions commonly found in cancer transcriptomics studies such as gene expression and transcription, RNA processing, and post-translational modifications (Supplementary Fig. 4d). However, we also found that a subset of genes participate in stress-related cellular responses, including regulation of cholesterol/lipid storage and MAPK-signaling (Supplementary Fig. 4d highlighted in red) suggesting that samples from the two clusters exhibit different cellular stress profiles.

To further explore possible tile characterization, we sought to use gene set enrichment analysis (GSEA) to identify similar pathways in the tile gene sets. Since GSEA requires ranking genes within a group we developed a probabilistic ranking method based on the CHESSBOARD model which account for both splicing changes and missingness rates. Specifically, we score each LSV based on the likelihood gain achieved in the learned tile configuration compared to an inverted tile configuration and then used the score for the highest scoring LSV in each gene as input to GSEA (see Supplementary Note 3.4 for details). Indeed, GSEA revealed an enrichment of differentially spliced genes in the tile among the hallmark mTORC1 signaling gene set (Fig. 3e), a signaling pathway centrally involved in stress response[35]. Drawing from experimentally validated interactions extracted from the Ingenuity Pathway Analysis software, we confirmed that several genes that harbor high-ranking LSVs in the tile interact directly with mTORC1 or one of its direct regulators, and that many of these genes activate mTOR signaling, although it is unclear how the splicing variations that we observe might affect the function

of the proteins (Supplementary Fig. 4e). Collectively, these results suggest that the main tile structure CHESSBOARD identified in the BeatAML data represents a highly reproducible and biologically relevant AML subtype.

CHESSBOARD Enables Scalable Recursive Clustering to Discover Alternate Subtype Definitions

Although CHESSBOARD was able to successfully discover a tile corresponding to an AML subtype characterized by a specific set of splicing events, other subtype definitions may exist. Alternative tile structures representing these subtypes can emerge when the inclusion of additional features or exclusion of selected features alters the amount of evidence supporting existing tile boundaries. Intuitively, a tile can be interpreted as a collection of correlated transcriptomic signatures that each capture a misregulated biological process. For example, the tile discovered in the previous section is partially explained by misregulation of RBPs. Removal of a signal dominated by certain processes can lead to discovery of tiles characterized by misregulation of orthogonal, possibly less pronounced (in terms of number of splicing changes and their magnitude), pathways where other splicing perturbations exist in a different subset of patients. Similarly, LSVs not present in the initial pre-filtered data matrix may provide additional support for such tiles. To address this scenario we developed a recursive clustering solution that naturally fits into CHESSBOARD's probabilistic model and serves as a scalable means to probe the entire transcriptome (Supplementary Note 4.1). In short, our approach iteratively reclusters the LSVs that are not assigned to a tile and after each recursive step, tests for termination by assessing the likelihood ratio of the tile model to a null model in which tile structure is removed. This null model can be interpreted as a distribution over tile structures in datasets where tiles are not expected to occur. Furthermore, the result of each recursive step can be extended to the whole transcriptome using MAJIQ or similar tools for differential splicing analysis between the

sample groups identified by CHESSBOARD.

We performed recursive clustering on the beatAML dataset using this approach, treating the result from the previous section as the base case. The first recursive step yielded a smaller signal tile (samples = 196, LSVs = 389) corresponding to a different subset of the patients (ARI = 0.0066 between recursive and base case sample clusters) (Supplementary Data 2). In addition, the new cluster had high correlation with several known AML mutations (Fig. 4a). Specifically, we observed a significant permutation test p-value for enrichment of mutations in *FLT3*-ITD ($p < 0.001$), *NPM1* ($p < 0.001$), and *CEBPA* ($p = 0.025$) in patients assigned to the tile cluster (Supplementary Note 4.2). Mutations in these 3 genes are associated with normal karyotype AML which is a known subtype of the disease[36]. We observed a fourth association of the cluster was with mutations in *NRAS* ($p = 0.025$), but mutations in this gene were actually depleted in the tile samples. Continued recursive tile discovery showed a sharp decrease in the likelihood ratio (Fig. 4b) indicating there are no more significant tiles in the matrix to discover.

CHESSBOARD Identifies Tiles that Correlate with Drug Responses

To demonstrate the translational utility of conducting a CHESSBOARD analysis, we assessed whether tiles discovered by the algorithm correlate with patient response to therapeutics. We ran the CHESSBOARD pipeline on the beatAML dataset again but now limited the analysis to only LSVs in 70 AML associated genes (Supplementary Data 3). These genes have been identified as commonly mutated, truncated or translocated in AML patients[36;37] and their mutational status is used by clinicians to decide on drug administration. This targeted tile finding approach based on known gene sets is motivated by several observations. First, we demonstrated above that a transcriptome wide approach can be dominated by signals orthogonal to pathways inhibited by a drug. Second, and as we show below, CHESSBOARD's unsupervised approach can detect splicing signals not directly captured by the mutational landscape in such AML associ-

ated genes. Finally, as demonstrated in Rivera et al. 2021 [10], clustering splicing changes across those 70 genes gives rise to clear groups and several candidate regulators.

Our splicing analysis of the 70 AML associated genes recovered 2 clusters (Fig. 5a) with resulting patient subgroups similar to the original clustering (ARI = 0.958). This result is notable since the LSV set used for this analysis was significantly different, with shared LSVs constituting only 0.57% of the original LSV set and 14.4% of the current set. This result suggests the splicing changes in AML related genes are part of perturbations to pathways captured in the original, unbiased, LSVs tile finding.

Since the mutational status in many of these AML associated genes is used by clinicians to decide on drug administration we correlated the samples belonging to each tile with drug response measured by area under the IC_{50} curve (AUC). Details about this measurement are discussed in Supplementary Note 5.1. We observed strong correlations between drug response and the tiles, and noticed the tiles included aberrant splicing in many gene targets of the most correlated drugs (Supplementary Data 4). We therefore first tested whether our splicing based patient stratification can serve as good predictors of drug response. Specifically, we computed for each drug the percent of AUC variance that can be explained (Supplementary Note 5.2) by CHESSBOARD's discovered sample groupings compared to that explained by known mutations. However, this analysis conclusively found the variance explained by the patient subgroups to be relatively low, maxing at 6.7% compared to a much higher percentage for known mutations and drug combinations (Fig. 5b). Specifically, the variance explained by *FLT3* mutation is highest for Gilteritinib and *FLT3*-ITD is highest for Sunitinib/Sorafenib which are known mutation-drug associations.

Next, given the possible functional consequences of splicing changes between the identified tiles in many AML associated genes, we hypothesized that our splicing based patient grouping could improve clinical decisions based on mutation analysis alone. Notably, the added value

of splicing changes to AML classification has been shown recently for *FLT3*-ITD and *NPM1* for FAB classification AML genes[38] as well as RUNX1 and SF3B1[39]. To assess usefulness of combining splicing changes and mutations in AML genes splicing we prioritize *FLT3*-ITD and Sorafenib and *NPM1* and Venetoclax due to reliable mutation calls and their prominent role in AML clinical diagnosis. Specifically, we developed a simple decision tree (Fig. 5c) combining both *FLT3*-ITD and patient subgroups which increased the AUC variance explained by *FLT3*-ITD for Sorafenib from 26.0% to 36.8% (Fig. 5d). We also looked at median change in AUC and IC_{50} fold change (FC) to confirm that the effect size differences between the groups is biologically meaningful. Accordingly, using *FLT3*-ITD alone had median(ΔAUC) = 64.36 and Log3FC = 2.09, while the combined classification had a median(ΔAUC) = 76.29 and Log3FC = 2.73 (p = 0.034 by permutation test, see Supplementary Note 5.3). Similarly for *NPM1*, using the *NPM1* mutation status alone had a median(ΔAUC) = 23.34 and Log3FC = 0.86, while the combined classification had a median(ΔAUC) = 57.91 and Log3FC = 2.65 (p = 0.048).

CHESSBOARD's identified tiles include splicing changes in AML drugs' target genes

To assess potential mechanisms by which CHESSBOARD tiles explain drug response as described above, we looked for specific targets of Sorafenib in the tile that were differentially spliced between the 2 groups. Sorafenib is commonly used as a treatment for AML patients with a *FLT3*-ITD mutation and functions as a tyrosine kinase inhibitor with high specificity for *FLT3*. At a molecular level, *FLT3*-ITD result in constitutive activation of receptors that lead to downstream activation of PI3K/AKT/mTOR, Ras/Raf/MEK/ERK, and JAK/STAT pathways. This activation in turn results in enhanced proliferation and reduced apoptosis of the myeloblasts, which contribute to leukemogenesis[40]. Inline with this known mechanism, we observed multiple LSVs in *FLT3* that were differentially spliced between the patients' subgroups.

We therefore analyzed several specific splicing events in *FLT3* to determine if there was enrichment of splice isoforms in the signal that may lead to reduced transcript viability and thus higher sensitivity. Indeed, for *FLT3* we identified two differential splicing events involving skipping of exon 4b (p = 1.13e-43, $\Delta\Psi = 0.110$) and exon 17b (p = 4.34e-33, $\Delta\Psi = 0.115$) that are highly correlated (r = 0.858) and have a higher skipping rate in the background. Notably, exon 4b has not been previously reported (de-novo exon and junctions) and both events have not been previously reported with respect to AML to the best of our knowledge. Skipping both of these exons results in the functional canonical isoform of *FLT3* which was correlated with an increase in expression of *FLT3* (Fig. 5e). In contrast, inclusion of this exon introduces a PTC or frameshift in the alternate isoform.

Taken together, the above analysis suggests that there is over-expression of *FLT3* in the background cluster due to constitutive expression of canonical *FLT3* and failure of regulatory systems to induce NMD. Noting that Sorafenib is a tyrosine kinase inhibitor, which includes *FLT3*, an increase in the concentration of its target would therefore be expected to increase drug sensitivity, as the splicing changes we detected could mimic the gain of function effect of *FLT3*-ITD. Inline with this mechanistic hypothesis, we find that when combining the splicing signals and *FLT3*-ITD status, the group of patients that were *FLT3*-ITD negative and assigned to the signal group had much worse responses than patients that were *FLT3*-ITD positive and assigned the background tile (Fig. 5d). Indeed, this latter group of patients has significant enrichment of patients (55/66 patients, p = 8.95e-8) with constitutive *FLT3* canonical isoform expression, defined as inclusion of both events being > 0.9. In contrast, the patients that were *FLT3*-ITD-and in the signal group showed enrichment for intermediate canonical isoform levels (152/159 patients, p = 4.59e-22).

Finally, in another investigation of tile associated splicing changes in AML genes we observed high enrichment of missing values in the background cluster for *EZH2*. The change in

inclusion levels was not particularly large, yet there was over a 15% increased missingness rate of two *EZH2* LSV in the background cluster (Fig. 5e). One of these events corresponds to an event recently reported by Rivera et al. 2021[10] and validated to introduce a PTC that induces NMD and results in reduced protein levels. The other splicing change we identified introduces an un-annotated exon into the highly conversed WD domain of the protein. This suggests there was rapid degradation of the transcript making it more difficult to sequence which in turn resulted in elevated missing values. In summary, our analysis of CHESSBOARD's tile with respect to drug response indicated that the RNA splicing tiles correlate with AML specific drug responses and offer insights into potential underlying mechanisms captured by both changes of Ψ and missing values.

CHESSBOARD Finds More Complex Tile Structures in other Leukemia Datasets

While our analysis focused on adult AML, we also applied CHESSBOARD to several other datasets and disease to demonstrate CHESSBOARD's general utility for splicing pattern discovery. First we applied CHESSBOARD to a joint dataset (samples = 1089, LSV = 2965) consisting of TARGET pediatric AML and beatAML samples (Supplementary Data 5). Studies have show that there are many genetic differences between pediatric and adult AML[41]. However the mutation burden in pediatric AML is lower suggesting that alternative disease causing modalities should be investigated. Specifically, LSVs that are included in tiles that are enriched for samples of a single disease type can be used to distinguish the diseases at the transcriptomic level. On the other hand, LSVs which appear in tiles with mixed sample composition represent splicing variations that are shared between diseases. CHESSBOARD discovered 5 clusters in this dataset. Notably, tiles segregate by disease with C1, C2, and C4 representing pediatric AML and C3 and C5 representing adult AML (Fig. 6a). However a subset of LSVs are unique

to adult (green) and pediatric (blue) AMLs respectively. Other LSVs are either shared between subtypes of each disease type (yellow) or unique to only a single subtype of a disease (purple). Many of these splice variations occur in genes that are commonly differentially mutated between pediatric and adult disease types[42;43].

Next we applied CHESSBOARD to TARGET B-ALL (B-cell Acute Lymphoblastic Leukemia) data (samples = 517,LSVs = 1562), a markedly different type of leukemia characterized by proliferation of lymphoid blasts in the bone marrow (Supplementary Data 6). We recovered 5 clusters with a distinctively more complex tile structure compared to the result on the beatAML dataset (Fig. 6b). Of note, one identified subgroup was enriched for patients which are *RUNX1-ETV6* fusion negative who also have high relapse rates. This mutation is often used as a positive prognostic marker which suggests the splicing signature associated with this tile can be used in a similar manner[44].

Discussion

There is increasing evidence for the pathogenicity of splicing aberrations in heterogeneous cancers such as AML and B-ALL[10;11;45;46], pointing to a need for methods dedicated to unsupervised discovery of splicing based disease subtypes. Here, we develop CHESSBOARD, a method which offers several contributions to the densely populated area of clustering and missing value modeling. Specifically, previous works on tile finding and biclustering approaches were either not domain specific[23;24] or tailored for other data modalities such as gene expression and genetic mutations[22;25;26]. Consequently, these algorithms do not consider crucial characteristics of heterogeneous splicing cancer data such as the uncertainty in splicing quantifications and missing values. We demonstrate here using both synthetic and real data, the usefulness of modeling these data characteristics. Furthermore, CHESSBOARD's MNAR model could also be applicable in domains well beyond RNA splicing or even clustering, for example in algo-

rithms for dimensionality reduction such as sparse probabilistic PCA or factor analysis[47;48].

Beyond the CHESSBOARD model, we also implement several additional algorithms and tools to enable more extensive exploration of the data. First, we developed a prefiltering and recursive clustering method to facilitate analysis of the entire transcriptome. We then used the recursive clustering to discover alternate AML subgroups definitions which strongly correlated with mutations in key AML genes. Second, we implement a LSV ranking system to enable prioritization of driver genes for use in downstream analysis like GSEA. This system is unique in that we can rank LSVs based on differences in Ψ distribution and enrichment of missingess value signals. Finally we implement the CHESSBOARD algorithm and all analytical tools in a Python package. The package is accompanied by an online interactive visualization tool called GAMBIT that enables users to manually inspect the LSVs and samples contained in each tile.

While we applied CHESSBOARD to several leukemia datasets, we focused on beatAML as it offered both a large set of samples and drug response measurements. In beatAML, we found a single strong "signal" tile that divided the dataset to two main subgroups of patients which were highly reproducible in an independent dataset. Investigating possible splicing factors which may form these tiles, we find that *SRSF1* is a key regulatory factor and affects the splicing of 2.49% of the junctions in the tile through direct binding. However it is important to note that taken together all of these RBPs can still only explain 11.75% of the splicing variations in the observed tiles. This arguably low fraction could be due to a myriad of reasons, including the difference between ENCODE's cell lines and tumor specimens, the limited number of RBPs served by ENCODE, and inherent noise in the CLIP and KD assays.

When we investigated possible functional consequences of the two BeatAML subgroups, we found that the genes containing events differentiating the groups were enriched for genes in the mTORC1 pathway. As mTOR is frequently activated in cancer in a manner that affects drug susceptibility[49;50], our clusters might reflect variations in cellular metabolism that could alter

drug susceptibilities in AML samples. Furthermore, we suspect that the signal tile corresponds to a subtype of AML that may be less adverse (see Supplemental Note 7.1 and Supplementary Fig. 7). Inline with this hypothesis, we find that the background tile was characterized by *SRSF1* misregulation which affects several oncogenes including *BIN1* and *CASP9* in the tile.

We demonstrated the utility of CHESSBOARD's recursive clustering by detecting an alternative tile in the BeatAML data which correlated with *FLT3*-ITD, *NPM1*, and *CEBPA* mutations, defined together as normal karyotype AML[36]. The discovery of this known subtype points to the power of recursive clustering. By removing the dominant signal driven by splicing variations caused by misregulation of RBPs/SFs, we enabled further discovery of an alternate tile structure associated with a different AML subtype characterized by weaker splicing signals but a strong mutation signature. We then demonstrated the clinical utility of CHESSBOARD by analyzing correlation of tiles with drug response data. Notably, we found that while mutations were better predictors of drug response than splicing signals, combining the two yielded a better prediction overall, specifically for *FLT3*-ITD and Sorafenib and *NPM1* and Venetoclax. An interesting hypothesis related to these results is that Sorafenib sensitivity may have been reduced by enrichment of the PI3K/mTOR pathway in the signal group as suggested by previous work[51]. Indeed, such a connection between Sorafenib and mTOR pathway has also been observed in hepatocellular carcinoma where treatment with Sorafenib in patients with increased PI3K/mTOR pathway activity results in reduced relapse rates[52;53;54]. A similar effect has recently been observed in AML patients too[55].

There are several limitations in this study which are important to highlight. Specifically, the narrow IC_{50} concentration ranges used in the beatAML experiments limited fitting of sensitivity curves and thus we had to use AUC as a proxy for sensitivity. Furthermore, despite the many advantages of the CHESSBOARD model, we make several modeling assumptions that could be improved upon. For example, CHESSBOARD assumes there is only a single signal distribution.

In many scenarios, there can be multiple sources of heterogeneity that lead to signal distributions that are a mixture of Beta distributions. We note though that in practice, given the noisy nature of splicing and its quantification from limited read counts, we did not find many clear LSV cases in the data used here that would justify the additional complexity beyond a two component mixture model of signal vs. background.

In summary, we developed CHESSBOARD, the first RNA splicing tailored algorithm for signal detection in heterogeneous RNA-seq datasets. We showed its applicability on several leukemia datasets, connecting the splicing tiles discovered to potential regulators, drug response, and known pathways. Although we present a model of splicing, CHESSBOARD can be easily adapted for alternate datatypes such as expression and multi-omics data integration using a multiview model[56]. We also hope the research community will take advantage of the open source code and apply CHESSBOARD to many other analysis tasks in large, heterogeneous cancer datasets, pushing further our understanding of the role of splicing in complex disease.

Methods

Filtering

To enable analysis of large datasets, CHESSBOARD uses a pre-filtering pipeline to select LSVs of interest followed by a recursive clustering procedure. This 2-step process allows the algorithm to analyze the most potentially interesting splicing events at a high resolution by removing noisy events that could potentially confound true signals. The filtering pipeline is detailed below:

- **Remove LSVs that correspond to lowly expressed genes.** We quantified gene expression using Salmon and aggregated transcript level quantifications into gene level quantifications by summing the TPMs. Any LSV corresponding to a gene with TPMs in the

lowest 5th percentile was removed from further analysis.

- **Remove LSVs with high missingness rates.** A LSV is considered quantifiable if at least 10 reads are observed as being mapped to its splice junctions. Any LSV that is not quantifiable in more than 80% of the samples is removed. Note that we allow for such a high missingness rate because the algorithm is designed to handle missing values.

- **Select highly variable LSVs.** For each LSV j in sample i, we compute the variance across all samples $\sigma_j^2 = \sum_i^N \frac{(\Psi_{ij} - \mu_j)^2}{N}$. We construct the empirical CDF of variances and choose a cutoff based on where the graph plateaus. This procedure selects for approximated 1500-2500 events in our datasets.

- **Select for LSVs with a bimodal Ψ distribution.** Intuitively, a mode that tends toward 0 or 1 with low variance is likely to represent a background distribution since most splicing events favor high or low inclusion. A mode with high variance favoring intermediate values is likely to represent an interesting biological signal that could explain disease state. To select for bimodal LSVs, we use the parametric-bootstrap Kolmogorov Smirnov test. Under this test, the null hypothesis H_0 is the data was drawn from a single component beta distribution while the alternate hypothesis H_1 is the data was drawn from multiple beta distributions. The steps for the test are as follows:

  - For each LSV j, fit a beta distribution to the observed Ψ values by obtaining the maximum likelihood estimates of α and β. Since there is no closed for solution for the MLE, we optimize it numerically.

  - Obtain the observed test statistic, the Kolmogorov Smirnov D, using a 1 sample KS test with the observed data and the CDF of $Beta(\hat{\alpha}, \hat{\beta})$

  - Given $\hat{\alpha}$ and $\hat{\beta}$, simulate B bootstrapped datasets.

- For each bootstrapped dataset, estimate $\hat{\alpha}_b$ and $\hat{\beta}_b$ and compute D_b.
- Compute the empirical p-value of the test as the fraction of boostrapped test statics that are greater than the observed test statistic.

We then select all LSVs with $p < 0.05$. This can be interpreted as selecting LSVs that are multimodal with a 5% chance of being a false positive. If a lower proportion of false positives is desired, one could correct the false discovery rate using a procedure such as Benjamini-Hochberg.

Modeling observed splicing events

Consider a data matrix $X_{n \times m}$ with n columns representing patient samples and m rows representing AS events or LSVs. For a given sample i and LSV j, x_{ij} contains the number of junction spanning reads that are mapped to a splice junction of interest while η_{ij} denotes the total number of reads mapped to all junctions in the LSV (e.g two alternative 5' splice sites of an exon). Under CHESSBOARD's formulation, each sample has an unobserved label $\{c_1, c_2, \ldots, c_n\}$ which assigns it to a patients' group or type $k \in \mathcal{Z}^+$. Each such group k is defined by a vector $\boldsymbol{r}_k \in \{0, 1\}^m$ where m is the dimension of the vector. The assignment $r_{jk} = 0$ indicates LSV j is not part of the unique pattern of group k such that observed inclusion levels for this LSV in samples that do not belong to group k follow some (learned) background Ψ distribution. In contrast, $r_{jk} = 1$ indicates an abnormal splicing signal in LSV j across all samples belonging to group k. We thus formulate the generative process for each observed Ψ entry of the data matrix

as

$$
\begin{aligned}
x_{ij} &\sim Binomial(\eta_{ij}, \Psi_{ij}) \\
\Psi_{ij}|c_i = k, r_{jk} &\sim r_{jk} Beta(\mu_{j1}, \kappa_1) + (1 - r_{jk}) Beta(\mu_{j0}, \kappa_0) \\
\mu_{j0} &\sim Beta(\alpha_0, \beta_0) \\
\mu_{j1} &\sim Beta(\alpha_1, \beta_1) \\
\boldsymbol{r}_k &\sim Bernoulli(\boldsymbol{\delta}) \\
\sum_k r_{jk} &\sim Exp(\lambda) \\
c_i &\sim Categorical(\phi) \\
\phi &\sim Dirichlet(\alpha_o / K)
\end{aligned}
\tag{1}
$$

A plate visualization of this model is shown in Supplementary Fig. 8. A table documenting all variables is given in Supplementary Note 6.1. Under this model, the read rate of sample i in LSV j follows a Binomial distribution with a Beta mixture prior over the level of inclusion Ψ_{ij}. This Beta-Binomial model naturally handles uncertainty in Ψ estimates since observations with low read counts will have higher variance. If observation x_{ij} is assigned to the signal in group k as denoted by $c_i = k, r_{jk} = 1$, its likelihood is evaluated using the signal prior distribution $Beta(\mu_{j1}, \kappa_1)$. Likewise, the observation is evaluated using the background prior distribution $Beta(\mu_{j0}, \kappa_0)$ when $r_{jk} = 0$. Notice that we reparameterize the Beta in terms of mean and variance using $\mu_j = \alpha_j/(\alpha_j + \beta_j)$ and $\kappa_j = (\mu_j(1 - \mu_j))/\sigma_j^2$ where the concentration κ is inversely proportional to variance. This reparameterization enables the use of a Beta hyperprior over the mean of each mixture component to capture known biological behavior of alternative splicing. Specifically, normal splicing dynamics have a propensity toward high or low inclusion levels which can be modeled using the Jeffery prior $\alpha_0 = 0.5, \beta_0 = 0.5$ and high concentration κ. Intermediate levels of inclusion modeled by a distribution with a long tail generally indicate aberrant splicing and can be modeled as a Beta distribution with $\alpha_1 = \beta_1$ and low concentration.

To control the number of tiles, we impose a L1 penalty with hyperparameter λ to induce sparsity in the number of groups for which LSV j is assigned to the signal distribution. Specifically, $\sum_k r_{jk} \sim Exp(\lambda)$.

In most biological contexts, the number of tiles is unknown a priori. This can be modeled as an infinite mixture of groups using a Dirichlet Process prior with Bernoulli base distribution

$$\begin{aligned} \Psi_i | c_i = k, \boldsymbol{r} &\sim f(\boldsymbol{r}_k) \\ \boldsymbol{r}_k &\sim G \\ G &\sim DP(G_0, \alpha_o) \\ G_0 &\equiv Bernoulli(\delta)^m \\ c_i &\sim CRP(\alpha_o) \end{aligned} \tag{2}$$

where G_0 is $Bernoulli(\delta)^m$ and α_0 is the concentration parameter. A larger value of α will result in the discovery of more sample groups. In taking the limit of k, the distribution of c_i can be interpreted as a distribution of partitions of natural numbers which is usually formulated as the Chinese Restaurant Process (CRP) or stick breaking process.

Modeling missing values

In gene expression data, missing values typically arise due to low sequencing coverage which results in some transcripts lacking any observable reads even if they are expressed. However in splicing data, a lack of junction spanning reads mapped to a transcript that is expressed indicates inclusion of an alternative exon. Many algorithms handle missing values under the MCAR (missing completely at random) model of missingness by integrating missing values out of the model[27]. Under this model, the missingness rate of a feature does not depend on any observed or unobserved values. However, this is generally only a valid assumption in scenarios when the data generating instrument malfunctions such as a defective microarray probe. The missingness

rate of transcriptomic quantifications from RNA-seq is proportional to sequencing depth thus a model in which values are MNAR (missing not at random) will yield better estimates for the missingness rate. Under this model, the missingness rate of features depends on observations in the data matrix and external factors. In cancer data specifically, values can also be systematically missing due to genetic mutations resulting in no reads being mappable to the splice junction. This can occur when a mutation near a splice site reduces junction usage due to changes in splice factor binding or when the mutation introduces a PTC into an exon resulting in rapid degradation of the transcript due to NMD. Thus it becomes necessary to treat missing values as a secondary signal when the missingness rate of an LSV is much higher than expected under the null missingness rate associated with sequencing limitations. To handle missing values under the MNAR model and detect missing value signals, CHESSBOARD identifies unquantifiable LSVs with $\eta_{ij} < 10$ and replaces x_{ij} with indicator $\omega_{ij} = 1$. The indicator is modeled as

$$\begin{aligned} \omega_{ij}|c_i = k, r_{jk} &\sim r_{jk}Bernoulli(\theta_{j1}) + (1 - r_{jk})Bernoulli(\theta_{j0}) \\ \theta_{j0} &\sim Beta(a_{j0}, b_{j0}) \\ \theta_{j1} &\sim Beta(a_{j1}, b_{j1}) \end{aligned} \tag{3}$$

where θ_0 is the background missingness rate that represents values that are missing due to techinical factors such as coverage and θ_1 represents the signal missingness rate that is expected to be higher and represents values that are missing due to mutations. The priors can be estimated empirically. Specifically, we estimate the background priors by fitting the following Beta-Binomial regression model.

$$\begin{aligned} \upsilon_j &\sim BetaBinomial(n, \mu_j\Phi, (1 - \mu_j)\Phi) \\ logit(\mu_j) &= \beta_0 + \beta_1\chi_j \end{aligned} \tag{4}$$

Here, $\chi_j = median(\{\eta_{1j}, \eta_{2j}, \ldots, \eta_{nj}\})$ is the median number of reads η that are mapped to LSV j across the n samples. υ_j is the number of samples with missing observations for LSV j. The fitted model returns MLE estimates for the coefficients β_0 and β_1 and the dispersion Φ. This trained model can then be used to estimate background priors $\alpha_{j0} = \mu_j\Phi$ and $\beta_{j0} = (1-\mu_j)\Phi$ by predicting μ_j from the median read depths for each LSV j in the cancer data to be analyzed. In this study the above model was fitted to whole blood samples from GTEX V8. Users can of course fit the model to more relevant healthy tissue samples for their specific cancer of interest. Generally, we expat that in healthy control samples the missingness rate will be inversely proportional to sequencing depth (MNAR) but these missing values would not represent signals caused by cancer. In a similar way, we use the same procedure over the training data (beatAML or TARGET) to get estimates for the matching signal prior α_{j1} and β_{j1}.

Posterior Sampling

The entire joint likelihood of the CHESSBOARD model is given by:

$$
\begin{aligned}
&P(\boldsymbol{x}, \boldsymbol{\omega} | \boldsymbol{\eta}, \boldsymbol{c}, \boldsymbol{r}, \boldsymbol{\Psi}, \boldsymbol{\theta}, \boldsymbol{\mu}, \boldsymbol{\kappa}, \boldsymbol{\alpha}, \boldsymbol{\beta}, \lambda) \propto \\
&\prod_{(i,j)\in\{i,j|\forall\omega_{ij}=0\}} P(x_{ij}|\eta_{ij}, \Psi_{ij}) P(\Psi_{ij}|c_i, r_{jc_i}, \mu_{j1}, \mu_{j0}, \kappa_1, \kappa_0) P(\mu_{j1}|\alpha_1, \beta_1) P(\mu_{j0}|\alpha_0, \beta_0) \\
&\prod_{i=1}^{n}\prod_{j=1}^{m} P(\omega_{ij}|c_i, r_{jc_i}, \theta_{j0}, \theta_{j1}) P(\theta_{j0}|, a_{j0}, b_{j0}) P(\theta_{j1}|, a_{j1}, b_{j01}) P(\sum_{k\in\{\boldsymbol{c}\}} r_{jk}|\lambda) \quad (5)
\end{aligned}
$$

where a bold variable indicates a vector containing the variables across all possible indices. To sample from the model's posterior, we develop an efficient blocked Gibbs sampling scheme which we will use to sample from each conditional posterior. The full conditional posterior of

c_i is denoted by

$$P(c_i = k|\boldsymbol{\eta}, \boldsymbol{r}, \boldsymbol{\Psi}, \boldsymbol{\theta}, \boldsymbol{\mu}, \boldsymbol{\kappa}, \boldsymbol{x}, \boldsymbol{\omega}, \boldsymbol{\alpha}, \boldsymbol{\beta}, \lambda) \propto$$
$$\prod_{j \in \{j|\forall \omega_{ij}=0\}} P(x_{ij}|\Psi_{ij}) P(\Psi_{ij}|c_i = k, r_{jk}, \mu_{j1}, \mu_{j0}, \kappa_1, \kappa_0) P(\mu_{j1}|\alpha_1, \beta_1) P(\mu_{j0}|\alpha_0, \beta_0)$$
$$\prod_{j}^{m} P(\omega_{ij}|c_i = k, r_{jk}, \theta_{j0}, \theta_{j1}) P(\theta_{j0}|, a_{j0}, b_{j0}) P(\theta_{j1}|, a_{j1}, b_{j01}) P(\sum_{k \in \{\boldsymbol{c}\}} r_{jk}|\lambda) P(c_i = k) \quad (6)$$

Due to beta-binomial conjugacy, we can integrate out $\boldsymbol{\Psi}$ and $\boldsymbol{\theta}$. This allows us to write

$$P(c_i = k|\boldsymbol{\eta}, \boldsymbol{r}, \boldsymbol{\Psi}, \boldsymbol{\theta}, \boldsymbol{\mu}, \boldsymbol{\kappa}, \boldsymbol{x}, \boldsymbol{\omega}, \boldsymbol{\alpha}, \boldsymbol{\beta}, \lambda) \propto$$
$$\begin{cases} \frac{n_k}{n-1+\alpha_0} \prod_{j=1}^{m} P(x_{ij}|r_{jk}, \Theta) P(\Theta) & \text{if } c_i = k \\ \frac{\alpha_0}{n-1+\alpha_0} \prod_{j=1}^{m} \int P(x_{ij}|r_{j(k+1)}, \Theta) P(\Theta) P(r_{j(k+1)}) dr_j & \text{if } c_i = k+1 \end{cases} \quad (7)$$

Here, the term before the product represents the prior $P(c_i = k)$ which Intuitively captures the cluster's proportion. The term Θ captures all other variables in the above likelihood not explicitly written again (for clarity). Since the integral over each $r_{j(k+1)}$ here is intractable, we follow Neal 2000 and sample vector $\boldsymbol{r}_{(k+1)}$ from its prior distribution when attempting to open a new cluster. We choose the prior to be $\delta_j = P(r_{jk} = 1)$ for each element for the vector j. Note that $\boldsymbol{\omega}$ is not included in the notation above for clarity but is trivial to include. With $\boldsymbol{\Psi}$ and $\boldsymbol{\theta}$ integrated out, we will only need to then explicitly sample $\boldsymbol{r}$, $\boldsymbol{\alpha}$ and $\boldsymbol{\beta}$. The full posterior conditional of r_{jc_i} is given below.

$$P(r_{jc_i} = 1|\boldsymbol{\eta}, \boldsymbol{r}, \boldsymbol{\Psi}, \boldsymbol{\theta}, \boldsymbol{\mu}, \boldsymbol{\kappa}, \boldsymbol{x}, \boldsymbol{\omega}, \boldsymbol{\alpha}, \boldsymbol{\beta}, \lambda) = \frac{P(x_{ij}|r_{jc_i} = 1, \eta_{ij}, \mu_{j1}, \mu_{j0}, \kappa_1, \kappa_0)}{\sum_{r_{jc_i}} P(x_{ij}|r_{jc_i}, \eta_{ij}, \mu_{j1}, \mu_{j0}, \kappa_1, \kappa_0)} \quad (8)$$

However, due to high correlation between the r_{jc_i}s, we must sample them simultaneously. In other words, rather than sampling r_{jc_i} for each $c_i = k$, sample a vector $r_{j\cdot}$ with length k. Note that as k becomes large (i.e. the number of clusters grows in each MCMC iteration), computing this posterior becomes intractable since there are 2^k binary vectors. Therefore, we approximate this posterior by sampling $r_{j\cdot}$ in blocks of $r_{ja}...r_{jb}$ where $b - a$ is the maximum blocksize.

Finally, to sample μ_{j0} and μ_{j1}, we use a discrete approximation. We use possible values of μ on the interval $p \in [0.025, 0.975]$ from 20 discrete bins. Thus we can evaluate the posterior of μ using a discrete categorical distribution defined as:

$$P(\mu_{j0} = p|\boldsymbol{r}, \boldsymbol{\Psi}, \boldsymbol{\theta}, \boldsymbol{\mu}, \boldsymbol{\kappa}, \boldsymbol{x}, \boldsymbol{\omega}, \boldsymbol{\alpha}, \boldsymbol{\beta}, \lambda) = \frac{P(x_{ij}|c_i, r_{jc_i}, \eta_{ij}, \mu_{j1}, \mu_{j0} = p, \kappa_1, \kappa_0)}{\sum_p P(x_{ij}|c_i, r_{jc_i}, \eta_{ij}, \mu_{j1}, \mu_{j0} = p, \kappa_1, \kappa_0)} \quad (9)$$

The concentration κ_0 and κ_1 are hyperparameters that are inversely proportional to the variance of the prior distribution. We choose $\kappa_0 = 20$ and $\kappa_1 = 10$ to model low expected variance of the background and high expected variance of the signal.

Posterior Summary and Convergence

To obtain a point estimate for r_{jk} and c_i, we apply the following posterior summary procedure. First, we obtain the pairwise matrix of probabilities that any two samples clustered together across all posterior samples (after burn-in and thinning). In other words, the probability that $c_i = c_j$ for any two samples i and j. Determining the portion of the chain to discard can be evaluated using the Heidelberger-Welch diagnostic (Supplementary Note 3.2). However in practice, we found for real high dimensional data splicing as we used here that the estimated model parameters converge very quickly and exhibit low posterior variance. In such cases, few MCMC iterations are needed and the optimization can be treated as a variational Bayes approximation (Supplementary Note 3.1). Convergence is then determined to be where the posterior likelihood of the model stops changing. We apply hierarchical clustering to the pairwise probability matrix with the number of clusters k being the median number of clusters across all posterior samples to obtain final clustering assignments. To obtain a point estimate for r, we obtain a matrix of the marginal probabilities that sample i in LSV j is assigned to the signal distribution. $r_{jk} = 1$ if the mean of $r_{jc_i} \forall c_i = k > 0.7$. Note that we also provide an alternative approach to point summary by using the posterior sample that minimizes the MSE to the

posterior mean. In other words, we generate the mean pairwise clustering matrix and pick the sample that minimizes MSE to this mean matrix.

Data Availability

The beatAML dataset can be accessed through the National Cancer Institute (NCI) at `https://www.cancer.gov/about-nci/organization/ccg/blog/2019/beataml`. The Therapeutically Applicable Research to Generate Effective Treatments (TARGET) dataset, phs000218, managed by the NCI can be accessed at `www.ncbi.nlm.nih.gov/projects/gap/cgi-bin/study.cgi?study_id=phs000218.v22.p8`. Information about TARGET can be found at `http://ocg.cancer.gov/programs/target`. The Penn HTSC dataset is available at GEO (GSE142514). The ENCODE knockout and eCLIP datasets from Van Nostrand et al. 2020 are available at `https://www.encodeproject.org`[57]. The GTEx v7 whole blood data is available at `https://www.gtexportal.org/home/datasets`. Source data are provided with this paper. All processed datasets are available in the Zenodo repository associated with this publication. The data generated in this study including algorithm output and data used to figures is described in the Supplementary Information and Source Data files and can be accessed in the Zenodo repository.

Code Availability

All code for the algorithm, Python API and GAMBIT is publically available at `https://bitbucket.org/biociphers/chessboard/src/master/`. A list of Python package dependencies (pandas, scipy, numpy, seaborn, statmodels, scikit-learn, matplotlib) are listed in the installation instructions in the repository and will be automatically installed when installing our software. The GAMBIT tool is available online at `https://paros.pmacs.upenn.edu/gambit/`. Sample data for GAMBIT can be downloaded from the bitbucket

repository. Documentation for the CHESSBOARD python API can be found at `https://chessboard.readthedocs.io/en/latest/index.html`. All code to reproduce figures and analysis can be found in the Zenodo repository.

References

[1] Lee, S.-I. *et al.* A machine learning approach to integrate big data for precision medicine in acute myeloid leukemia. *Nature communications* **9**, 1–13 (2018).

[2] Way, G. P. *et al.* Machine learning detects pan-cancer ras pathway activation in the cancer genome atlas. *Cell reports* **23**, 172–180. e3 (2018).

[3] Huang, C. *et al.* Machine learning predicts individual cancer patient responses to therapeutic drugs with high accuracy. *Scientific reports* **8**, 1–8 (2018).

[4] Shen, R., Olshen, A. B. & Ladanyi, M. Integrative clustering of multiple genomic data types using a joint latent variable model with application to breast and lung cancer subtype analysis. *Bioinformatics* **25**, 2906–2912 (2009).

[5] Robertson, A. G. *et al.* Comprehensive molecular characterization of muscle-invasive bladder cancer. *Cell* **171**, 540–556. e25 (2017).

[6] Choi, P. S. & Thomas-Tikhonenko, A. RNA-binding proteins of COSMIC importance in cancer. *Journal of Clinical Investigation* **131**, e151627 (2021).

[7] Hentze, M. W., Castello, A., Schwarzl, T. & Preiss, T. A brave new world of RNA-binding proteins. *Nature reviews Molecular cell biology* **19**, 327–341 (2018).

[8] Gebauer, F., Schwarzl, T., Valcárcel, J. & Hentze, M. W. RNA-binding proteins in human genetic disease. *Nature Reviews Genetics* **22**, 185–198 (2021).

[9] Alexandrov, L. B. *et al.* Signatures of mutational processes in human cancer. *Nature* **500**, 415–421 (2013).

[10] Rivera, O. D. *et al.* Alternative splicing redefines landscape of commonly mutated genes in acute myeloid leukemia. *Proceedings of the National Academy of Sciences* **118** (2021).

[11] Kim, E. *et al.* SRSF2 mutations contribute to myelodysplasia by mutant-specific effects on exon recognition. *Cancer cell* **27**, 617–630 (2015).

[12] Tyner, J. W. *et al.* Functional genomic landscape of acute myeloid leukaemia. *Nature* **562**, 526–531 (2018).

[13] Cherry, S. & Lynch, K. W. Alternative splicing and cancer: insights, opportunities, and challenges from an expanding view of the transcriptome. *Genes & Development* **34**, 1005–1016 (2020).

[14] Li, B. & Dewey, C. N. RSEM: accurate transcript quantification from RNA-Seq data with or without a reference genome. *BMC bioinformatics* **12**, 1–16 (2011).

[15] Patro, R., Duggal, G., Love, M. I., Irizarry, R. A. & Kingsford, C. Salmon provides fast and bias-aware quantification of transcript expression. *Nature methods* **14**, 417–419 (2017).

[16] Bray, N. L., Pimentel, H., Melsted, P. & Pachter, L. Near-optimal probabilistic RNA-seq quantification. *Nature biotechnology* **34**, 525–527 (2016).

[17] Katz, Y., Wang, E. T., Airoldi, E. M. & Burge, C. B. Analysis and design of RNA sequencing experiments for identifying isoform regulation. *Nature methods* **7**, 1009–1015 (2010).

[18] Tiberi, S. & Robinson, M. D. BANDITS: Bayesian differential splicing accounting for sample-to-sample variability and mapping uncertainty. *Genome biology* **21**, 1–13 (2020).

[19] Vaquero-Garcia, J. *et al.* A new view of transcriptome complexity and regulation through the lens of local splicing variations. *elife* **5**, e11752 (2016).

[20] Trincado, J. L. *et al.* SUPPA2: fast, accurate, and uncertainty-aware differential splicing analysis across multiple conditions. *Genome biology* **19**, 1–11 (2018).

[21] Shen, S. *et al.* rMATS: robust and flexible detection of differential alternative splicing from replicate RNA-Seq data. *Proceedings of the National Academy of Sciences* **111**, E5593–E5601 (2014).

[22] Gu, J. & Liu, J. S. Bayesian biclustering of gene expression data. *BMC genomics* **9**, 1–10 (2008).

[23] Meeds, E. & Roweis, S. Nonparametric bayesian biclustering. Tech. Rep., Citeseer (2007).

[24] Givoni, I., Cheung, V. & Frey, B. J. Matrix tile analysis. *arXiv preprint arXiv:1206.6833* (2012).

[25] Gao, C., McDowell, I. C., Zhao, S., Brown, C. D. & Engelhardt, B. E. Context specific and differential gene co-expression networks via Bayesian biclustering. *PLoS computational biology* **12**, e1004791 (2016).

[26] Xu, Y. *et al.* Nonparametric bayesian bi-clustering for next generation sequencing count data. *Bayesian analysis (Online)* **8**, 759 (2013).

[27] Rubin, D. B. Inference and missing data. *Biometrika* **63**, 581–592 (1976).

[28] Dhillon, I. S. Co-clustering documents and words using bipartite spectral graph partitioning. In *Proceedings of the seventh ACM SIGKDD international conference on Knowledge discovery and data mining*, 269–274 (2001).

[29] Slaff, B. *et al.* MOCCASIN: A method for correcting for known and unknown confounders in RNA splicing analysis. *Nature Communications* **12**, 1–9 (2021).

[30] de Necochea-Campion, R., Shouse, G. P., Zhou, Q., Mirshahidi, S. & Chen, C.-S. Aberrant splicing and drug resistance in AML. *Journal of hematology & oncology* **9**, 1–9 (2016).

[31] Anande, G. *et al.* RNA splicing alterations induce a cellular stress response associated with poor prognosis in AML. *bioRxiv* (2020).

[32] Das, S. & Krainer, A. R. Emerging functions of SRSF1, splicing factor and oncoprotein, in RNA metabolism and cancer. *Molecular Cancer Research* **12**, 1195–1204 (2014).

[33] Anczuków, O. *et al.* The splicing factor SRSF1 regulates apoptosis and proliferation to promote mammary epithelial cell transformation. *Nature structural & molecular biology* **19**, 220–228 (2012).

[34] Massiello, A. & Chalfant, C. E. SRp30a (ASF/SF2) regulates the alternative splicing of caspase-9 pre-mRNA and is required for ceramide-responsiveness. *Journal of lipid research* **47**, 892–897 (2006).

[35] Reiling, J. H. & Sabatini, D. M. Stress and mTORture signaling. *Oncogene* **25**, 6373–6383 (2006).

[36] Port, M. *et al.* Prognostic significance of FLT3 internal tandem duplication, nucleophosmin 1, and CEBPA gene mutations for acute myeloid leukemia patients with normal kary-

otype and younger than 60 years: a systematic review and meta-analysis. *Annals of hematology* **93**, 1279–1286 (2014).

[37] Network, C. G. A. R. Genomic and epigenomic landscapes of adult de novo acute myeloid leukemia. *New England Journal of Medicine* **368**, 2059–2074 (2013).

[38] Wojtuszkiewicz, A. *et al.* Maturation State-Specific Alternative Splicing in FLT3-ITD and NPM1 Mutated AML. *Cancers* **13**, 3929 (2021).

[39] van der Werf, I. *et al.* Splicing factor gene mutations in acute myeloid leukemia offer additive value if incorporated in current risk classification. *Blood Advances* **5**, 3254–3265 (2021).

[40] Steelman, L. S. *et al.* Contributions of the Raf/MEK/ERK, PI3K/PTEN/Akt/mTOR and Jak/STAT pathways to leukemia. *Leukemia* **22**, 686–707 (2008).

[41] Bolouri, H. *et al.* The molecular landscape of pediatric acute myeloid leukemia reveals recurrent structural alterations and age-specific mutational interactions. *Nature medicine* **24**, 103–112 (2018).

[42] Gröbner, S. N. *et al.* The landscape of genomic alterations across childhood cancers. *Nature* **555**, 321–327 (2018).

[43] Ma, X. *et al.* Pan-cancer genome and transcriptome analyses of 1,699 paediatric leukaemias and solid tumours. *Nature* **555**, 371–376 (2018).

[44] Krentz, S. *et al.* Prognostic value of genetic alterations in children with first bone marrow relapse of childhood B-cell precursor acute lymphoblastic leukemia. *Leukemia* **27**, 295–304 (2013).

[45] Black, K. L. *et al.* Aberrant splicing in B-cell acute lymphoblastic leukemia. *Nucleic acids research* **46**, 11357–11369 (2018).

[46] Sotillo, E. *et al.* Convergence of acquired mutations and alternative splicing of CD19 enables resistance to CART-19 immunotherapy. *Cancer discovery* **5**, 1282–1295 (2015).

[47] Pierson, E. & Yau, C. ZIFA: Dimensionality reduction for zero-inflated single-cell gene expression analysis. *Genome biology* **16**, 1–10 (2015). ISBN: 1474-760X Publisher: BioMed Central.

[48] Argelaguet, R. *et al.* MOFA+: a statistical framework for comprehensive integration of multi-modal single-cell data. *Genome biology* **21**, 1–17 (2020).

[49] Mossmann, D., Park, S. & Hall, M. N. mTOR signalling and cellular metabolism are mutual determinants in cancer. *Nature Reviews Cancer* **18**, 744–757 (2018).

[50] Jiang, B.-H. & Liu, L.-Z. Role of mTOR in anticancer drug resistance: perspectives for improved drug treatment. *Drug resistance updates* **11**, 63–76 (2008).

[51] Lindblad, O. *et al.* Aberrant activation of the PI3K/mTOR pathway promotes resistance to sorafenib in AML. *Oncogene* **35**, 5119–5131 (2016).

[52] Yi, H. *et al.* Inhibition of the AKT/mTOR pathway augments the anticancer effects of sorafenib in thyroid cancer. *Cancer biotherapy & radiopharmaceuticals* **32**, 176–183 (2017).

[53] Kim, M. N., Lee, S. M., Kim, J. S. & Hwang, S. G. Preclinical efficacy of a novel dual PI3K/mTOR inhibitor, CMG002, alone and in combination with sorafenib in hepatocellular carcinoma. *Cancer chemotherapy and pharmacology* **84**, 809–817 (2019).

[54] Gedaly, R. *et al.* The role of PI3K/mTOR inhibition in combination with sorafenib in hepatocellular carcinoma treatment. *Anticancer research* **32**, 2531–2536 (2012).

[55] Damnernsawad, A. *et al.* Genome-wide CRISPR screen identifies regulators of MAPK and MTOR pathways mediating sorafenib resistance in acute myeloid leukemia. *Haematologica* (2020).

[56] Burdziak, C., Azizi, E., Prabhakaran, S. & Pe'er, D. A nonparametric multi-view model for estimating cell type-specific gene regulatory networks. *arXiv preprint arXiv:1902.08138* (2019).

[57] Van Nostrand, E. L. *et al.* A large-scale binding and functional map of human RNA-binding proteins. *Nature* **583**, 711–719 (2020).

Acknowledgements: We would like to thank Matthew Gazzara, Dr. Martin Carroll and Dr. Sara Cherry for their input on the data analysis interpretation, Dr. Caleb Radens for assistance with data processing, and Joseph Aicher for early input on the model/algorithm.

This work was supported by NIH grants R01 GM128096 (Y.B.) and U01 CA232563/CA232563-01S3 (Y.B., A.T.T, and K.W.L.) and grants from CureSearch For Children's Cancer (A.T.T. and Y.B.) and Emerson Collective (A.T.T., Project 886246066). DW is supported by NIH fellowship 1F31CA265218-01.

Author Contributions: YB conceived the project. DW and YB developed the methods and planned the experiments. DW wrote the code and carried out the experiments and analysis. MQV performed the gene ontology analysis. PJ and ME developed the GAMBIT tool. MQV, KL and ATT provided feedback on data analysis interpretation. DW and YB wrote the manuscript with input from MQV, KL and ATT. All authors read and approved the final manuscript.

Competing Interests: The authors declare that they have no competing interests.

Figures

Fig. 1: CHESSBOARD Pipeline. **(A)** Input: splice junction read counts (red and blue reads) extracted from patients' RNA sequencing. Each row in the input data matrix is a LSV (e.g. cassette exon shown) and each rubric contains the junction spanning read counts for that LSV in a specific sample. Complex LSV involving more than two junctions the most variable junction is selected (Methods). **(B)** Task: CHESSBOARD's objective is to identify latent tiles in the input matrix. A tile consists of a subset of samples and a subset of LSVs where the Ψ distribution of each LSV for samples within the tile differs from the background distribution. Note that the matrices shown contain Ψ values for visualization purposes but CHESSBOARD acts on the matrix described in (a) and it may not be possible to embed each tile as a continuous square in a 2D image as shown here. **(C)** CHESSBOARD Pipeline: The pipeline includes three steps. Filtering: Lowly expressed genes (lower 5% by default) are removed and LSVs observed in too few samples (default 20%), retaining only those exhibiting high Ψ variability between samples and multiple modes in the Ψ value distribution (Methods). MCMC: Blocked Gibbs sampling based on CHESSBOARD's model and the input data matrix yields posterior samples for potential tile configurations. Intuitively, the algorithm iterates through a chain of solutions that tend towards higher likelihood while varying the number of tiles using the Chinese Restaurant Process (Methods). Analysis: The MC samples are summarized into marginal posterior distributions and possible point estimates for tiles. Tile analysis includes sample assignment to subgroups, LSV assignment to a signal tile, the $\Delta\Psi$ and missingness rate associated with a particular LSV in a tile (Methods). Visualization and analysis are conducted using the accompanying visualization package, GAMBIT.

Fig. 2: Model Evaluation. **(A)** Error in Ψ variance estimation under Gaussian (red), Beta

(blue), and Beta-Binomial (green) models as a function of LSV coverage (x-axis). Absolute error in Ψ variance estimates (y axis) is compared to the true variance, assuming a Beta(10,90) distribution. Inset histograms show empirical distributions of LSV coverage in beatAML and TARGET data. **(B)** Error in $\hat{\Psi}$ quantification estimates under a naive and empirical shrinkage model as a function of read coverage (x-axis, 1000 samples from the same Beta as above for each point). Naive approach uses only read ratios to estimate $\hat{\Psi}$ while shrinkage model uses the expectation over the posterior for the Beta. Error bars represent the 90% confidence interval for the error in Ψ. **(C)** Correlation between Ψ and $\hat{\Psi}$ estimates under a naive (left) and empirical shrinkage model (left). Ψ was sampled as in (A) while number of reads n was sampled randomly from [10,500]. **(D)** Information gained (Supplementary Note 2.2) from missing signals. Here a background matrix was used, consisting of 100 samples and 100 LSVs with a fixed missingness rate of 10%, into which a signal tile was implanted. The signal tile consisted of 50 samples and a varying number of LSVs (x-axis) with an elevated missingness rate of 60%. The observed values in both tile and background were drawn from the same distribution. Green represents the CHESSBOARD model (MNAR), red represents a missing completely at random (MCAR) version of CHESSBOARD. As a reference, we also plot (grey) the information gain from a similarly sized signal tile where the signal is based on a significantly different Ψ distribution simulated with parameters estimated from real data (Supplementary Note 2.3). Missing signals (green) contribute to increase in information gain as the number of missing signals increases. **(E)** Evaluation of CHESSBOARD's (top right) performance on synthetic data, sampled to mimic BeatAML, compared to hierarchical clustering (bottom left) and spectral co-clustering (bottom right). Ψ values are represented as heatmap, sample groups as colored bars and tiles as red rectangles. Note that tiles may appear permuted. Performance was evaluated using a modified version of recovery relevance score (Supplementary Note 2.2) which are permutation invariant.

Fig. 3: BeatAML Dataset Analysis. **(A)** Heatmap showing the tile discovered by CHESSBOARD on the beatAML dataset (samples = 477, LSVs = 2299). The signal (samples = 217, LSVs = 1910) is outlined in red. Note that although CHESSBOARD was run on junction spanning read rates as input, the heatmap shows Ψ values to facilitate visualization. **(B)** Heatmap of Ψ values in the Penn HTSC dataset (samples = 77, LSVs = 2299), showing reproducibility of the tiles originally identified by CHESSBOARD in the beatAML dataset. The signal tile (samples = 32, LSVs = 1899) is outlined in red. **(C)** Correlation between $median(\Delta\Psi)$ in the beatAML and HTSC datasets for the representative junction in each LSVs belonging to the tile. The $median(\Delta\Psi)$ value was computed between the 2 groups discovered by CHESSBOARD in both datasets. Correlation is measured using Pearson's correlation coefficient (r) and the two-sided p-value is the probability of observing a coefficient $> |r|$ under the exact null distribution. **(D)** ENCODE based analysis of possible tile regulators. Top bar plot shows the percentage of splice junctions (y-axis) in the tile that overlap with splice junctions in one of three categories associated with each RBP/SF (x-axis). DS (blue) is the set of junctions that are differentially spliced between case-control samples in ENCODE K562 cell lines. CLIP (orange) is the set of junctions that are bound by the RBP in a 250bp region flanking the junction. The "Both" bar (green) represents junctions in the intersection of DS and CLIP sets. The bottom barplot shows whether the overlap is significant (bonferroni corrected cutoff) based on a one-sided fisher's exact test for enrichment. The red circles indicate whether the matching RBP/SF is differentially spliced (in at least one junction) and/or differentially expressed in the tile's samples and whether it is a component of the spliceosome or a cis/trans acting splice factor. **(E)** mTORC1 GSEA: Enrichment of genes ranked by log(likelihood gain) of LSVs among the HALLMARK_MTORC1_SIGNALING gene set as performed with GSEA v. 4.1.0 and visualized with the fgsea R package.

Fig. 4: Recursive Clustering Analysis. **(A)** Heatmap showing CHESSBOARD clustering results after the first recursive step. A single tile (samples = 196, LSVs = 389) was identified. The tracks above the heatmaps indicate whether a patient was positive (red) or negative (blue) for each mutation. Missing annotations are marked by white. The p-values were computed using fisher's exact test for enrichment and corrected for multiple testing using the min-p method to account for missing annotations. **(B)** Boxplots showing the likelihood ratio distributions of LSVs after each recursive step. Each boxplot represents a recursive step with 0 being the base case. Within a boxplot, each data point (0:LSVs=2299, 1:LSVs=389, 2:LSVs=330, 3:LSVs=319) represent the log likelihood ratio of a LSV under the tile model (learned by CHESSBOARD on the original data) and a null model (learned by CHESSBOARD on the original data with tile structures removed by randomly permuting each row of the data matrix). The median is denoted by a red line, the upper and lower quartiles are denoted by the box, the whiskers denote points that lie within 1.5 IQRs of the lower and upper quartile, and observations that fall outside this range are outliers which are independently displayed.

Fig. 5: AML Drug Response Analysis. **(A)** Heatmap showing the tile discovered by CHESSBOARD in LSVs from 70 AML related genes (samples = 477, LSVs = 90). The signal (samples = 214, LSVs = 66) is outlined in red. Top "Genome Wide Clustering" track shows sample grouping in Fig. 3a. **(B)** Barplot showing for each categorical variable (mutation presence or splicing cluster assignment, left) the drug (right) with the maximum AUC variance (x-axis) explained by the corresponding variable. **(C)** The proposed decision tree for administering Sorafenib based on splicing patterns and mutations. Patients with *FLT3*-ITD- and signal group splicing pattern exhibit a worse response (low AUC) compared to patients with *FLT3*-ITD+ and a background group splicing pattern(high AUC). **(D)** Violin Plots of AUC values for patients' response to Sorafenib when split according to the groups indicated on the x-axis. When com-

bining both splicing and mutations information using the decision tree in Fig. 5c, the variance explained increases to 36.8%. The bars at the top indicate the total number of samples that fall into each category. Notably, the groups exhibiting favorable drug response (*FLT3*-ITD+ & Background) are enriched for abnormal splicing (55/66 patients) while the group with poor response (*FLT3*-ITD- & Signal) are enriched for normal splicing (152/169). Here, abnormal splicing is defined as constitutive expression of the canonical isoform with $\Psi 1 > 0.9$ and $\Psi 2 > 0.9$. **(E)** Differential splicing events in *FLT3* and *EZH2* between the subgroups. For *FLT3*, the inclusion of exon 4b in LSV1 and exon 17b in LSV2 results in introduction of a frameshift or PTC respectively. Scatterplot (bottom left) shows correlation between Ψ values for the skipping event in *FLT3* ($\Psi 1$ for LSV1, $\Psi 2$ for LSV2), while correlation plots (bottom middle and right) show Pearson's correlation between Ψ and *FLT3* expression. The red line indicates the linear regression fit and the band represents the 95% confidence interval. For *EZH2*, the $\Delta\Psi$ values between the clusters for these deleterious events in *EZH2* are low (< 0.2), but are part of a change involving a higher rate of missingness in the background cluster (>0.15).

Fig. 6: Pediatric AML and B-ALL Analysis. (A) Heatmap showing the tile discovered by CHESSBOARD when applied the joint beatAML and TARGET pediatric AML dataset (samples = 1089, LSV = 2965). Tiles are outlined in red. Track on the y-axis groups the LSVs into groups defined as: unique to pediatric (blue), shared between diseases (red), unique to adult (green), unique to one subtype in each disease (yellow), unique to only 1 disease disease and subtype (purple). **(B)** Heatmap showing the tile discovered by CHESSBOARD when applied the TARGET B-ALL dataset (samples = 517,LSVs = 1562). Tiles are outlined in red. The top track indicates whether the patient is positive (blue) or negative (red) of *RUNX1-ETV6* fusion. The second track indicates where the sample is primary (blue) or relapse (red).

A Bayesian model for unsupervised detection of RNA splicing based subtypes in cancers

David Wang,[1,2] Mathieu Quesnel-Vallieres,[1,3] Paul Jewell,[1] Moein Elzubeir,[1] Kristen Lynch,[1,3] Andrei Thomas-Tikhonenko,[4,5] Yoseph Barash[1,6*]

[1]Department of Genetics, Perelman School of Medicine, University of Pennsylvania
[2]Graduate Group in Genomics and Computational Biology, Perelman School of Medicine, University of Pennsylvania
[3]Department of Biochemistry and Biophysics, Perelman School of Medicine, University of Pennsylvania
[4]Department of Pathology and Laboratory Medicine, Perelman School of Medicine, University of Pennsylvania
[5]Division of Cancer Pathobiology, Children's Hospital of Philadelphia
[6]Department of Computer and Information Sciences, School of Engineering, University of Pennsylvania

*To whom correspondence should be addressed; E-mail: yosephb@upenn.edu.

Contents

Supplementary Notes

1 Data Description

Supplementary Note 1.1: Datasets

beatAML Dataset

The beatAML data used in this study includes RNA-seq data from 451 specimens from 411 patients from Tyner et al. 2018 and an additional 26 samples for a total of 477 samples[1]. The samples were sequenced from purified mononuclear cells collected from peripheral blood or bone marrow. All patients were diagnosed with AML or closely related disease. We downloaded FASTQ files from www.synapse.org to use in subsequent processing and analysis. Sequencing adaptors and low quality base calls were trimmed from FASTQs using trim galore. For expression data, we obtained TPM values using SALMON with Hg38 decoys from the FASTQ files. For splicing data, we aligned the FASTQs using STAR and sorted the BAM files using samtools. We mapped reads in the BAM files to splice junctions using MAJIQ using ensembl GRCh38 v94 annotations. The build contained all the beatAML, TARGET pediatric AML and TARGET B-ALL samples such that all 3 datasets use the same splice graph.

beatAML Drug Sensitivity Data

The drug sensitivity data was taken directly from the supplied beatAML metadata. The IC_{50} values were generated using an ex vivo drug sensitivity assay described in Tyner el. al 2018[1]. The screen applied 122 small molecule inhibitors to isolated mononuclear cells from the AML patient samples. The IC_{50} value for each drug-sample pair was calculated by fitting a sigmoid curve to 7 data points representing cell viability at varying drug concentrations and estimating the concentration that resulted in 50% viability. Note that the drug concentrations were measured using 3-fold serial dilution in the range of 10uM to 0.0137 uM. Many drugs-

sample pairs had an IC_{50} of 10uM indicating that a wider range of concentrations were needed to properly fit the curve. In such cases, measuring the area under the curve (AUC) provides a better representation of drug sensitivity.

beatAML Mutation Data

All mutation annotations for beatAML samples used in this study were taken directly from the supplied metadata. The consensus variant calls were generated from multiple genotype callers applied to whole exome sequencing (WES) data and assigned a mutation status based on ensembl VEP GRCh37 annotations as described in Tyner et. al 2018[1]. *FLT3*-ITD, *NPM1* and *CEBPA* mutations for a subset of patient sample were verified using additional experimental assays.

Penn HTSC Dataset

The Penn HTSC dataset contains 77 AML patient samples sequenced by the University of Pennsylvania high-throughput screening core. All samples used in the sequencing were confirmed to be at least 90% AML blasts. A subset of the samples (29) was previously published in Rivera et al. 2021[2]. This data can be obtained from GEO (GSE142514).

ENCODE Knockout Dataset

The ENCODE knockout dataset was taken from Van Nostrand et al. 2020 and processed as described in Slaff et al. 2021[3;4]. The subset of data we used in this study was limited to knockout experiments of 106 RBPs/SFs in K562 cell lines that had matching eCLIP data. This data was generated in 32 batches with each batch containing at least 2 replicates for controls and knockout experiments. The batch effect was corrected across the knockout experiments using MOCCASIN as described in Slaff et al. 2021[4]. For differential splicing analysis, controls across the batches were aggregated into "virtual controls" and compared against each knockout experiment using MAJIQ.

ENCODE eCLIP Dataset

The data was downloaded from `www.encodeproject.org`. We only used data for which there was a matching RBP/SF knockout experiment (106 RBPs/SFs) in K562 cell lines. Each RBP/SF experiment had 2 replicates. Consensus binding peak calls were obtained using irreproducible discovery rate (IDR) (`https://www.encodeproject.org/data-standards/terms/#concordance`).

TARGET B-ALL

The TARGET B-ALL data used in this study includes 517 RNA-seq samples from 250 unique patients with ALL diagnosis. Most patients are represented by samples taken at primary leukemia diagnosis and samples taken after relapse with 2-3 replicates. Only samples with annotated B-cell origin or inferred B-cell origin were considered. The inferred cell origin labels were taken from Slaff et al. 2021 which annotated the cell origin of unannotated samples based on the cell origin of the samples they clustered with[4]. Batch effects were corrected using MOCCASIN for sequencing instrument (HiSeq 2500 vs HiSeq 2000).

TARGET Pediatric AML

The TARGET Pediatric AML data was downloaded from https://portal.gdc.cancer.gov. We selected only samples with an age under 23. The curated dataset contained 612 samples.

Supplementary Note 1.2: Splicing Event Definitions

The input to CHESSBOARD is a data matrix $X_{n\times m}$ with n columns representing patient samples and m rows representing splicing events. The definition of what constitutes an alternative splicing event may vary depending on the quantification tool the chosen by the user. Regardless of the tool, a user can supply two TSV files where each row is a splicing event. The entries in the first file represent reads mapped to the junction of interest while the entries in the second file represent the sum of all reads not mapped to this junction of interest but are still contained in the splicing event/normalization unit. For example, if a user chooses to quantify 'classical'

events such as cassette exons using a method such as rMATS[5], they can parse rMATS output such that the input files reports inclusion reads vs the total exclusion reads per AS event. In terms of CHESSBOARD's model, the reads mapped to a cassette exon's inclusion junction are the successes x_{ij} and the total reads mapped to the AS event (in this example exclusion reads plus inclusion reads) are η_{ij}.

In this study, we use MAJIQ[6] which defines AS events using the concept of local splice variations (LSVs). Briefly, Each LSV has a reference exon to which other exons (or introns, for intron retention events) are spliced to. A source LSV contains a set of splice junctions downstream of the reference exon while a target LSV contains a reference exon that is spliced to other exons or introns upstream of it. MAJIQ's LSV formulation is able to capture classic event types such as cassette exons, but also many other splicing variations that are more complex (involve more that two alternative junctions) as well as unannotated junctions and exons. Within an LSV, a splice junction is quantified by percent splice in (Ψ) which is the ratio of reads spanning the junction to all other reads in the LSV. However, CHESSBOARD's binomial model is designed to handle one junction per splicing event. Thus during the processing of MAJIQ's output, we select as a representative junction per LSV. This junction is determined as the one in the LSV with the highest variance in Ψ across all samples. We note that CHESSBOARD is able to support any representation of splicing as input as long as it can be expressed as ratios of reads. This includes isoform ratios although this is not recommended due to complications with resolving isoform abundance using only short read RNA sequencing.

2 Additional Methods and Evaluation

Supplementary Note 2.1: Parametric Bootstrap Kolmogorov-Smirnov Test

Part of CHESSBOARD's prefiltering pipeline involves removing non-informative LSVs which only exhibit a single Ψ modality using a parametric bootstrap Kolmogorov-Smirnov (PBKS)

test. CHESSBOARD assumes that the Ψ distribution of each LSV is a mixture of 2 Beta distributions representing a signal and background component thus LSVs that can be modeled by a single distribution are unlikely to contain any signal with notable effect size. The null hypothesis (H0) of the PBKS test is that the data is Beta distributed. The alternate hypothesis (H1) is that the data is not Beta distributed. Rejecting H0 would suggest that a single component Beta is a poor fit for the data and thus multiple modalities exist. We first fit a Beta distribution to the Ψ values for a given LSV using MLE. We then compute the 2 sided 1 sample KS statistic D* using the fitted Beta distribution as the reference

$$D* = max(\hat{F}(x) - G(x)) \tag{1}$$

$\hat{F}$ is the empirical CDF and G is the CDF of the fitted Beta. Note that because, the reference distribution was obtained from the data, D* no longer follows the Kolmogorov distribution. Thus we used a parametric bootstrap approach to estimate the p-value. First, we sample n datasets of equal size to the empirical data from the fitted Beta distribution where n is the number of user defined bootstrap samples (default = 500). We then compute D* on each dataset. The PBKS test p-value is equal to the proportion of bootstrapped D* statistics that are greater than the observed D* statistic.

Supplementary Note 2.2: Statistical Methodology for Model Evaluations

Information Gain

Information gain (IG) can be interpreted as the amount of information gained by a random variable given that a dependent variable is observed. The quantity is related to Kullback-Leibler divergence (KLD) in that IG = 0 when the KLD between the joint distribution and product of marginals = 0 (i.e. variables are independent). In a k = 2 clustering setting, IG can be interpreted as the amount of information gained by performing/observing a split of the population into two

groups. We compute IG as follows

$$1 - \sum_{k}^{2} -\frac{I(\hat{c}_i = k)}{N}[P(c_i = 2|\hat{c}_i = k)log_2(P(c_i = 2|\hat{c}_i = k)) + P(c_i = 1|\hat{c}_i = k)log_2(P(c_i = 1|\hat{c}_i = k))] \quad (2)$$

The population entropy in our experimental setting is 1 since the 2 classes have equal size. The second term represents a weighted average of the clustering entropy given observation of an inferred binary label for each sample $\hat{c}_i$.

Precision and Recall Based on Recovery and Relevance Score Precision τ_{pr} and recall τ_{rc} metrics based on Recovery Relevancy Score are given by

$$\tau_{pr} = \frac{\sum_{(G_1,C_1)\in M_1} max_{(G_2,C_2)} \frac{|G_1 \cap G_2| + |C_1 \cap C_2|}{|G_1| + |C_1|}}{|M_1|}$$
$$\tau_{rc} = \frac{\sum_{(G_1,C_1)\in M_1} max_{(G_2,C_2)} \frac{|G_1 \cap G_2| + |C_1 \cap C_2|}{|G_2| + |C_2|}}{|M_1|} \quad (3)$$

Here, M_1 represents a set of reference tiles and each tile is defined by a feature set G_1 and sample set C_1. M_2 represents the set of inferred tiles where each tile is defined by a feature set G_2 and sample set C_2. For each tile in the reference set, select 1 tile in the inferred set which best represents this tile based on the maximum precision or recall of the inferred feature and sample sets. Compute this quantity of each reference tile and average the quantities.

Adjusted Rand Index

Adjusted Rand Index is a measure of clustering agreement. A value of 1 indicates perfect agreement. The score is computed as

$$ARI = \frac{\sum_{ij}(\binom{n_{ij}}{2} - [\sum_i \binom{a_i}{2} \sum_j \binom{b_j}{2})]/\binom{n}{2}}{\frac{1}{2}[\sum_i \binom{a_i}{2} + \sum_j \binom{b_j}{2})] - [\sum_i \binom{a_i}{2} \sum_j \binom{b_j}{2})]} \quad (4)$$

Here, n_{ij} represents the number of samples assigned to group i in the first clustering and group j in the second clustering. $a_i = \sum_j n_{ij}$ and $b_j = \sum_i n_{ij}$.

Effective Dimensionality

Effective dimensionality represents the strength of the concentration parameter as a function of the number of features. Intuitively, as the number of features increases, the impact of the concentration parameter decreases since the total likelihood of a sample compounds multiplicatively with the number of features. Therefore a smaller concentration is needed as the feature space increases to maintain constant effect. We define effective dimensionality as

$$\alpha_d = \frac{\alpha}{N - 1 + \alpha}^{d} \tag{5}$$

where α_d is the concentration parameter chosen to have the same effect as α when the effective feature set size is d and N is the number of samples. This reduces to the standard CRP prior for opening a new cluster in a 1 dimensional setting (i.e d = 1).

Supplementary Note 2.3: Generative Synthetic Data Simulations

To generate synthetic data for testing our model, we first simulated a background matrix with 100 samples (columns) and 100 LSVs (rows). For each LSV j and sample i, we simulated the read rate x_{ij} from the following generative process:

$$\begin{aligned} x_{ij} &\sim Binomial(n_j, \Psi_j^{BG}) \\ \Psi^{BG} &\sim Beta(\alpha_j^{BG}, \beta_j^{BG}) \\ n_j &\sim Poisson(\lambda_j) \end{aligned} \tag{6}$$

We then implanted tiles with read rates simulated from the following generative process:

$$\begin{aligned} x_{ij} &\sim Binomial(n_j, \Psi_j^{S}) \\ \Psi^{S} &\sim Beta(\alpha_j^{S}, \beta_j^{S}) \\ n_j &\sim Poisson(\lambda_j) \end{aligned} \tag{7}$$

Here $Beta(\alpha_j^{BG}, \beta_j^{BG})$ represents the background Ψ distribution, $Beta(\alpha_j^{S}, \beta_j^{S})$ represents the signal Ψ distribution and $Poisson(\lambda_j)$ represents the distribution over read coverage or expres-

sion level for a LSV. Poisson modeling for gene expression has been used in several methods because read dispersion tends to increase with expression levels. The parameters of the simulation were estimated from the beatAML dataset. To estimate α and β, we fit a 2 component Beta mixture to the Ψ values of the 1000 most variable LSVs. The estimated parameters for the component with lower variance were used for the background. To estimate λ, we use the median read rate for each LSV. We selected a random subset of 100 LSVs with $|E(\Psi^{BG}) - E(\Psi^{S})| > 0.2$ to parameterize each LSV in the simulation.

Supplementary Note 2.4: Infinite Mixture Behavior Evaluation

CHESSBOARD is able to learn the number of tiles without a priori knowledge using an infinite mixture modeling approach with a Chinese Restaurant Process (CRP) prior (Methods). This can be particularly effective for analyzing cancer data where the number and size of disease subtypes is unknown. Specifically, CHESSBOARD naturally models both common and rare subtypes since the CRP prior expects a gradient of group sizes. However, like other unsupervised methods, CHESSBOARD depends on both hyperparmeters and characteristics of the data. Here, we show how the CRP concentration hyperparameter, which regularizes the number of tiles, interacts with two characteristics of the data: The number of LSVs supporting a tile and the signal strength within an LSV measured by Kullback Leibler (KL) divergence between the signal and background distributions. In addition, we evaluate the algorithm's ability to identify tiles under two different scenarios: One where it must assign samples to the correct tile such as when the clusters are initialized using k-means and another when it must overcome the concentration parameter to create a new tile. The results of these evaluations are summarized in Fig. 2e. As expected, we find that as the distance between the signal and background distributions increases, fewer supporting LSVs are needed to assign a sample to the tile with high (> 0.99) probability, ranging from 1 for KL = 6.697 to 10 for KL = 0.617 (Supplementary Fig.

1a left). We also find that learning new clusters requires far more supporting LSVs or a much stronger signal-background discrepancy due to the penalty of opening a new cluster induced by the concentration parameter. By setting the concentration parameter using effective dimensionality (Supplementary Note 2.2), we show that it takes at least 20 supporting LSVs to find a new cluster when the KL divergence is high (Supplementary Fig. 1a right). Finally, we evaluate whether these observations hold on more realistic synthetic data by simulating a matrix with 2 tiles where the average $\Delta\Psi$ between signal and background distributions is 0.2, a threshold commonly used in the RNA field to define a significant splicing change. We then add a 3rd tile with a varying number of supporting LSVs. We find such data requires at least 12 supporting LSVs to find the 3rd tile (Supplementary Fig. 1b).

Supplementary Note 2.5: Runtime and Memory Evaluation

To assess the runtime and memory usage of CHESSBOARD, we ran the algorithm on synthetic data constructed with a varying number of samples n and features m and k equal sized tiles along the diagonal of the matrix (Supplementary Fig. 2). We only vary these variables because the runtime complexity of a single iteration of the MCMC (excluding sampling time) is $O(nmk)$. In each iteration, the likelihood of each sample vector is computed under each of the k existing clusters and the likelihood of each feature vector is computed under the signal and background models. Note that the value of k can change between iterations. To avoid fluctuating values of k significantly affecting runtime, the simulated data was constructed with a large difference between the signal (BetaBinomial(n=100, a=90,b=10)) and background (BetaBinomial(n=100, a=10,b=90)) distributions and runs of the algorithm were initialized with the correct number of clusters (using k-means). We ran the algorithm for 10 iterations on each dataset on a Intel(R) Xeon(R) Gold 6238R CPU @ 2.20GHz CPU machine with 512 GB of RAM and recorded the mean runtime per iteration.

To assess memory usage, we used the mprof utility from the python package memory_profiler. This tool samples the memory usage of the algorithm in 1 second intervals. We report the peak usage as the max memory (Mb) used across the sampled data points. Together, these results indicate that the runtime and memory increase with the number of features and samples. Increasing sample count increases runtime at a higher rate than increasing the number of features. Cluster count has a the largest effect on runtime and memory. For very large datasets, the runtime and memory usage appear to increase at different rates from the established trends (e.g. for the 2 cluster test, there was an increase in memory usage at 800 samples and 5000 LSVs but a decrease at 1000 samples and 5000 LSVs). This occurred because the CRP was opening singleton clusters which we hypothesize was due to increasing data complexity. However, these singleton clusters are still ultimately removed in the posterior summary process.

3 CHESSBOARD Pipeline and Features

Supplementary Note 3.1: Standard CHESSBOARD Pipeline for Real Data Analysis

This section details the default/standard settings and parameters used to run CHESSBOARD on our real datasets. We used MAJIQ to estimate read rates from STAR aligned BAM files. The median of the bootstrapped read rates for the most variable splice junction in each LSV was used as the representative junction in the data matrix. Note that CHESSBOARD supports non-MAJIQ processed input for any splicing quantification metric that can be interpreted as a ratio of junction inclusion or relative isoform proportions. We applied our filtering procedure to the data matrix using default parameters as described (Methods). We then ran CHESSBOARD on this data matrix with a k-means initialization (on Ψ) of k = 5 and conc = 1e-100. We used hyperpriors $\alpha_0 = \beta_0 = 0.5$ (Jeffery's prior modeling propensity for high or low junction inclusion) for background and $\alpha_1 = \beta_1 = 5$ (Beta prior modeling intermediate inclusion with

moderate variance) for the signal. For missing value priors, we estimated the values empirically on GTEX whole blood data using beta-binomial regression (Methods). The regularization parameter by default is 50. We ran the MCMC chain for 1000 iterations with a burn in of 200 and thinning step size of 2 but noted that the algorithm essentially converged at 20 iterations (model likelihood stopped changing after each iteration). We observed the same convergence using chains with alternate k-means initializations. This is because we are working with very high dimensional data. Although MCMC procedures are typically run for multiple chains where each chain is assessed for convergence using a stationary test diagnostic such as Heidelberger and Welch, we opt to use treat the optimization procedure using an expectation maximization approach with multiple start conditions given that model likelihood stop changing relatively quickly. This can be interpreted as an approximation for a variational inference approach where the MCMC is treated like a gradient based optimization.

Supplementary Note 3.2: Convergence Diagnostics

To enable alternative approaches for users to evaluate convergence of the MCMC samples to a stationary distribution, we implemented the Heidelberg-Welch test[7]. The hypotheses of the test are

- H_0: The chain is from the stationary distribution.
- H_1: The chain is not from the stationary distribution.

The input for this test is the log-posterior likelihood Markov chain obtained by running the CHESSBOARD algorithm. For a chain of length n, subchains are constructed by removing the first z samples of the chain in increments of a specified step size up to half the chain length $(n/2)$. For each subchain, we compute the statistic vector $B_z(t)$ for $\{t|0 \leq t \leq n \wedge nt \in \mathcal{Z}^+\}$ where $\mathcal{Z}^+$ is the set of positive integers and θ_j is the j^{th} element of the subchain.

$$
\begin{aligned}
T_k &= \sum_{j=1}^{k} \theta_j \\
\bar{\theta} &= \frac{\sum_{j=1}^{n-z} \theta_j}{n-z} \\
B_z(t) &= \frac{T_{\lceil (n-z)t \rceil} - \lceil (n-z)t \rceil \bar{\theta}}{\sqrt{(n-z)S(0)}}
\end{aligned}
\tag{8}
$$

Here, $S(0)$ is a scalar defined as the spectral density of the latter half of the full chain at frequency zero. There are multiple approaches to computing $S(0)$ but we use an auto-regressive model approach by first fitting the following auto-regressive model with lag p to the chain. The degree (lag) of the model was selected using AIC.

$$
\theta_j = \sum_{i=1}^{p} \beta_i \theta_{j-p} + \epsilon \tag{9}
$$

Then we compute $S(0)$ using the Yule-Walker method.

$$
S(0) = \frac{Var(\epsilon)}{(1 - \sum_{i=1}^{p} \beta_i)^2} \tag{10}
$$

To compute a p-value for this test, we compute the test statistic

$$
\int_0^1 B_z(t)^2 dt \tag{11}
$$

We approximate the integral using a Reimann sum and compute the p-value using the CDF for the Cramer-Von-Mises Distribution. Please note that the null distribution indicates that the chain is stationary. Thus the first subchain that has a non-significant p-value is used since we cannot reject the null.

To further assess the robustness of CHESSBOARD's posterior distribution, we applied the algorithm to three synthetic datasets and evaluated convergence of the MCMC (Supplementary Fig. 3). In each dataset (samples = 30, LSVs = 30), LSVs were simulated from the same Beta-Binomial distribution to ensure that there was a consistent difference in variability between datasets. The distribution and tile structure used in each dataset is shown in Supplementary Fig. 3. The left most dataset has the highest variability while the right most has the lowest. We ran CHESSBOARD on each datset for 2000 MCMC iterations with $\alpha = 0.1$ and a k-means initialization of $k = 5$. In the highest variance dataset, 5 clusters were found. Although this does not match the ground truth cluster number from the the simulation, the result is reasonable as shown by the pairwise clustering probability plot. The samples from both clusters are clearly separated as there is 0 probability of them clustering together. However, within each of the original clusters, there appear to be 2 subclusters. One is a high confidence group (higher probability of samples clustering together/darker color in the heatmap) and the other is a low confidence group (low probaiblty of samples clustering together/lighter color in the heatmap). The final cluster is an outlier group with samples that are a poor fit for all of the other clusters. The marginal probabilities of each cell in the matrix belonging to the signal distribution also shows high variability. Note that the signal-background distribution designation does not matter here since the clusters are equal in size. The tile structures still are correctly identified (just at a finer resolution). The Heildelberg-Welch diagnostic indicates that the log-posterior of the model has converged to the stationary distribution after 229 iterations (step size = 1, p = 0.695). As the variability of the datasets increase, the variance of the posterior begins to decrease. The second dataset converges at 114 iterations (step size = 1, p = 0.114). The third dataset converges almost instantly (step size = 1, p = 0.320). This is due to extremely low data variability which results in a low variance posterior with high confidence clusters. It should be noted that this dataset is representative of our realistic simulated data (Supplementary Note 2.3) and our real

data (Supplementary Note 1.1). Consequently, the algorithm converges very quickly to a stable solution on these datasets. Furthermore, these datasets are very high dimensional in comparision to the datasets used in this analysis which makes it much harder for the algorithm to explore the full posterior. For such data, it is recommended to use the convergence evaluation criteria described in Supplementary Note 3.1.

Supplementary Note 3.3: Using CHESSBOARD as a Predictive Model

To use CHESSBOARD as a predictive model, we want to assess the probability of a sample being assigned to one of the clusters. This is given by

$$P(c_i = z|x_i) \propto [r_z P(x_i|\alpha_1, \beta_1) + (1 - r_z)P(x_i|\alpha_0, \beta_0)]P(c_i = z) \tag{12}$$

CHESSBOARD learns the parameter values from a training dataset by using the parameters from the sample that minimizes MSE to the posterior mean. We can then predict the likelihood of a sample given all the learned parameters using a mixture of Beta Binomial likelihoods. We used this approach to predict clustering assignments for the samples in the Penn HTSC dataset. First, we ran the algorithm on the beatAML datasets to learn all of the above parameters. Since there were 2 clusters, $z \in [1, 2]$. Then we assigned 2 likelihoods to each sample in the Penn HTSC dataset. One for $z = 1$ and one for $z = 2$. Finally, we made a hard clustering assignment by placing the sample in the cluster for which it had the higher likelihood.

Supplementary Note 3.4: Statistical Testing in Regulation Analysis with ENCODE Data

In this section, we detail the statistical tests used to generate Fig. 3f. To determine if a RBP/SF is differentially expressed between the signal and background clusters, we use DeSeq2 on Salmon transcript expression quantifications. If a gene has multiple quantified transcripts, we consider the gene differentially expressed if at least one transcript is DE. A transcript is DE if its log2FC

> 1 and its bonferroni corrected p-value is < 0.05. To determine if a RBP/SF is differentially spliced, we use MAJIQ Het on MAJIQ quantifications. If a gene has multiple LSVs, we consider the gene differentially spliced if at least one junction is DS. A LSV is DS if its median $\Delta\Psi$ is > 0.2 and bonferroni corrected Wilcoxon p-value is < 0.05 for a junction. To determine which splice junctions were regulated by each RBP/SF, we performed a differential splicing analysis using MAJIQ between RBP/SF knockdown samples and controls. For each RBP/SF, there were 2 replicates for knockdowns and 2 for controls. The RBP/SF knockdown experiments in the ENCODE dataset were generated in 32 batches in K562 cell lines. We used MOCCASIN to correct for batch effects. The controls were considered all together as one group. Any junction with posterior probability of $\Delta\Psi > 0.2$ was considered differentially spliced and thus regulated by the RBP/SF. To determine which junctions had CLIP binding of the RBP/SF, we first identified binding regions by assessing whether the Irreproducible Discovery Rate (`https://www.encodeproject.org/data-standards/terms/#concordance`) p-value was < 0.05 for CLIP peaks generated from 2 replicates. Then we checked whether the region bounded by the RBP overlapped with a 250 bp window flanking each side of the junction. A window that overlaps with at least one binding region indicated that the junction had CLIP binding. To compute p-values for enrichment of overlap, we used a 1 sided fisher's exact test for enrichment on the following 2 x 2 table. The null hypothesis for the 1 sided test is that the odds ratio of junctions in tile to junctions not in tile is greater than 1.

	Junction in Tile	Junction not in Tile
Regulated/Binding/Both	Junction in Tile & Regulated/Binding/Both	Junction not in Tile & Regulated/Binding/Both
$\sim$(Regulated/Binding/Both)	Junction in Tile & $\sim$(Regulated/Binding/Both)	Junction not in Tile & $\sim$(Regulated/Binding/Both)

Regulated indicates the junction was in a LSV that was determined to be regulated in the ENCODE analysis. Binding indicates CLIP binding of the RBP/SF was observed in the ENCODE

data near the junction. Both is the intersection of regulated and binding.

Supplementary Note 3.5: CHESSBOARD can Rank Tile's Splicing Events for Downstream Analysis

In many genomic analysis tasks such as differential splicing or gene expression researchers are interested in a ranked list of entities (e.g. mutations, genes), which they then test for enrichment of some biological signal (e.g. pathways). Ranking is also desirable since the tile structure is clearly an approximation of the underlying biological signals. Specifically, some LSV may exhibit a strong pattern that closely matches the patients subgroups thus "driving" the tile formation while others can be considered as "passengers" with a much less clear pattern. However, standard differential splicing analysis can not be applied in this setting as it is based solely on observed Ψ and ignores the missingness signal discussed above. Thus, to address the need for splicing changes ranking we developed a LSV ranking procedure that takes advantage of CHESSBOARD's probabilistic framework. The ranking score is computed as

$$Rank(LSV_j) = \sum_{i}^{N} [log(P(x_{ij}|\alpha_{j1}, \beta_{j1}, c_i, r_{c_i})) - log(P(x_{ij}|\alpha_{j1}, \beta_{j1}, c_i, r*_{c_i}))] \tag{13}$$

The first term represents the likelihood of the LSV under the learned tile model while the second term represents the likelihood under an inverted model. Recall that the likelihood under the tile model is computed as the mixture of a signal and background Beta distribution where subscript 1 indicates signal and subscript 0 indicates background.

$$P(x_{ij}|\alpha_{j1}, \beta_{j1}, c_i, r_{c_i}) = r_{c_i} P(x_{ij}|\alpha_{j1}, \beta_{j1}) + (1 - r_{c_i}) P(x_{ij}|\alpha_{j0}, \beta_{j0}) \tag{14}$$

Under the inverted model, we compute the likelihood in the same way except $r*_{c_i}$ is defined as $r*_{c_i} = 1 - r_{ci}$. Intuitively, the ranking score is the total likelihood a LSV "gained" from the learned model compared to the alternative. If the the signal and background distributions are similar (i.e. KLD is low), then the data has similar likelihood under both models. This

indicates that the LSV does not strongly drive tile structure because a sample would have approximately equal probability of being assigned to signal or background if classified using this feature alone. Conversely, LSVs with a high score gain substantial likelihood from the learned tile configuration and strongly contribute to the tile structure. When applied to the tiles derived from the AML associated genes, we find a distinct exponential shape of the score distribution. This result indicates that a few "driver genes" define the tile shape while the majority of LSV features contribute much less to the structure (Supplementary Fig. 5a). We then confirmed that the score is directly correlated with the amount of separation between modalities. We observed that the highest ranking LSV has large separation while LSVs with scores in 75th, 50th and 25th percentile show decreasing separation (Supplementary Fig. 5b). The modalities in the lowest scoring LSV were almost completely overlapping. Finally, we assessed the rankings of the notable LSVs we analyzed in the previous section. *FLT3* LSV1 and LSV2 rank near the 50th percentile while the two *EZH2* LSVs ranked lower. However, we note that *EZH2* was prioritized due to its missingness pointing to the importance of our missing value model. The top ranking LSV was *U2AF1* which is expected to regulate a substantial number of the events in the tile given the regulatory analysis described above.

4 beatAML Analysis

Supplementary Note 4.1: Recursive Clustering and Termination

The CHESSBOARD framework naturally enables recursive clustering through its tile based clustering approach and probabilistic framework. The recursive clustering algorithm is presented below.

Algorithm 1: Recursive Clustering

$M_0 = CHESSBOARD(X_{F_0})$;
$\bar{X}_{F_0} = Shuffle(X_{F_0})$;
$M_0* = CHESSBOARD(\bar{X}_{F_0})$;
$LR_0 = median(LL(X_{F_0}, M_0) - LL(\bar{X}_{F_0}, M_0*))$;
while $|LR_n - LR_{n-1}| < T$ **do**
  $F_n = \{LSV_j | \forall j \ s.t. \sum_c r_{jc} = 0\}$;
  $M_n = CHESSBOARD(X_{F_n})$;
  $\bar{X}_{F_n} = Shuffle(X_{F_n})$;
  $M_n* = CHESSBOARD(\bar{X}_{F_n})$;
  $LR_n = median(LL(X_{F_n}, M_n) - LL(\bar{X}_{F_n}, M_n*))$;
end

Let X represent a data matrix with rows representing all LSVs in the transcriptome and columns representing all patient samples in the dataset. Define F_0 as the feature set of the initial input matrix (i.e. all features that pass the pre-filtering pipeline). We apply CHESSBOARD to the matrix X_{F_0} to obtain posterior model M_0. M_0 represents all latent posterior random variables learned from the data. We then generate a null matrix with the same feature set denoted as $\bar{X}_{F_0}$ by shuffling the rows of X_{F_0}. The shuffling procedure involves independently and randomly permuting each row of the input matrix to break tile structure. Next, we obtain the posterior model M_0* by running the algorithm on $\bar{X}_{F_0}$. We then use both posterior models to evaluate the log likelihood ratio LR_0 of each LSV in X_{F_0} under model M_0 to $\bar{X}_{F_0}$ under model M_0*. To conduct the first recursive step, define feature set containing LSVs not assigned to a tile F_n as $\{LSV_j | \forall j s.t. \sum_c r_{jc} = 0\}$. We then apply CHESSBOARD to X_{F_n} and null matrix $\bar{X}_{F_n}$ to obtain model M_n and M_n* respectively as in the base case. This procedure is repeated until the termination criteria is met. We terminate the algorithm once the median LR stops changing from the previous iteration. We use the median LR of the LSV LR distributions so the likelihoods are comparable between iterations. Using the likelihood of the entire matrix for example would not be comparable since the cardinality of the feature sets decrease after each iteration. Non-changing can be defined as either the difference between medians being

below some threshold T or a test can be used to reject the null hypothesis that the likelihood means are unequal.

We also present an augmented version of the recursive algorithm. In our applications, we did not notice any major differences between the two versions of the algorithm and opted to not use the augmented algorithm due to run-time considerations. Under this approach, feature sets F are not confined to only LSVs in the input data matrix but include all LSVs in the transcriptome which are correlated to the unique structures that contain the LSVs in F. Specifically, for each unique binary vector r_j, divide the samples i into groups such that $r_{jc_i} = 0$ and $r_{jc_i} = 1$. Remove any LSVs from X that are differentially spliced between the groups. Once all LSVs correlated to each unique r_j have been removed, proceed to the next recursive step. Next, when generating the null model, instead of generating a single model, we can generate a bootstrapped empirical distribution over models through multiple permutations of X. Formally, we have $\bar{X}^1, \bar{X}^2 \ldots \bar{X}^B$ where B is the number of bootstrapped samples. Then we can evaluate whether the likelihood X_F is significant in the context of the distribution of $\bar{X}^{1,2,\ldots,B}$ and terminate the algorithm if the likelihood of X_F is an outlier.

Supplementary Note 4.2: Multiple Testing Correction with Missing Data

To correct for the family wise error rate in multiple testing, we use a min-P procedure. Let $C = \{c_i | \forall i \in [N]\}$ be the set of cluster labels for all N samples. For each mutation m out of M total mutations, we compute an observed p-value p_m using a two-sided fisher's exact test on the following 2 x 2 matrix.

	$c_i = 0$	$c_i = 1$
Mutation+	Mutation+ & $c_i = 0$	Mutation+ & $c_i = 1$
Mutation-	Mutation- & $c_i = 0$	Mutation- & $c_i = 1$

Missing mutation annotations are ignored. We then generate a bootstrapped null p-value distribution. For each of B bootstrapped samples, we randomly permute C such that each c_i

takes on the value of a random $c_i \in C$ without replacement. Then given these new clustering assignments, we compute fisher's exact test again for each mutation to obtain $p*^b_m$. For each bootstrapped sample, we record the minimum p-value across all mutations $minP_b = min(p*^b_1, p*^b_2, \ldots, p*^b_M)$. To compute the corrected p-value for a mutation m, we count the number of times the observed p-value $\sum_b I(p_m > minP_b)$ and divide by B.

5 Drug Response Analysis

Supplementary Note 5.1: Drug Response Correlation

Drug response in the BeatAML dataset was quantified using two measurements: IC_{50} and AUC. IC_{50} is the concentration (μM in the beatAML study) at which a drug inhibits a target biological process by 50%. In the beatAML study, inhibition was measured as the normalized cell viability which is a quantity derived from the optical density of surviving tumor cells in a plate after treatment with the drug. The IC_{50} value for each drug was obtained (by the beatAML study) by fitting a sigmoid curve to 7 data points defined as the tuple (concentration, inhibition) and selecting the concentration at which inhibition is 50% of its maximum value. Each of the 7 concentrations represents a 3 fold dilution starting at $10\mu M$ and ending at $0.0137\mu M$. When we compare IC_{50} changes in this study, we use a log3 transform since each unit change of IC_{50} in log3 space is 1 fold change unit given the 3 fold dilution. AUC is the area under the IC_{50} curve which is an unbounded positive quantity and a larger value indicates poor drug response. We chose to use AUC over raw IC_{50} because concentration ranges used in the beatAML experiments limited fitting of sensitivity curves. However, we note that the median AUC and IC_{50} changes between the groups in beatAML were highly correlated (r = 0.727) indicating that quantitative comparisons in AUC space translated to IC_{50} space (Supplementary Fig. 6a). Interestingly, most outliers in this figure (i.e cases when there is low agreement between IC_50 and AUC changes) had median IC_{50} values of 10 in both clusters ($\Delta IC_{50} = 0$) and were likely

the most affected by the concentration range. This can occur when there is no sigmoid curve but rather a horizontal line due to the fact no concentration in the chosen range produced any noticeable effect. In this case, relative AUC quantities can still be compared but IC_{50} cannot. A change in AUC quantity should thus be used to assess whether there is difference in response between groups. However, the difference itself does not have a clear interpretation. Instead, given a significant difference in AUC, one can then look for a large fold change in IC_{50} to determine if the change is biological meaningful. The patient groups differ most significantly by their sensitivity to JQ1 (p = 7.21e-05, Δmedian(AUC) = 25.76) (Supplementary Data 4) based on a Kruskal-Wallis test. The tile group had a higher median IC_{50} compared to the background group corresponding to a Log3 fold change of 0.773 indicating a higher drug sensitivity in the background (Supplementary Data 4). The drugs Tramatenib (p = 5.77e-03, median(ΔAUC) = 42.50, Log3FC = 2.65) and Venetoclax (p = 1.56e-02, Δmedian(AUC) = 38.08, Log3FC = 2.34) were the two highest ranking drugs in terms of AUC and IC_{50} change and are common drugs administered to AML patients. In contrast, drugs with poor correlation are associated with other conditions. For example, Vemurafenib (p = 0.831, Δmedian(AUC) = 1.39) is a Melanoma drug and Lovastatin (p = 0.962, Δmedian(AUC) = 8.80) is intended to reduce cholesterol levels. To assess potential functional significance, we first looked for differential splicing of specific gene targets of these drugs. Specifically, we observed differential splicing at multiple junctions in *BRAF* (JQ1 target), *MAP2K1* (Tramateib target) and *BCL2* (Venetoclax target).

Supplementary Note 5.2: Variance Explained

$$
\begin{aligned}
Var(IC_{50}) &= E[Var(IC_{50}|\textbf{\textit{FLT3}} - ITD^{+})] + Var[E(IC_{50}|\textbf{\textit{FLT3}} - ITD^{+})] \\
VarianceExplained(\textbf{\textit{FLT3}} - ITD^{+}) &= \frac{Var[E(IC_{50}|\textbf{\textit{FLT3}} - ITD^{+})]}{Var(IC_{50})}
\end{aligned} \tag{15}
$$

Supplementary Note 5.3: Decision Tree Permutation Test

To compute whether AUC gained from including splicing profiles in the decision tree was significant, we used a permutation test. To construct the decision tree for a specific drug and mutation pair, we first split samples based on $M_i = +$ or $M_i = -$ and then conduct a second split based on splicing profiles (i.e. cluster identity). We use variance explained by the decision tree as the test statistic. The null hypothesis for this test is that combining the mutation and splicing classification does not improve drug response prediction compared to using mutation status alone. The alternate hypothesis is that the combined classification improves drug response prediction compared to using mutation status alone. For the observed test statistic, we use the variance explained of our decision tree in Fig. 5c. We then compute 1000 bootstrapped trees where we randomly permute clustering assignments into equal sized groups for the second split. To compute a p-value, we use $I(VarExp_{boot} > VarExp_{obs})/1000$.

6 Model Details

Supplementary Note 6.1: Variable Table

Variable Definitions		
Variable	Description	Distribution
n	The total number of samples or number of matrix columns. Samples indices are denote as $i \in \{1, 2, \ldots, n\}$	NA
m	The total number of LSVs or number of matrix rows. LSV indices are denote as $j \in \{1, 2, \ldots, m\}$	NA
x_{ij}	The number of reads mapped to the representative splice junction of LSV j in sample i.	Binomial
η_{ij}	The total number of reads mapped to LSV j in sample i.	NA
ω_{ij}	An indicator variable which denotes whether LSV j in sample i is missing/unquantifiable. $\omega_{ij} = \begin{cases} 1 & \text{if observation is missing} \\ 0 & \text{otherwise} \end{cases}$	Bernoulli
c_i	Denotes assignment of sample i to cluster k if $c_i = k$.	CRP
r_{jk}	Denote assignment of LSV j for all samples such that $c_i = k$ to either the background ($r_{jk} = 0$) or signal ($r_{jk} = 1$) distributions.	Bernoulli
Ψ_{ij}	Percent splice in of LSV j in sample i	Beta
θ_{js}	Missingness rate of LSV j for the signal distribution $s = 1$ or background distribution $s = 0$	Beta
μ_{js}	Mean of Ψ of LSV j for the signal distribution $s = 1$ or background distribution $s = 0$	Beta
κ_{js}	Concentration/inverse variance of Ψ of LSV j for the signal distribution $s = 1$ or background distribution $s = 0$	Prior
α_{js}, β_{js}	The priors for the Beta distribution modeling μ_{js}	Hyperprior
a_{js}, b_{js}	The priors for the Beta distribution modeling missingness rate θ_{js}	Prior
$\sum_k r_{jk}$	Regularization term.	Exponential
λ	Regularization hyperparmeter.	Hyperparameter
α_o	CRP concentration parameter	Hyperparameter

7 Survival Analysis

Supplementary Note 7.1: Survival Analaysis

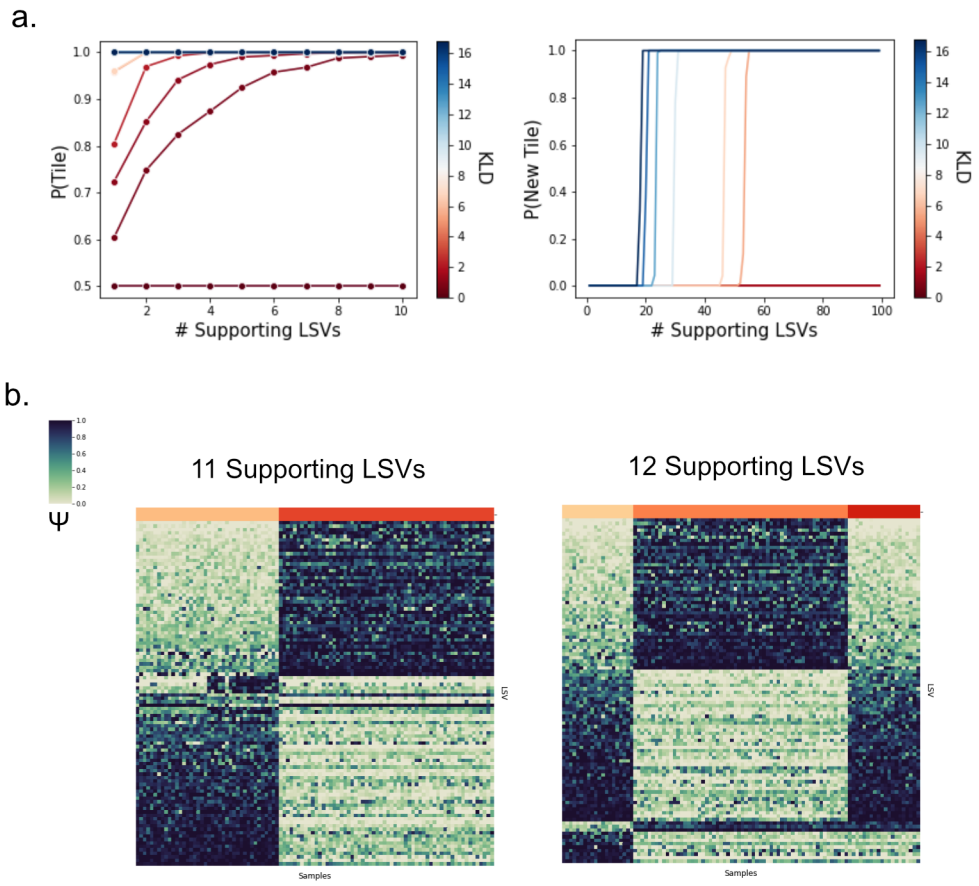
We performed a survival analysis on the beatAML samples. Patients were divided into two groups based on whether they were assigned the signal or background cluster (Fig 3a). Out of the 217 samples in the signal cluster, 132 had survival data (60.83%). Out of 260 samples in the background cluster, 184 had had survival data (70.77%). There were no censored observation in which patients dropped out of the study for reasons other than the event of interest (death). We generated the survival curves for the background and signal groups denoted as $S_0(t)$ and $S_1(t)$ respectively for all available time points t using the Kaplan-Meier method (Supplementary Fig. 7). To assess whether the survival distributions were significantly different, we used the log rank test which is the standard in the field. The hypotheses of the test are

- $H_0 : S_0 = S_1$
- $H_0 : S_0 \neq S_1$

The log-rank p-value was 0.6265. Although this is not significant, it is likely due to the fact that the log rank test is under-powered to detect differences in non-diverging survival distributions[8]. The log rank test is a multi group extension of a chi-squared test which assesses whether the observed cumulative deaths (after accounting for censored data) in each group different significantly from the expected cumulative deaths in the dataset (i.e. groups are combined). Thus the test is under-powered when the survival curves converge resulting in similar cumulative death totals. In such scenarios, any divergence or omnibus test would be more appropriate. Since we do not have censored data, we can also assess significance using a 2 sample Kolmogorov-Smirnov test. The KS test statistic is $D = 0.158$ and the KS p-value is 0.054. The test statistic can be interpreted as the largest difference in survival probability between the

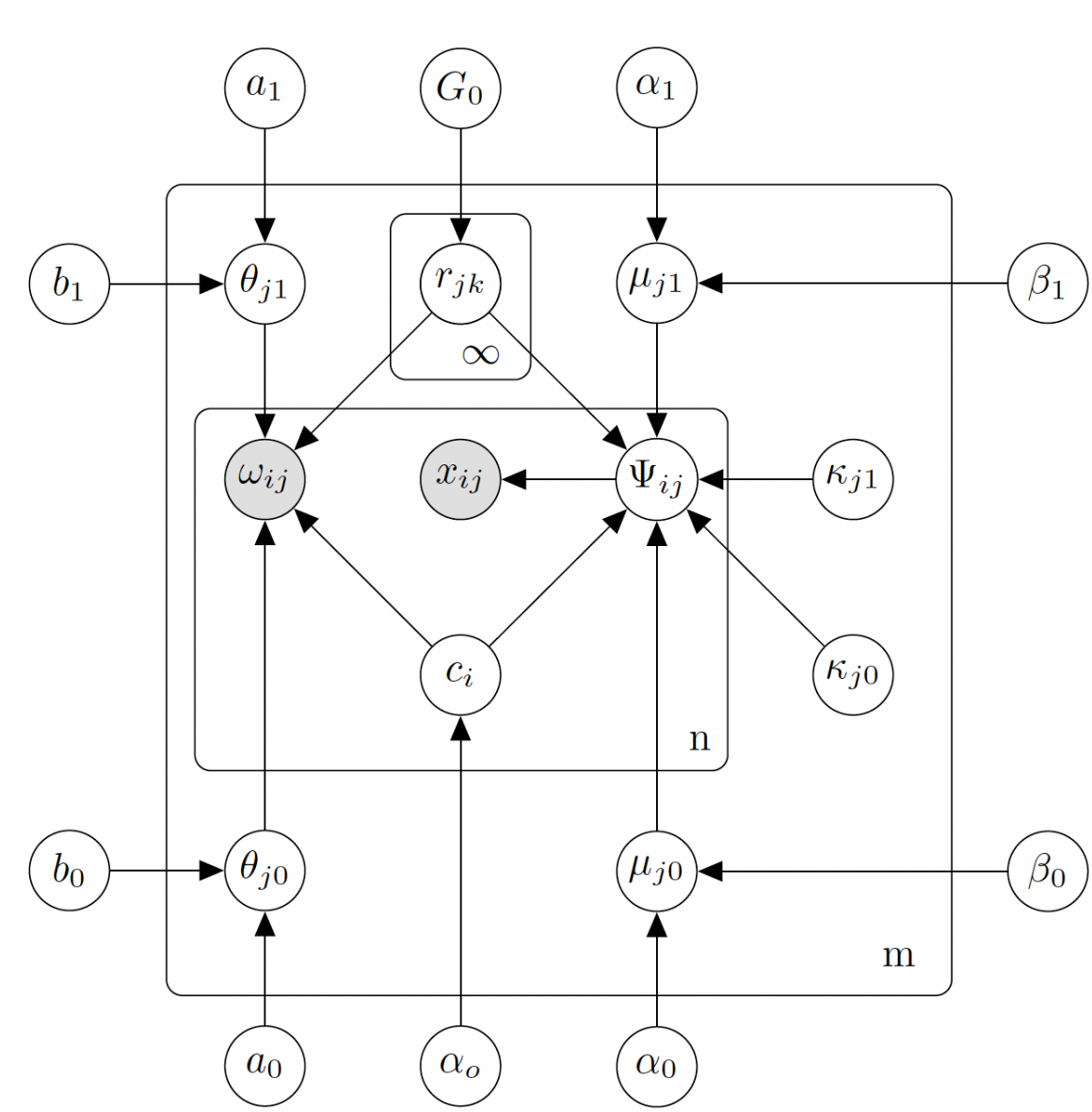
groups which is 15.8%. Specifically, The survival of the first seven months after diagnosis is similar and the ultimate survival after several years is also the same. Only in between these two time points some difference could be observed. While this represents a statistically significant difference in survival rate and we report it here for completeness, the biological significance of this difference is unclear and also difficult to explain.

Supplementary Figures



Supplementary Fig. 1: Additional Model Evaluation.

(A) Probability of cluster discovery based on evidence in data. LEFT: Probability of the CHESSBOARD model assigning a sample to the tile/signal distribution of equal size (i.e. k = 2 initialization) as a function of increasing KL divergence (red low, blue high) w.r.t to a background Beta distribution of Beta(10,1). and number of supporting LSVs in the tile. RIGHT: Probability of assignment when discovering new tiles with the concentration parameter set using an effective dimensionality (Supplementary Note 2.2) of 50. **(B)** Number of supporting LSVs required to identify tile in realistic synthetic data. Left heatmap shows the clustering result on data with 3 tiles where one of the tiles only has 11 supporting LSVs. The algorithm is however only able to find the 2 main tiles. Right heatmaps shows the clustering result when the 3rd tile has 12 supporting LSVs. The algorithm is able to successfully find all 3 tiles.



Supplementary Fig. 8: Plate Diagram.

A plate model showing the relationship between latent variables in a model. An arrow indicates the child node is dependent on the parent. Observed variables are shown in grey. Latent variables are white. See variable table in Supplementary Note 6.1.

Supplementary Data

The Supplementary Data include the human readable output of CHESSBOARD on each of our datasets. These are formatted as excel files where each tile k is defined by the contents under 3 tabs: Sample k, Cluste k and Background k. Sample denotes the sample IDs in the tile. Cluster defines the LSVs that belong to the signal group in the tile. Background defines the LSVs the belong to the background group in the tile. There are 2 additional tables in each file: Consensus Background and Probability Missing Signal. Consensus Background defines all LSVs that don't belong to a signal in any tile. Probability Missing Signal defines the p-values that the signal group for a given LSV is enriched for missing values based on fisher's exact test. All files can be found in the Zenodo repository at https://zenodo.org/record/7245323#.Y1apPFLMKQc.

Supplementary Data 1: beatAML

Format defined above for the beatAML dataset.

Supplementary Data 2: beatAML Recursive Step 1

Format defined above for the first recursive step applied to the beatAML dataset.

Supplementary Data 3: beatAML AML Genes

Format defined above for the beatAML dataset using only AML related genes.

Supplementary Data 4: Drug p-values

Kruskal-Wallis p-values for differential drug response (measured as AUC) between the clusters discovered in the beatAML dataset using AML genes.

Supplementary Data 5: TARGET AML

Format defined above for the joined dataset of beatAML and TARGET pediatric AML datatsets.

Supplementary Data 6: TARGET B-ALL

Format defined above for the TARGET B-ALL dataset.

Supplementary References

[1] Tyner, J. W. *et al.* Functional genomic landscape of acute myeloid leukaemia. *Nature* **562**, 526–531 (2018).

[2] Rivera, O. D. *et al.* Alternative splicing redefines landscape of commonly mutated genes in acute myeloid leukemia. *Proceedings of the National Academy of Sciences* **118** (2021).

[3] Van Nostrand, E. L. *et al.* A large-scale binding and functional map of human RNA-binding proteins. *Nature* **583**, 711–719 (2020).

[4] Slaff, B. *et al.* MOCCASIN: A method for correcting for known and unknown confounders in RNA splicing analysis. *Nature Communications* **12**, 1–9 (2021).

[5] Shen, S. *et al.* rMATS: robust and flexible detection of differential alternative splicing from replicate RNA-Seq data. *Proceedings of the National Academy of Sciences* **111**, E5593–E5601 (2014).

[6] Vaquero-Garcia, J. *et al.* A new view of transcriptome complexity and regulation through the lens of local splicing variations. *elife* **5**, e11752 (2016).

[7] Heidelberger, P. & Welch, P. D. Simulation run length control in the presence of an initial transient. *Operations Research* **31**, 1109–1144 (1983).

[8] Li, H., Han, D., Hou, Y., Chen, H. & Chen, Z. Statistical inference methods for two crossing survival curves: a comparison of methods. *PLoS One* **10**, e0116774 (2015).