1 Implementation of Clustering

Prior to performing the clustering, we took the log of the gene and isoform counts and further standardized the data across each feature, specifically by centering the data by the median and scaling the data by the median absolute deviation (mad). We then found the pairwise distance within each feature by utilizing the Euclidean distance as the dissimilarity measure between observations for gene and isoform.

We similarly found the distance for each feature in the case where we treated the gene as a vector of proportion. Less obvious in this case in the choice of which distance measure to use. In order to explore this, we evaluated five distances on proportions. We only show the Hellinger distance, Jeffrey's divergence, and Log-likelihood distances in our plots, since Euclidean and χ^2 resembled Hellinger and Jeffrey's divergence, respectively, in our simulations (Supplementary Figure S1) and on implementation on real data. Dropping the dependence on the gene j, the distance between two proportions, p_i and $p_{i'}$, is defined as:

- Squared χ^2-measure:

$$d(p_i, p_{i'}) = \sum_{k=1}^{K} \frac{(p_{ik} - p_{i'k})^2}{p_{ik} + p_{i'k}} \tag{1}$$

- Euclidean distance:

$$d(p_i, p_{i'}) = \sqrt{\sum_{k=1}^{K} (p_{ik} - p_{i'k})^2} \tag{2}$$

- Jeffrey's divergence:

$$d(p_i, p_{i'}) = \sum_{k=1}^{K} (p_{ik} - p_{i'k}) \ln \frac{p_{ik}}{p_{i'k}} \tag{3}$$

- Hellinger distance:

$$d(p_i, p_{i'}) = \sqrt{2 \sum_{k=1}^{K} (\sqrt{p_{ik}} - \sqrt{p_{i'k}})^2} \tag{4}$$

- Log-likelihood based distance (Berninger *and others*, 2008; Witten, 2011): Under the log-likelihood based dissimilarity measure, we measure whether sets of isoforms from different individuals seem to have been drawn from the same multinomial distributions. This distance measure differs from the other ones we explored as this is calculated on isoform counts directly rather than isoform relative frequency.

$$d(i, i') = \frac{\mathcal{L}(x_i, x_{i'} | p_i \neq p_{i'})}{\mathcal{L}(x_i, x_{i'} | p_i = p_{i'})} = \sum_{k=1}^{K} x_{ik} \ln \frac{x_{ik}}{x_{i+}} + x_{i'k} \ln \frac{x_{i'k}}{x_{i'+}} - (x_{ik} + x_{i'k}) \ln \frac{x_{ik} + x_{i'k}}{x_{i+} + x_{i'+}} \tag{5}$$

After calculating these per-feature distances, we then take a weighted sum of these per-feature distance matrices to calculate one distance object. A distance object itself may be used as input into many different clustering algorithms. Here, we utilized hierarchical clustering from the package **stats** and k-mediods clustering functions from the package **cluster** in R, which both take distance objects as input. For clustering methods that take raw data rather than distance matrices, it is possible perform multidimensional scaling (MDS) on the distance object to return a two-dimensional data matrix which preserves the distances between individuals. Many clustering algorithms which do not take distance objects as inputs will instead take this type of data frame, such as the k-means clustering function from the package **stats**.

2 Details of Simulations

2.1 Clustering Scenarios

We examined several different ways in which clusters with different proportional usage of isoforms could occur. For any particular gene, we considered that it could demonstrate clustering in three ways, as described in the main text,

1. the gene expression counts different between groups, while the proportion levels are constant across groups (Figure 1a);
2. the gene expression is constant across samples, while the proportion levels vary between groups (Figure 1b);
3. both the gene expression and proportion levels vary across groups (Figures 1c and 1d).

We also simulated a setting where some genes followed Case 1 (i.e. had only gene expression differences) while other genes followed Case 3 (both gene and proportion differences in the same gene).

In simulations where both gene and proportion clustering coexisted (Case 3), we needed to be able to distinguish between the effect of the gene clusters and the proportion clusters. We set the proportion clusters to be different (and smaller) than that of the gene clusters. The proportion clusters were either nested within the gene clusters or spanned across gene clusters . When the proportion clusters were nested within the gene clusters, there were 3 gene clusters each with three proportion clusters nested inside making 9 proportion clusters (Figure 1c). In the case where the proportion clusters are not all nested within gene clusters, there were the same 3 clusters with different gene expressions, and 6 clusters with different proportion usage values; because the clustering directly on isoform expression levels should theoretically be able to detect both gene and proportion differences since they result in different isoform values, this results in 9 clusters with different isoform expression values (Figure 1d). In our plots of this case, we show isoform clustering's ability to detect the three gene clusters as well as the nine isoform clusters (isoform naturally does not detect the six proportion clusters).

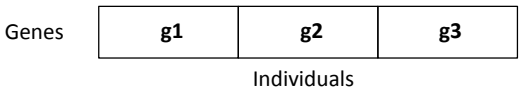
To simulate from these cases, a percentage of the genes were choose to all have a set pattern in their gene expression and isoform proportions corresponding to one of the cases above, while the remaining genes had both constant gene and isoform expression (no clustering signal). The percentage of genes with the pattern varied from 0.5%, 1%, 2%, 4%, 8%, and 10%.

We in the setting where some genes followed Case 1 while other genes followed Case 3, we held the percentage of genes following Case 1 fixed at 25% and allowed the percentage of genes following Case 3 to vary according to the same proportions given above (and with the remaining genes held constant in gene and isoform expression).

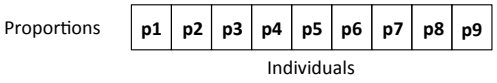
2.2 Generating Isoform Counts

In each simulation, we simulated 5,000 genes across 135 samples. In order to determine the number of isoforms per gene, we determined that the mode for the number of expressed isoforms of genes with multiple isoforms was two in the TCGA data (described below in Section 3. For our simulation, we chose the number of isoforms we would simulate for each gene from a Poisson distribution with $\lambda = 2$. Since we were interested in only multiple isoform genes, we filtered this generated set of random numbers to include only values greater than one. Such a distribution had a mode of 2 isoforms, with a mean typically between 2 and 3 isoforms, which is similar to the distribution we saw in multiple isoform genes in real data.

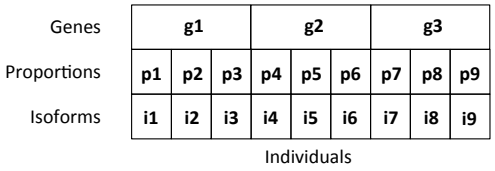
To define proportional usage differences, we generated for each sample a different set of isoform values for each gene; these isoform values then determined the proportional usage values for the gene for



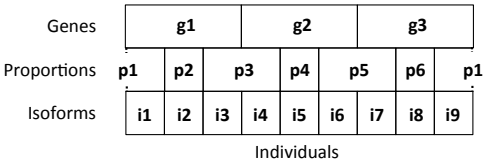
(a) Case 1: Different Gene Expression



(b) Case 2: Different Relative Isoform Levels



(c) Case 3: Genes and isoforms varying (within gene clusters)



(d) Case 4: Genes and isoforms varying (across gene clusters)

Figure 1: Illustration of the different possible clusters simulated. (a) and (b) show the groups used when we allow for only gene clusters (g1-g3) or proportion clusters (p1-p9), respectively. (c) and (d) illustrate the two sets of clusters used when we combine the proportion clustering groups and the gene clustering groups in the same simulation; in both setting the clusters showing differences in gene expression are the same as in (a) (g1-g3), but the clusters showing differences in proportions differ. (c) illustrates the case where the nine proportion clusters (p1-p9) define subgroups of the three larger groups defined by gene expression differences (i.e. proportion groups are nested within gene groups). (d) illustrates the six clusters (p1-p6) used when the proportion groups can span the gene groups. Note that for this setting that while there are six groups showing differences in proportional isoform usage, the combination with differing gene expression levels mean there there are *nine* groups showing differences in isoform expression (i1-i9). Each of the small (nine) rectangles consists of 15 samples resulting in 135 samples.

that sample by dividing these isoform counts by their total within the gene. These isoform values were simulated from a negative binomial with the same parameters within each proportion group. To determine the parameters of negative binomial, we used edgeR Robinson *and others* (2010) to estimate the parameters of a negative binomial distribution on the TCGA data, estimating a mean and dispersion parameter for each isoform. Then for each proportion group, we sampled a set of mean and dispersion parameters from those calculated from the real TCGA data. Once the parameters for the proportion cluster were decided, we then sampled counts for each sample within the cluster from a negative binomial distribution with those parameters and calculated the proportion vectors from the isoform counts.

Because we wanted to be able to control the gene cluster groups and proportion cluster groups independently, we did not use these isoform totals to create the gene counts. Instead, the total gene count was simulated separately in the same manner. For every gene in a gene cluster, we again drew parameter values for each gene cluster from the distribution of values defined by the TCGA data and then simulated isoform values for each sample from a negative binomial with those parameters. These isoform counts were then summed to get the gene estimates, per sample.

The final isoform counts were derived by multiplying the gene counts by the final proportion vector.

2.3 Evaluating the performance of the clustering

The simulated isoform counts, gene counts, and proportions were then clustered as described in Section 1 for appropriate values of K. Each simulation run consisted of 1,000 simulations.

In order to quantify how well the clustering performed, we calculated the Jaccard similarity between the clusters we observed and the clusters we expected. Then for two cluster assignments A and B,

where one is the expected clustering and one is the observed clustering, the equation for the Jaccard similarity is given as:

$$J(A, B) = \frac{|A \cap B|}{|A \cup B|} = \frac{TP}{TP + FP + FN} \tag{6}$$

where TP, FP, and FN are defined as followed. The number of true positives, TP, is defined as the number of pairs that cluster together in the observed clusters as well as the true clustering. The number of true negatives, TN, is defined as the number of pairs that do not cluster together in the expected clustering nor the observed. The number of false positives, FP, is defined as the number of pairs that cluster together in the observed clustering but not the expected clustering.

For each clustering, we calculated the Jaccard similarity that corresponded to the true signal(s) we expected the method to be able to detect. For example, in the case shown in Figure 1d with the proportion clusterings spanning across the gene clusters, we calculated how well the gene clustering with $K = 3$ performed at catching the three gene groups; how well the isoform clustering with $K = 3$ performed at catching the three gene groups as well as how isoform clustering with $K = 9$ performed at catching the nine isoform groups; and how well the proportion clustering with $K = 6$ performed at catching the six proportion groups. In the case illustrated in Figure 1c the proportion clusters were nested within the gene and we were interested in how well the gene clustering with $K = 3$ caught the three gene groups; how well the isoform clustering with $K = 3$ performed at catching the three gene groups as well as how isoform clustering with $K = 9$ performed at catching the nine isoform groups; and how well the proportion clustering with $K = 9$ performed at catching the nine proportion groups.

3 Details of Analysis of LAML Data

We consider tumors collected as part of the TCGA studies on acute myeloid leukemia (AML) and we performed gene, isoform, and proportion clustering on counts downloaded from the TCGA data portal (download via https://tcga-data.nci.nih.gov/tcga/dataAccessMatrix.htm on 10/14/14). We downloaded the RNASeqV2 level 3 raw isoform counts from the portal. The TCGA data pipeline used to generate these counts included alignment to the reference genome using Mapsplice (Wang *and others*, 2010) and quantitation of transcripts using RSEM (Li and Dewey, 2011). RSEM returns two kinds of estimates, which include an estimate of the number of fragments that are derived from a given isoform or gene as well as an estimate of relative expression called transcripts per million (TPM). For our purposes, we used the estimates of the number fragments (counts) derived per isoform. We used the annotation file provided by TCGA to associate isoforms with genes, which can be found at https://tcga-data.nci.nih.gov/docs/GAF/GAF.hg19.June2011.bundle/outputs/TCGA.hg19.June2011.gaf.

The TCGA LAML dataset contained expression data for 73,599 isoforms in 173 individuals. We initially began by filtering out lowly expressed isoforms, specifically those isoforms that had a mean value of less than 25 counts across all samples. After filtering out these low expressed isoforms, 28,014 isoforms representing 12,218 genes. We then performed variance filtering in order to reduce the size of the datasets. The gene count and isoform count datasets were filtered by selecting the 5,000 most variable genes or isoforms. We then calculated the distance matrix for each of these features. In the case of the isoform proportions, we first calculated the distance matrix for all features (i.e. genes). For the AML data, we focused on Jeffrey's divergence for the proportions. We then chose the 5,000 genes with the largest summed distance matrix. We performed consensus clustering (Monti *and others*, 2003) using both hierarchical clustering and k-mediods clustering on these filtered datasets. Briefly, this involved performing repeated subsampling of the entire dataset. In each iteration, a new subset of the data, both subsetted by individuals as well as by features, is sampled. This subsample is clustered using both hierarchical and k-mediods clustering. In each iteration, which samples are clustered together are enumerated. After performing the desired number of iterations, in our case 1,000, a consensus matrix is calculated in which each entry is the fraction of times two samples were clustered together when they were sampled together. After performing all subsamples, this

consensus matrix is clustered using hierarchical clustering in order to achieve our final clustering. Ideally, consensus clustering should give us an idea of the stability of our clusters. If the clusters are well defined, we should see many entries in the consensus cluster close to 0 or 1, whereas unstable clusters may contain individuals that cluster together as often as they do not.

In the process of our analysis, we identified 11 samples which did not cluster well with other samples at most k. Typically, these samples were found to have either very low overall expression or an overrepresentation of isoforms showing no expression. We removed these samples from further analysis as they were tending to drive clustering to several one-sample clusters rather than larger clusters.

Calculation of 5' to 3' bias: In order to determine whether a 5' or 3' bias is present, we implemented the Python module geneBody_coverage.py found as part of RSeQC (http://rseqc.sourceforge.net/) (Wang *and others*, 2012). This module calculates the read coverage across genes to determine if read coverage is uniform or contains bias in coverage across the genes. RSeQC divides the gene into equally spaced bins and calculates the number of sequences falling in the bin, relative to the overall number sequences assigned to the gene region. For more details see Wang *and others* (2012).

This module requires raw BAM files as input, which we downloaded from the Cancer Genomics Hub (https://cghub.ucsc.edu/index.html); the raw BAM files (unlike the count summaries) are held under controlled access and only available to those who have applied for and received a Data Access Request (DAR). Additionally, a BED file describing a set of housekeeping genes is also required as input to RSeQC and is available on the RSeQC site:
(http://sourceforge.net/projects/rseqc/files/BED/Human_Homo_sapiens/hg19.HouseKeepingGenes.bed).

However, our reference annotation was based on hg18, and we instead downloaded the hg18 bed file from the UCSC genome browser (https://genome.ucsc.edu/cgi-bin/hgTables) and then limited the set of UCSC genes to those also found in the housekeeping gene file on the RSeQC website. We further limited the housekeeping genes to those which showed expression in only one isoform in our dataset (that is, were single isoform genes) so as to avoid any issue of differential coverage due to alternative splicing. We separated the bed file by the size of the gene and looked separately at genes with less than 1,000 base pairs (318 genes), 1,000-1,500 base pairs (460 genes), 1,500-2,000 base pairs (532 genes), 2,000-2,500 base pairs (493 genes), and 2,500-3,000 base pairs (474 genes).

References

Berninger, Philipp, Gaidatzis, Dimos, van Nimwegen, Erik and Zavolan, Mihaela. (2008). Computational analysis of small {RNA} cloning data. *Methods* **44**(1), 13 – 21. MicroRNAs: Part B.

Li, B and Dewey, C N. (2011). RSEM: accurate transcript quantification from RNA-Seq data with or without a reference genome. *BMC Bioinformatics* **12**(1), 323.

Monti, S, Tamayo, P, Mesirov, J and Golub, T. (2003). Consensus clustering: a resampling-based method for class discovery and visualization of gene expression microarray data. *Machine learning* **52**(1), 91–118.

Robinson, Mark D, Mccarthy, Davis J and Smyth, Gordon K. (2010, January). edgeR: a Bioconductor package for differential expression analysis of digital gene expression data. *Bioinformatics (Oxford, England)* **26**(1), 139–140.

Wang, Kai, Singh, Darshan, Zeng, Zheng, Coleman, Stephen J, Huang, Yan, Savich, Gleb L, He, Xiaping, Mieczkowski, Piotr, Grimm, Sara A, Perou, Charles M, MacLeod, James N, Chiang, Derek Y, Prins, Jan F *and others*. (2010, October). MapSplice: accurate mapping of RNA-seq reads for splice junction discovery. *Nucleic Acids Research* **38**(18), e178.

Wang, L., Wang, S. and Li, W. (2012). Rseqc: quality control of rna-seq experiments. *Bioinformatics* **28**, 2184–2185.

Witten, D M. (2011, December). Classification and clustering of sequencing data using a Poisson model. *The Annals of Applied Statistics* **5**(4), 2493–2518.