Methods

*Custom database generation for JUM junctions*. To generate database entries containing virtual splice peptides, splice junctions contributing to AS structures detected by JUM were first matched to transcript entries in the hg38 v98 Ensembl reference annotation as well as the assembled transcriptome from Strawberry. To be considered a match to a splice junction, transcripts were required to have exactly two flanking exons within a tolerance of 2nt. Then for each matched transcript, the cDNA sequence was extracted according to the reference human GRCh38 assembly. Three-frame translation was performed using the SeqinR package in R. To reduce the likelihood of including non-coding regions in our database, we discarded all translation frames which contained a stop codon upstream of the splice junction, but not if a start codon (methionine) lay between the upstream stop codon and splice junction. For each remaining translation frame, the virtual peptide sequence associated with the nucleotide sequence 50nt-upstream and 50nt-downstream of the splice junction was extracted and amino acids (AA) after the first stop codon were discarded. Furthermore, we required the resulting virtual peptide sequence to be longer than 17 AA in order to cross the splice junction. All remaining unique virtual peptides were used to create a FASTA file. The R script used to generate a custom peptide database for junctions using this method is available at github.

*Custom database generation for PSI-Sigma exons*. To generate database entries containing virtual splice peptides, exons detected by PSI-Sigma were first matched to transcript entries in the hg38 v98 Ensembl reference annotation as well as the assembled transcriptome from Strawberry. To be considered a match to an IR exon, transcripts were required to have exactly two flanking exons within a tolerance of 2nt. For non-IR exons, each transcript was required to have exactly one overlapping exon within a tolerance of 2nt. Then for each matched transcript, the cDNA sequence was extracted according to the reference human GRCh38 assembly. Three-frame translation and filtering of translation frames resulting in a stop codon upstream of the exon was done in a similar manner to that for JUM junctions. For each remaining translation frame, the virtual peptide sequence associated with the nucleotide sequence of the whole exon was extracted and AAs after the first stop codon were discarded. Resulting virtual peptides shorter than 7 AA were discarded. All remaining unique virtual peptides were used to create a FASTA file. The R script used to generate a custom peptide database for exons using this method is available at github.