Figure S1. APA MPRA Library Statistics. Related to Figure 1

(A) Distribution of mapped RNA-Seq read count (corresponding to cleavage position) across each random library UTR sequence. Only a region 50 bp upstream and 130 bp downstream of the proximal PAS CSE is displayed. Reads mapping to the distal (non-random) PAS farther downstream in the UTR are omitted. The location of the pPAS CSE is annotated and corresponds to position 0. The exact cleavage position of Alien2 transcripts could not be confidently determined from the RNA-Seq data due to differences in library prep and sequencing. For the purposes of this figure, all Alien2 reads mapping to a location somewhere in the [PAS + 5, PAS + 30] and [PAS + 95, PAS + 120] regions were uniformly distributed (to better visualize the cumulative amount of cleavage occurring at each site).

(B) (Top) Distribution of the total number of barcoded replicates covered by RNA sequencing of the designed MPRA library (The array containing wild type human PASs, ClinVar/HGMD/mutagenesis variant PASs and forward-engineered PASs). Each PAS was synthesized as 5 replicates with different random barcodes, and the entire experiment was replicated twice, resulting in a total of 10 possible replicates per PAS. The X-axis displays the number of sequenced barcode replicates and the Y-axis displays the number of distinct PASs. (Bottom) The distribution of RNA-Seq read depth per assayed PAS, pooled across each PAS's

replicates. The X-axis displays the pooled read count and the Y-axis displays the number of distinct PASs.
(C) (Top) The distribution of measured isoform proportion variance across barcode replicates per PAS in the designed MPRA. The X-axis displays the standard deviation of isoform proportion per PAS across replicates and the Y-axis displays the number of distinct PASs. The deviation is centered at zero with a low spread (Isoform replicate deviation < 0.1 for 99% of the MPRA). (Bottom) The distribution of measured average cleavage position variance per assayed PAS, pooled across barcode replicates. The X-axis displays the standard deviation in average cleavage position across replicates per PAS and the Y-axis displays the number of sequences.
(D) Scatter plot of measured isoform proportion (top) and isoform log odds (bottom) for the two experimental replicates of the designed MPRA library. The two replicates are in strong agreement (R^2 = .98).

Figure S2. Linear Model Comparison and Convolution Layer Motif Identification. Related to Figure 2
(A) Performance comparison between a 6-mer linear logistic regression model and APARENT's isoform predictions. The 6-mer model uses separate occurrence counts of every possible 6-mer (disregarding exact position) in the USE, CSE and DSE as features and is trained by an LM-BFGS optimization procedure to minimize proximal isoform proportion KL-divergence. Both models were trained on 9 of 12 UTR libraries. Scatter plots of predicted vs. observed isoform log odds are shown for the test sets of 4 libraries (TOMM5 constant DSE, TOMM5 random DSE, Alien1 and Alien2, n=8,000). R^2 is significantly higher on all test sets using APARENT (difference in R^2 ranges between 0.10 and 0.30 depending on library).
(B) Cross-validation test of isoform prediction accuracy on a subset of UTR libraries using APARENT. The DNN is trained on 9 libraries and tasked with predicting the proximal isoform log odds of 8,000 sequences from each of the held-out libraries *AARS*, *ATR* and *SOX13*.
(C) Log odds ratio analysis of 6-mers in the USE, CSE and DSE regions of the proximal PAS across the Alien1 and Alien2 UTR libraries. Effect size is measured as the natural log of odds of proximal isoform selection. For each effect size estimation, a 95% confidence interval is formed by 50-fold bootstrapping. A sequence logo was generated for the CSE hexamer by sampling k-mers according to the exponential of the corresponding log odds ratios.
(D) Illustration of the visualization and effect quantification method for the first convolutional layer. Pearson's coefficient of correlation is measured between layer 1 filter activations at every given position and proximal isoform logodds on the 3' UTR test set. Consensus sequence logos are generated by accumulating the top 5,000 subsequences from the test set that maximally activate each filter and stacking them into a PWM.
(E) RBP binding motif logos and position-specific effects for seven select layer 1 convolutional filters. Each convolutional filter PWM was cross-referenced against the Tomtom motif comparison tool to find the closest matching RBP. Position-specific effects are illustrated as heatbars (the heatbar ranges from CSE-25 to CSE+46).
(F) Additional Layer 2 convolutional filter sequence logos and position-specific effects.
(G) An identical set of layer 1 and 2 convolutional filter motifs as shown in Figure 2E and S2D-F were found when re-training the model with a different random initialization of the network weights, showing that the identified motifs are robust to network initialization.
(H) Illustration of the optimization procedure for maximally activating neurons in the dense (fully-connected) layer. The activation of a neuron (red) is chosen as the maximizing objective. A random start sequence is fed through the network, activating that neuron. The gradient of the neuron activation is propagated back through the network, such that the weight matrix of the sequence can be iteratively updated (see STAR methods).
(I) Gradient-based optimization performed on 50 random start sequences which maximize the activation of a particular dense neuron. The generated sequences are stacked into a PWM for

each neuron, and the average predicted proximal isoform use is obtained from APARENT. The sequence logos for a selection of four dense neurons are shown, ordered in descending order on the average proximal isoform use. To constrain variability, the sequences were encoded with the Alien2 background and a canonical proximal CSE.

Figure S3. Cleavage Site Regulatory Determinants. Related to Figure 3
(A) Illustration of the motif-effect visualization scheme used to interpret each filter's regulatory impact on cleavage site selection. Using a random sample of 120,000 sequences from the Alien1 library, each filter's activation at every position is recorded and correlated with the magnitude of cleavage at every given position, resulting in a two-dimensional plot that describes the regulatory impact at every cleavage site as a function of motif position.
(B) Additional convolutional filter motifs identified by the network to impact cleavage site selection, including the presumed binding site of CSTF (TGT[G/C]T), polyG and polyC.
(C) Log odds ratio analysis of 2-mers (dinucleotides) occurring in the DSE, where the effect size is measured as the natural log of odds of cleavage occurring at the given 2-mer. A sequence logo was generated for the cleavage site dinucleotide by sampling 2-mers according to the exponential of the corresponding log odds ratios.

Figure S4. Forward-Engineering of Polyadenylation Signals. Related to Figure 4
(A) Measured proximal isoform use (log odds) of the sequences optimized for different target isoform objectives. The violin plots display the generated sequences of each UTR library context separately (the UTR library context refers to the non-optimized background sequence surrounding the PAS). The 0% - 100% objectives refer to target isoform proportions. The 'Max' objective refers to sequences optimized for maximal proximal log odds (i.e., getting the sequences as close to 100% proximal use as possible). The 'Native' category refers to the 1,085 native human PAS sequences synthesized in the Array, and puts the strength of the generated sequences in relation to human polyadenylation. 10 sequences were measured for each target objective and library, except for the 'Max' objective where between 100 and 250 sequences were generated per library. Note that the 'Max' sequences from the Alien1, Alien2 and TOMM5 libraries had a measured mean isoform log odds higher than any of the human sequences. We optimize most of the USE and DSE for these libraries, whereas part of the DSE is fixed for other libraries.
(B) Predicted vs. measured proximal isoform use (log odds) for the generated sequences. All sequences are shown in the scatter plot to the left, and separated by UTR library context in the smaller scatter plots to the right. The color indicates the target objective of each generated sequence.
(C) Measured proximal isoform use (log odds) of the sequences optimized for maximal proximal preference, shown for the three held-out library contexts not trained on by APARENT.
(D) Selection of generated (and synthesized) sequences for the TOMM5 sequence context, ordered from top to bottom by increasing target isoform objective (0% - 100% and 'Max'). The sequence logos depict the optimized PWMs. The target and measured isoform use (mean log odds) are annotated in the top left corner of each logo. The line curves to the left illustrate the measured percentile of each optimized sequence with respect to the proximal isoform use of native human sequences. UTR sequence contexts (non-optimized sequence regions) are annotated with black letters in the sequence logos.
(E) Selection of generated (and synthesized) max isoform-sequences for 6 different UTR sequence contexts. The sequence logos depict the optimized PWMs. UTR sequence contexts (non-optimized sequence regions) are annotated with black letters in the sequence logos. The source UTR sequence context is annotated in the top left corner of each logo. All synthesized sequences of the optimized PWMs had a measured proximal use higher than any of the native human sequences synthesized in the same array.

(F) Predicted vs. measured average cut position for the sequences optimized for target cleavage. The color indicates target cleavage position downstream of the CSE. The three plots ('Original Target', 'Hardcoded AT' and 'Punish A-runs') correspond to different constraints or objective functions optimized for (see STAR Methods for a detailed description of the different objectives).
(G) Average measured cleavage profiles across the DSE for the optimized sequences, where the color indicates target cleavage objective. The dotted line indicates the mean native human cut position. The two plots ('Original' and 'Hardcoded AT') correspond to different constraints or objective functions optimized for (see STAR Methods for a detailed description of the different objectives). The third objective ('Punish A-runs') is omitted for brevity.
(H) Mean MFE of the sequences optimized for target cleavage. The X-axis denote target cleavage position downstream of the CSE. The MFE was predicted by NUPACK. The standard deviation is depicted by black error bars.
(I) A selection of sequence PWMs generated for maximal cleavage at target positions within the DSE of the Alien1 UTR. The three sequence logo columns ('Original Target', 'Hardcoded AT' and 'Punish A-runs') correspond to different constraints or objective functions optimized for (see STAR Methods for a detailed description of the different objectives). UTR sequence contexts (non-optimized sequence regions) are annotated with black letters in the sequence logos. Shown above each sequence logo are the measured (black) and predicted (red) cleavage profiles averaged across all 6 sequences sampled from the corresponding PWM. Each logo is annotated with the measured percentile of proximal use among native human pA sites synthesized in the same array. Shown below the sequence logos are the minimum free energy secondary structures predicted by NUPACK.

Figure S5. APADB Data Statistics and Model Training. Related to Figure 5
(A) The prediction accuracy on a held-out test set of alternative APADB sites, measured in R^2 between observed and predicted proximal isoform log odds, increases monotonically as the APA events are filtered on higher read count. Isoform predictions are made using APARENT tuned on a training set (75%) of APADB. Four separate tests, corresponding to increasingly higher read count filtering, are shown.
(B) The two curves depict the training and test error (mean squared isoform log odds error) on the APADB dataset (filtered on events with >= 1000 reads) when training a DNN model exclusively on a training set of APADB events. The training error eventually reaches zero, while the test error saturates at $R^2 = 0.34$. Interestingly, the test error does not eventually start to increase, indicating that overfitting is not an issue.
(C) When tuning APARENT's isoform predictions on the APADB dataset, the log distance between two competing APA sites is used as a linear feature. The odds ratio of the exponentiated distance weight (compared to a reference distance of 40 bases) is plotted as a function of site distance. The curve shows that at a site distance of 500 nt, the odds of selecting the proximal isoform over the distal isoform is approximately doubled compared to a site distance of 40 nt (all other effects equal). At a distance of 4000 nt, the odds are approximately three-fold.
(D) Bar chart of the average read depth per APA isoform (left) and total number of retained APA sites after filtering (right) per tissue/cell type in the APADB and Leslie datasets. We show these statistics for all APA events (green bars) and for a subset of APA events where both isoforms must have supporting reads in the given tissue (orange bars).
(E) The measured isoform log odds of all 3' UTR APA sites for a given tissue or cell type were correlated against the isoform log odds of all other tissues/cell types, resulting in an R^2 confusion matrix (visualized as a heatmap for both the APADB and Leslie data sources). All 3' UTR APA pairs with a total read count larger than 20 were included. The heatmap shows that in general tissues/cell types do not exhibit much differential behavior in the 3' UTR (mean R^2 =

0.97, mean difference in isoform proportion = 0.031). On average 3.3% of all 3' UTR APA events have an isoform difference > 0.25, but this set becomes even smaller with a larger read count threshold (2.5% using a total read count threshold of 100). The largest differential trends are observed for B Cells (for example, mean difference in isoform proportion compared to Brain = 0.052).

Figure S6. Human APA Variant Prediction and Non-Linear Variant Effects. Related to Figure 6

(A) Predicted vs. measured proximal isoform log fold change (log odds ratio) for every assayed human APA SNV (the same data as in Figure 6B). $R^2 = 0.64$ (n = 12,348).

(B) Correlation between predicted and measured isoform log odds ratios due to SNVs in the GEUVADIS dataset.

(C) The subset of human APA variants with statistically significant measured variant fold change (p = 0.00001 using a two-sided proportion difference test) where APARENT's predicted direction of change (whether the variant is up or down- regulatory) differs from the direction predicted by a linear logistic 6-mer regression model. Shown here are the few variants (n = 35) where APARENT predicts the direction incorrectly, as compared to Figure 6C where APARENT calls the direction correctly. As can be seen, the variants have mostly low measured effects (fold change < 2).

(D) The subset of human APA variants that result in either a gain (left column) or loss (right column) of a CSTF binding site (TGT[C/G]T). The top row shows predicted (color intensity) log fold changes made by APARENT while the bottom row shows predictions made by the linear 6-mer model. While the two models generally agree on the direction of change (Gain of CSTF in the DSE leads to upregulation, Loss of CSTF leads to downregulation), the correlation between predicted and observed fold change magnitudes are much higher using APARENT.

(E) Example SNVs in the PASs of RYR2 and INS that result in gain of CSTF and loss of CSTF respectively. Shown are the measured wildtype cut distribution (black), the measured variant cut distribution (red or green) and the predicted variant cut distribution (dashed blue). Interestingly, the measured direction of isoform fold change for each respective variant is opposite of the general trend observed in Figure S6F. RYR2: Creation of a DSE CSTF binding site leads to significant downregulation of isoform selection, by knocking out the de facto main cut site and promoting cleavage to a less used upstream cut site. INS: Destruction of a DSE CSTF binding site leads to more than 2-fold upregulatory effects on isoform selection, presumably because the CSTF site is too far away from the de facto cut site, rendering it non-functional and better suited as a polyT enhancer motif. Note that APARENT can call the alteration to the cut distributions accurately.

(F) The subset of human APA variants that result in either a gain (left column) or loss (right column) of a CFIm25 binding site (TGTA). The top row shows predicted (color intensity) log fold changes made by APARENT while the bottom row shows predictions made by the linear 6-mer model. The two models generally agree on the direction of change (Gain of CFIm25 in the USE leads to upregulation, Loss of CFIm25 leads to downregulation).

(G) Example SNVs in the PASs of INS and SNRNP200 that both result in gain of a CFIm25 binding site that partly overlaps the CSE (CPSF binding site). The fold change magnitudes are large (>1.7) but, even though they both result in competitive binding of CFIm25 and CPSF, they have opposite effects on APA. The INS variant results in strong upregulatory effects while the SNRNP200 variant downregulates selection.

(H) (Top) Scatter plot of measured wildtype isoform use (log odds) vs. measured log fold change, for the subset of variants resulting in gain of a CFIm25 binding site overlapping the CSE. (Bottom) Scatter plot of measured wildtype isoform use vs. 6-mer corrected log fold change, where the log fold change has been subtracted the effect sizes accounted for by USE 6-mer motifs of the linear model. Both plots suggest a trend where the effect of competitive

binding of CFIm25 and CPSF is net-positive if the wildtype PAS is weak, and net-negative if the PAS was already strong before creation of the CFIm25 binding site.
(I) Example SNVs in the PASs of COL3A1 and TMEM43 that both result in loss of a cryptic CSE hexamer in the DSE, where the net variant effects are downregulatory. COL3A1: The main wildtype cut site occurs over a cryptic CSE hexamer ATTAAA in the DSE. The variant knocks out the CSE hexamer while creating a new CA cut site. Interestingly, the cleavage occurring downstream of cryptic CSE is upregulated after its knock-out, indicating that the cryptic CSE was not a functional CPSF signal. TMEM43: The main wildtype cut site occurs over a cryptic CSE hexamer GATAAA in the DSE. The variant knocks out the CSE hexamer while creating a new CA cut site. Opposite the phenomenon observed in COL3A1, here cleavage at the main cut site is unaffected, while the cleavage downstream of the knocked-out CSE is significantly repressed, indicating that the GATAAA motif was in fact a functional CPSF signal. APARENT predicts these alterations to the cut distributions accurately.
(J) Example SNVs in the PASs of PEX5 and TGFBR2 that both result in strong downregulatory fold changes to isoform selection. By inspecting the MFE structures predicted by NUPACK, we see that the SNVs alter the folding hairpins such that they more tightly encapsulate the cut sites (PEX5) or change MFE folding structure entirely (TGFBR2).
(K) Example SNVs in the PASs of FOLR1 and ARSA that both result in loss of a used CA cut site, but where the net variant isoform effect has opposite direction. FOLR1: Wildtype cleavage occurs at two adjacent CA dinucleotides, and knockout of one of them results in strong downregulation of proximal APA. ARSA: Wildtype cleavage occurs at a CA cut site and at an adjacent cryptic cut site CT. Opposite to FOLR1, however, destruction of the CA cut site leads to an upregulatory fold change. APARENT predicts these cut alterations accurately.

Figure S7. Variants in Disease-Implicated Polyadenylation Signals. Related to Figure 7
(A) 10 SNVs that were synthesized in our experimental array and are annotated in ClinVar as having ‘Conflicting’ clinical significance. The X-axis illustrates the relative position of each variant within the corresponding PAS. The Y-axis shows measured isoform log fold change (log odds ratio) due to the variant. The color intensity corresponds to predicted isoform log odds ratio. The ClinVar variant identifiers are displayed in a table to the right.
(B) T->G SNV in the TP53 proximal PAS with highly repressive effects on isoform selection. The SNV knocks out the CSTF binding site (TGTCT) while also altering the hairpin folding structure of the DSE such that it tightly encapsulates the main cut site. The isoform use is measured to be downregulated >9-fold due to the SNV.
(C) Experimental and computational saturation mutagenesis of three disease-implicated PASs. The upper heatmaps display measured isoform log fold changes due to any given SNV, and the lower heatmaps display corresponding predicted log fold changes. Min- and max log fold changes are annotated to the right of each heatmap. The sequence logo letters are scaled by the mean log fold change across all possible nucleotide substitutions at each position. Benign (green), VUS (blue), Pathogenic (red) and Conflicting (orange) variants present in ClinVar or HGMD are annotated in each heatmap. Each of these are annotated with a variant identifier, the measured fold change and predicted fold change (in parenthesis). The three PASs have the following correlation between measured and predicted log fold change: (HBA2) R^2 = 0.79 (0.52 for non-CSE only), (INS) R^2 = 0.75 (0.62 for non-CSE only), (HBB) R^2 = 0.91 (0.56 for non-CSE only). They have the following occurrence frequencies of >2-fold variants outside the CSE: (HBA2) 15.8%, (INS) 7.47%, (HBB) 7.17%. The direction of change (up or down- regulation) of the >2-fold variants are predicted with 100% accuracy.
(D) Experimental and computational saturation mutagenesis of 3’ UTR PASs in ACMG genes (BRCA1 and TPMT). See caption for Figure S7C for a description of the heatmaps. The two PASs have the following correlation between measured and predicted log fold change: (BRCA1) R^2 = 0.87 (0.58 for non-CSE only), (TPMT) R^2 = 0.71 (0.50 for non-CSE only). They have the

following occurrence frequencies of >2-fold variants outside the CSE: (BRCA1) 10.23%, (TPMT) 8.87%. The direction of change (up or down- regulation) of the >2-fold variants are predicted with 100% accuracy.
(E) Example SNVs in the PASs of the BRCA1 and TPM1 gene that alter the cleavage distribution. Shown are the measured wildtype cut distribution (black), the measured variant cut distribution (red) and the predicted variant cut distribution (dashed blue). BRCA1: A new cut dinucleotide TA is created between two active CA cut sites, shifting the mode of cleavage from a bimodal to a unimodal distribution, for a significant (>1.3 fold) net loss-of-function. TPM1: The wildtype TA cut site is transformed into the canonical CA dinucleotide and shifted by 1 nt, which results in a >1.7-fold measured loss of function.

Movie S1. Max Isoform Optimization with SeqProp. Related to Figure 4
For each sublibrary, the model was initialized with random USE and DSE regions. The sequence was then optimized through 6000 iterations of SeqProp for maximal expression of the local isoform. The animation shows every 10th iteration. The CSE is at position 50.

Movie S2. Target Cleavage Optimization with SeqProp. Related to Figure 4
Sequence convergence of the Alien1 library when optimized for a specific cleavage position (animation progresses through increasing distance; target distance relative to the CSE displayed in title). The line plot (below each logo) shows the relative usage for each position. The CSE is at position 50.

CONTACT FOR REAGENT AND RESOURCE SHARING

Further information and requests for materials should be directed to and will be fulfilled by the Lead Contact, Georg Seelig (gseelig@uw.edu).

EXPERIMENTAL MODEL AND SUBJECT DETAILS

Human Embryonic Kidney cells (HEK293; female) (R70007, Thermo Fisher Scientific) were cultured in DMEM high glucose, pyruvate (ThermoFisher Scientific, 11995-065) supplemented with fetal bovine serum (Atlanta Biologicals, S11150) and penicillin/streptomycin antibiotic.

METHOD DETAILS

Massively Parallel Reporter Assay

Cloning
For the *TOMM5* library, a vector was constructed by replacing the bGH pA signal with an IDT gBlock designed with the *TOMM5* 3' UTR sequence. The vectors were then linearized at the site of randomization and assembled with Klenow-extended degenerate oligos that overlapped at the pPAS site and included compatible overhangs. For all other libraries, two oligos were designed to anneal at the pPAS and included 25-45 bases of randomization and UTR-specific overhangs. Matching overhangs specific to each UTR were added to a linearized reporter by inverse PCR. All libraries were constructed using Gibson Assembly. Library sizes were estimated by plating a small amount of transformation and extrapolating based on colony counts. The remaining transformants were grown in 50mL overnight culture. Individual plasmid libraries, and individual clones from each library, were Sanger sequenced to confirm the expected structure and diversity.

Cell Culture and RNA Extraction
HEK293 cells were grown on ECM-coated plates in DMEM supplemented with 10% FBS and antibiotics. Cells were trypsinized, washed with PBS, and lysed 36-48 hours post-transfection. mRNA was purified with the NEB polyA purification kit that utilizes T20-linked, magnetic beads.

Sequencing Library Construction
Polyadenylated RNA was reverse-transcribed with an anchored polyT primer containing Illumina adapter sequences and a unique molecular identifier (Adapter-UMI-T18VN). Library cDNA was then amplified using a library-specific forward primer containing additional Illumina adapter sequences (Adapter-Library-FWD) and reverse primer matching the adapter sequence added during RT. Amplification was conducted and monitored with a qPCR instrument and stopped early to minimize PCR biases.

Barcoding and Mapping
For libraries with PASs relatively close together, we sequenced and searched 5' to 3' across both signals for the site of polyadenylation. For longer UTRs, we identified the polyadenylation site via reverse-complement mapping of sequence adjacent to the paired-end poly-T read.

To record full library sequences, including those downstream of the proximal cleavage site, we sequenced amplicon prepped directly from the plasmid library. The first 20-25 degenerate bases upstream of the proximal PAS was used as a barcode to enable mapping of cleaved mRNA back to the full UTR sequence. Sequencing reads were long enough to precisely locate cut sites for all proximal and some distal isoforms. The resulting dictionary of sequences was used to map reporter RNA back to the originating plasmid sequence.

Defining the Site of Polyadenylation
For all libraries, with the exception of TOMM5 and Alien2, both proximal and distal polyadenylation sites were identified with a single, sense-strand sequencing read that primed directly upstream of the randomized USE. We used the open-source adapter trimming software package cutadapt v1.15 (Martin 2011) with polyA trimming parameters (-a 'A'*18 -m 2). For TOMM5, exact polyadenylation positions were identified only for the proximal isoform while distal cleavage was inferred from reads that lacked polyadenylation within the proximal DSE. For Alien2, the sequence downstream of T18 in the anti-sense read was mapped back to full-length library members.

Data Processing
Having collected sequencing reads from all MPRAs, the raw data was filtered, clustered and transformed into a set of well-defined 3' UTR libraries. Full-length RNA reads were filtered on a sufficiently high quality, after which they were clustered on the randomized region upstream of the pPAS, resulting in a dictionary of sequence variants for each library. The dictionary was further expanded by sequencing the plasmid library to include members that expressed no distal isoform. RNA reads were mapped to each respective dictionary entry by matching the upstream region with shortest hamming distance.

For each mapped read, the Polyadenylation cleavage site was determined by scanning the read for the Poly-A tail (cleavage site distributions shown in Figure S1A). For some libraries, reads can map to a non-degenerate distal site found 2-300 bases downstream of the degenerate proximal site. The cleavage positions of all the mapped reads for one particular sequence variant was stored as a vector associated with that variant. Reads mapping to far-away (non-random) distal sites were recorded in a special "distal" position of each count vector. The resulting dataset consisted of a per-library dictionary of unique sequence variants with associated cleavage-position count vectors. The library datasets were passed through a final filtering step to select for high-confidence variants. This step removes sequences that are supported by fewer than 10-20 unique-UMI RNA reads (the exact number depends on the specific library and its read coverage), or sequences that contain >75% A-nucleotides in a 12-20bp region (to safeguard all libraries against internal priming artifacts).

Human Wildtype, Variant & Engineered Sequence Array

In addition to the 12 random 3'UTR APA libraries, we constructed an MPRA of designed (not random) sequences. The designed MPRA consisted of (1) all human wild type PAS sequences from the reference genome (n = 1,085) having at least one annotated variant in the ClinVar database, (2) SNV and InDel variants of ClinVar and HGMD occurring within 50 nt upstream or 100 nt downstream of an annotated PAS (n = 1,811), (3) SNVs from saturation mutagenesis of disease-implicated human PASs (n = 10,537), and (4) de-novo engineered sequences generated by SeqProp (n = 2,740).

Construction
All curated PASs were synthesized in a custom oligo array with 50 nt of USE sequence and 107 nt of DSE sequence from the original UTR context. To robustly identify variants after sequencing, we included 20 nt random barcodes 70 nt upstream of the CSE. To average out barcode effects, we synthesized 5 replicates per sequence with unique barcodes. The PASs were inserted into 3' UTR reporters upstream of a distal PAS, expressed in HEK293 cells on plasmids, and their APA profiles were measured by 3' sequencing (Figure S1B). The array was experimentally replicated twice with identical barcodes. The measured APA estimates agreed

well among replicates with distinct barcodes (Figure S1C), with almost perfect correlation between sequences with identical barcodes measured in the two separate experimental replicates (Figure S1D).

See the STAR methods section related to each particular experiment (Forward-engineering or Human variant analysis) for a description of how the MPRA measurements were used in downstream analyses.

APARENT Model (Isoform Prediction)

We trained two architectures for the APA prediction network (APARENT, APA REgression NeT), one that predicts the total relative isoform abundance (as a proportion) of polyadenylation occurring anywhere within +10 to +35 nt downstream of the CSE start position, and one that predicts the per-nucleotide probability of cleavage and polyadenylation occurring across the input sequence. Here we describe the Isoform prediction model, as presented in Figure 2A.

Architecture
APARENT is based on a Convolutional Neural Network (CNN). The input DNA sequence is aligned such that the CSE hexamer of the proximal PAS starts at position 50. The sequence is transformed as a 1-Hot-Coded matrix, i.e. where nucleotide A is encoded as the vector [1, 0, 0, 0], C is encoded as [0, 1, 0, 0], G is encoded as [0, 0, 1, 0] and T is encoded as [0, 0, 0, 1]. The coding can best be thought of as a 1-dimensional signal (over sequence position) spanning 4 signal channels (nucleotides). The input sequence is 186 nt long, a number that was chosen such that all randomized regions of all 3' UTR libraries would fit the input window.

The CNN was constructed from two convolutional layers using ReLU activation functions, interlaced with a MaxPool layer. The first convolutional layer had 70 1-dimensional filters covering all 4 nucleotide input channels. The filters of the first layer were 8 positions (nucleotides) wide. The second convolutional layer had 110 1-dimensional filters covering all 70 output channels from the previous layer, where the filter width was set to 6. The MaxPool layer subsampled the 1D signal by a factor of 2. The flattened convolution output is passed to a fully connected layer of 80 hidden ReLU units with 0.2x dropout, which finally connects to a logistic regression node that outputs the predicted proximal isoform proportion. To account for non-sequence-related cross-library variation, a vector of bias weights indexed by the source UTR library was added to the final regression layer.

Note that in the base version of APARENT, only the sequence covering the proximal PAS is actually used as input, not the distal PAS. Since APARENT is trained on UTR libraries where only the proximal signal is randomized, the distal signal will contribute a constant, additive bias towards distal selection. Of course, each UTR library has a different distal context, which is why we capture each library-specific distal bias in different intercept terms of the final regression layer. Here we assume PASes are regulated independently when separated by >186 nt, i.e. when they are not both covered in the CNN input window. At shorter distances, competing signals may very well act non-linearly upon each other (e.g. by competitive binding of overlapping USE/DSE elements), however the neural network is able to capture such nonlinearity as it is covered in the input window.

Training
Of the 12 random 3' UTR libraries, 9 were used for training while 3 libraries were held out entirely. 95% of the data from these 9 libraries was used for training (~2.4M sequences), 2% for

validation (~50,000 sequences) and 3% for testing (~80,000 sequences). The four held-out libraries were HSPE1, SNHG6, WHAMMP2, and the non-randomized human wild type PAS library.

The quality of the measured (target) isoform proportions depend heavily on the RNA sequencing read depth from which they were estimated. We wanted to keep high-quality measurements for the test set, but we also wanted some fraction of high-quality estimates in the training set. Additionally, we wanted an equal number of sequences from each library in the test set. To this end, we devised an elaborate scheme to shuffle the 9 libraries together. First, each library was shuffled individually:

1. The sequences of Library L were sorted in ascending order on total read count.
2. Every odd-numbered sequence in the sorted order was deposited in bucket L_A. Every even-numbered sequence was deposited in bucket L_B. The sort order was then altered by concatenating the entire subset L_B after L_A. As a consequence, half of the high read count-sequences are placed in the lower half of the library sort order and the other half is placed in the upper half. This operation balances having high-read count variants in both training- and test sets.

The libraries were then shuffled together by placing one sequence from each library after another in round-robin order, traversing all libraries simultaneously in descending order of read count and inserting the sequences in the combined dataset in descending order. As a consequence, the back of the combined library (where the test set will be taken from) contains an equal number of high read count sequences from each library, thus balancing the test set with regard to library representation. The round-robin placement is aborted once any library runs out of sequences. Remaining sequences are placed in chunks in the beginning of the combined library. Finally, the training set was shuffled.

The network was trained by SGD (mini-batch stochastic gradient descent) to minimize the mean symmetric KL Divergence against the observed isoform proportions of the training data. Specifically, the loss function per sequence was defined as:

$$L = p_{Prox} * \log(p_{Prox} / y_{Prox}) + (1 - p_{Prox}) * \log((1 - p_{Prox}) / (1 - y_{Prox})),$$
where p_{Prox} and y_{Prox} are observed and predicted proximal isoform proportions.

We used the python library Theano for training (The Theano Development Team et al. 2016). Training was limited to 10 epochs, but was halted prematurely by Early Stopping evaluated on the validation set. The training set was re-shuffled after each training epoch, which typically improves mini-batch SGD. The learning rate in SGD was chosen as 0.1; we empirically lowered the learning rate starting from 1.0 until we stopped observing oscillating behavior in the loss function on the training data.

We tried several hyper-parameter settings (convolution filter width. number of convolutional filters, number of fully-connected dense layer neurons, etc.) and tested these re-trained models on the validation set, but found no difference in performance as long as the network was reasonably large. We settled on a configuration that had slightly more convolutional filters than what was needed to not increase the validation loss.

Evaluation

APARENT was evaluated by predicting the proximal isoform proportion of the test set sequences from the trained-on UTR libraries (n > 80,000). We compared the predictions against the measured proportions as Log Odds values (log p / (1 - p)), as this representation is more suitable to measure correlation between (in Figure 2B). Correlation was calculated as R^2 = 1 - SSE/SST, where SSE is the sum-of-squares error between predicted and measured isoform log odds, and SST is the variance of the measured isoform log odds. We calculated R^2 on the joint set of libraries and on each library separately, obtaining min, mean and total correlation metrics. While we noted a difference in R^2 between library test sets, the mean prediction error (SSE) was nearly constant, meaning the R^2 statistic naturally increases or decreases depending on the amount of variance in the library-specific data (some libraries have relatively low variance, e.g. in two of the TOMM5 libraries where the DSE is not randomized).

We similarly predicted and compared proximal isoform log odds on the four held-out UTR libraries (in Figure 2D), but we define the correlation metric differently. Here R^2 is calculated as Pearon's r squared. This definition is more suitable for UTR libraries not trained on, as the context and bias of an unseen library may cause our predictions to be off by a constant intercept term. Rather, with this metric we are strictly evaluating whether APARENT's predicted isoform log odds covary with the measurements. This is the same evaluation metric we used when training 6 separate CNNs on each individual library (All 4 TOMM5 libraries were trained on with one CNN) and testing each network on the test sets of other libraries (in Figure 2C).

Convolutional Layer 1 Motif Analysis

The position-specific effect of each filter in the first convolutional layer was measured by computing correlation coefficients between filter activations at every given position and the proximal isoform predictions (Figure S2D). The test sequences of the Alien2 library were passed through the network, predicting the proximal isoform log odds for each sequence. The activations of each filter at every position j were recorded per sequence. The Pearson r coefficient was then calculated between filter activations at position j and isoform log odds predictions across the test set. The coefficient values were used as color intensities for the corresponding positions in the filter heatmap.

To generate consensus sequence logos for the filters representing their maximal activation, the 5,000 input subsequences of the test set that resulted in maximal filter activation were stacked into a position weight matrix (PWM) and used to generate a sequence logo for each filter (Alipanahi et al. 2015). By purposefully biasing the selection of sequences to be maximally activating examples, the consensus logos effectively visualize the canonical form of the motif learned by a filter. To visualize mean filter sensitivity (used to score CSE variants in Figure 2F), we randomly sampled 40,000 sequences from the Alien2 library, and their contribution to the PWM was scaled by the filter response. The generated sequence logos were cross-referenced against published binding data using the Tomtom comparison tool (Gupta et al., 2007). All motif matches shown in Figure 2/S2 had a p-value less than 10^-4.

The filter motifs were validated directly in the data using a log odds ratio analysis, where every possible 6-mer occurring in the USE, CSE, and DSE were scored and ranked according to the expected increase in odds ratio of proximal isoform selection (Rosenberg et al., 2015). For each 6-mer S, the log odds ratio LOR(S) is calculated as:

$$LOR(S) = \log((p^{S}_{AVG} / (1 - p^{S}_{AVG})) / (p^{\hat{}S}_{AVG} / (1 - p^{\hat{}S}_{AVG})))$$

where p^{S}_{AVG} is the average isoform proportion of library sequences containing S in the region of interest. $p^{\hat{}S}_{AVG}$ is the average proportion of sequences not containing S. In Figure S2C, the

natural log is used. To estimate the certainty in the calculated values, we used 50-fold Bootstrapping with replacement to obtain a 95% confidence interval.

A similar analysis was performed to estimate the odds of cleaving at each of the 16 dinucleotides (Figure S3C). For each sequence we recorded the positions in the cut distribution with non-zero cleavage probability. The average cleavage probability pc^{S}_{AVG} was calculated from the subset of cleavage probabilities that were recorded with dinucleotide S. Similarly, $pc^{\wedge S}_{AVG}$ was averaged from the remaining set of cleavage probabilities. These estimates were passed to the log odds ratio analysis LOR(S) as defined above.

Convolutional Layer 2 Motif Analysis
We extended the method used in the first convolutional layer to measure position-specific effects and generate consensus sequence logos for the second layer of filters. The idea is that the first layer will capture short motifs or "sequence building blocks" with lengths less than the filter width (8 nt), and the second layer will combine these features into longer, yet still local, sequence determinants (spatially close motif combinations or longer motifs). The procedure is very similar to the original method: Alien2 UTR library sequences are passed to the network, predicting proximal isoform log odds while simultaneously recording layer 2 filter activations at every position per sequence. The isoform predictions are correlated with the activations of each filter at every position using Pearson's r. The 5,000 subsequences of the test set resulting in maximal layer 2 filter activation were stacked into a PWM, generating a maximal-activation sequence logo for each filter. Note that our layer 2 filters are 6 positions wide, and between the two layers is a MaxPool layer with subsampling factor 2. Hence, our layer 2 filters cover a window of 6 (layer 2 filter width) x 2 (layer 1 subsampling) + 8 - 1 (layer 1 convolution extension) = 19 nt of the input sequence, meaning they can detect up to 19 nt wide motifs.

APARENT Model (Cleavage Prediction)

The second APARENT architecture predicts the entire probability distribution of cleavage occurring across the input PAS (in Figure 3A). The model predicts one cleavage probability per nucleotide position in the input sequence. It also predicts the probability of distal isoform selection (the total proportion of distal isoform), which means the per-nucleotide proximal cleavage probabilities are relative to both the total proximal and distal cleavage. The model is a generalization of the proximal isoform prediction architecture, since the total proximal isoform proportion can be calculated by aggregating the per-nucleotide cleavage probabilities within the region of interest.

Architecture
The cleavage prediction network has an identical architecture and parameter setting (number of filters, hidden neurons, etc.) as the isoform prediction network, except for the output layer which is now a 187-way Softmax function (multinomial probability layer) rather than a scalar sigmoid function. The 186 first Softmax probabilities are trained to predict the cleavage proportions of the 186 nucleotides in the input sequence. The 187-th probability predicts the remaining proportion of polyadenylation outside the sequence window (i.e. the distal isoform).

Library bias terms are handled a bit differently for the cleavage prediction network. For the isoform network, the output layer had one scalar intercept term per library (capturing the mean isoform log odds of each library). Here, all of the 187 Softmax outputs have their own library intercept terms, each capturing the library-specific cleavage bias at the given position.

Training
The network was trained by minimizing the KL divergence between predicted and observed cleavage probability distribution of each sequence. Specifically, the loss function per sequence was defined as:

$L = (\sum_{k=1 \text{ to } 186} p_k * \log(p_k / y_k)) + p_{dist} * \log(p_{dist} / y_{dist})$,
where p_k and y_k are observed and predicted cleavage probabilities per position in the input sequence. p_{dist} and y_{dist} are the total distal isoform proportions. Note that y_1, …, y_{186}, y_{dist} are predicted from one Softmax layer, meaning $(\sum_{k=1 \text{ to } 186} y_k) + y_{dist} = 1$.

Because raw cut data was missing (or removed due to potential internal priming artifacts) at certain positions depending on library, each specific library was used to train the model only on a subset of softmax probability outputs. In particular, for Alien2 the model could not be trained on the cut positions at all due to the lack in knowledge of exact cut position. Instead, for Alien2 sequences the loss function was replaced by the proximal isoform proportion KL divergence (the isoform network objective). The same training, validation and test splits as was used for the isoform prediction network was used to train this model.

Evaluation
The model performance was evaluated by predicting the cleavage distributions of the Alien1 and WHAMMP2 test sets (~10,000 sequences from each library) and comparing the mean predicted cut positions against observed mean positions (in Figure 3B). The mean cut position for a sequence is computed as the dot product between a position vector (the vector [1, 2, …, 186]) and the cleavage distribution. The correlation was measured by R^2 (Pearson r squared) between predicted and observed mean position.

We also compared the proximal proportion predictions of the isoform network against the aggregated cut probabilities predicted by the cleavage network (in Figure 3C). The predicted cut probabilities were aggregated in the same range that the isoform network's proximal isoform proportion targets were estimated from. Correlation (Pearson r squared) was measured between the log odds of these two quantities.

Convolutional Layer 1 Motif Analysis
We estimated convolutional filter consensus logos and position-specific effects of the first layer with a method similar to the isoform network filter estimation (in Figure 3E, S3A-B). The main difference is that, since the output predictions are multinomial cleavage site probabilities, we are now correlating every filter activation position with multiple outputs rather than one output (the total isoform proportion). We applied APARENT to a random sample of 120,000 sequences from the Alien1 library, predicting their entire cleavage distribution. Each Layer 1 filter activation at every position was recorded and correlated with the predicted log odds of cleavage at every other position (using Pearson's r), resulting in a two-dimensional heatmap of correlation intensity per filter. For a given filter, it's corresponding heatmap at coordinate (x, y) describes the correlation between the filter firing at start position x and cleavage occurring at position y. Maximal-activation sequence logos were generated for each filter as previously described, using a sample of 5,000 subsequences from the Alien1 test set that maximally activate a given filter.

Linear Logistic 6-mer Regression Baseline

As a baseline comparison to the neural network model (in Figure S2A), we performed linear logistic regression on the combined library dataset. We also used this regression model to

search for disagreeing SNV calls compared to APARENT when investigating complex human variants (in Figure 6C, S6C, S6D, S6F).

Architecture
The output response variable was the proximal isoform proportion (same as for the isoform prediction neural net). The input features consisted of 6-mer occurrence counts in the USE, CSE and DSE regions, and a 1-Hot-coded vector of each sequence variant's origin library.

Specifically, the input feature matrix X was constructed as follows: For each sequence, the USE, CSE and DSE regions were separately scanned with a 1-nt stride for 6-mer sequence motifs. For each 6-mer, the corresponding occurrence count in the input matrix X was incremented by one. Finally, a binary library indicator feature was set to one (and encoded in matrix X) to indicate source library for each sequence.

The logistic regression model consists of a weight vector $w = (w_1, \ldots, w_m)$ of length m and a scalar bias term w_0, where m is the input feature dimensionality (the total number of 6-mer features and library indicators). The predicted proximal proportion y_{Prox} is computed as the sigmoid activation of the weighted sum of input features:

$$y_{Prox} = 1 / (1 + e^{-(w0 + w1 * x1 + \ldots + wm * xm)}),$$
where $x^i = (x^i_1, \ldots, x^i_m)$ is the input feature vector of sequence i

The regression weights corresponding to the library indicator features encode individual per-library intercept terms that absorb the library-specific mean proximal isoform log odds, effectively debiasing the 6-mer regression weights.

Training
The model is trained by minimizing the mean proximal proportion KL divergence (same as for the isoform prediction neural net):

$$L = p_{Prox} * \log(p_{Prox} / y_{Prox}) + (1 - p_{Prox}) * \log((1 - p_{Prox}) / (1 - y_{Prox})),$$
where p_{Prox} and y_{Prox} are observed and predicted proximal isoform proportions.

Training is done using an LM-BFGS optimization procedure from the python package SciPy (Jones et al. 2016). The model was L_2-regularized (imposed with a loss-term $\lambda * |\mathbf{w}|^2$) and cross-validated to find a suitable λ, however the optimal parameter value found was 0. The same training, validation and test splits used for APARENT were used to train this model.

Evaluation
Similar to the evaluation of the isoform prediction network, we tested the 6-mer regression model by comparing the predicted proximal isoform proportions against the library measurements on a log odds scale (in Figure S2A). Correlation was computed as R^2 = 1 - SSE/SST, where SSE is the Sum-of-squares error and SST is the library variance.

Generation of pA Sequences

An iterative gradient ascent optimization procedure was used to computationally generate PAS sequence PWMs from the pre-trained neural network which conformed to user-defined target objectives and constraints (Figure 4A). It is an extension of a popular image recognition method used for visualizing input features that the network is sensitive towards.

This section describes the computational method for generating sequences. See STAR Methods subsection “Human Wildtype, Variant & Engineered Sequence Array” for a detailed description of the construction, synthesis and experimental measurement of the generated sequences.

Optimization Procedure

The original computer vision method optimizes a randomly initialized start image to maximally activate a target neuron in the neural net by performing gradient ascent on the input image (Szegedy et al., 2014; Simonyan et al., 2013; Olah et al., 2017). Consider the activation $A^L_j(I)$ of a neuron j within network layer L when receiving input pattern I. The objective is to maximize $A^L_j(I)$ by optimizing I. I.e., the goal is to find:

$$I_{MAX} = argmax_I(A^L_j(I))$$

I is directly optimized by treating its elements as free weights and applying backpropagation of error through the CNN (Simonyan et al. 2013). In the context of computer vision, I is an unconstrained image with pixel values in range (-inf, +inf). This exact method has previously been used in genomics to visualize TF binding motifs, by viewing the optimized “image” as a dense One-hot sequence from which a TF binding PWM is extracted (Lanchantin et al., 2016).

We extended this method to make it suitable for de-novo generation of pA sequences while also providing richer optimization controls compared to the vanilla algorithm. Specifically, our algorithm (SeqProp) applies a self-normalization step to the input pattern during optimization, making the generation more robust. It also provides direct control of the PWM entropy via the objective function, enabling generation of both high and low- temperature PWMs. It further allows users to specify both ‘hard’ and ‘soft’ sequence constraints: ‘hard’ constraints are defined as non-optimized sequence regions hardcoded into the optimization (e.g. to have a hardcoded AATAAA at position 50 in the sequence). The ‘soft’ constraints are defined in the objective function (e.g. to penalize generating polyG in region 50-90 in the sequence).

In summary, the SeqProp algorithm takes a randomly initialized sequence PWM and iteratively optimizes it by making small changes to all nucleotides at the same time according to the target APA objective (as a continuous relaxation to sequence optimization). The output PWM has low entropy over nucleotides that strongly affect the objective and high entropy over less impactful regions. Thus, we can sample multiple sequences that satisfy the objective by sampling according to the PWM. The detailed steps of the algorithm are as follows:

1. We initialize a random weight matrix W of the same shape as a One-hot-coded input sequence. We also encode a mask matrix with all 1’s in the nucleotide positions (columns) to be optimized and 0’s in fixed locations (where we want to inject a constant sequence region). Similarly, we encode a template matrix with all 0’s in the nucleotide positions (columns) to be optimized and a proper 1-Hot coding in fixed locations (1-Hot-coded according to the desired fixed sequence content).
2. Next, different from DeepMotif, we modify APARENT’s network topology by placing a column-wise Softmax layer on top of the input matrix W, turning it into a differentiable sequence distribution, or PWM. By directly integrating the PWM into the network during optimization, we impose a self-regularizing effect on the optimization by constraining the

input magnitudes sent to the network, and it gives us direct control of the sequence distribution entropy via the objective function.

3. We multiply the PWM (Softmax(W)) element-wise with the mask matrix, effectively zeroing columns not to be optimized. We add the template matrix to the masked PWM, effectively injecting a fixed sequence context that will not be updated in the backpropagation step.
4. [Optional Step] We apply a multinomial sampler to the PWM, in order to sample a proper, 1-Hot-coded, input sequence following the PWM distribution. To make this step differentiable, we use a Straight-Through Estimator and pass the gradients of only the sampled nucleotide channels back to the PWM (ref straight through estimator, stochastic binary networks). The sampled, discrete, input now follows the input specification of APARENT.
5. The PWM (or the sampled 1-Hot sequence) is passed as input to APARENT. APARENT predicts the proximal isoform proportion y_p and cleavage distribution y_1, …, y_{186}.
6. Finally, we form an objective function and perform backpropagation through the network to update the PWM weights according to the objective function gradient. Our objectives involve three components: the APARENT isoform or cleavage output (y_p or y_1, …, y_{186}), a target entropy for the final PWM (defined on Softmax(W)), and 'soft' sequence constraints (defined on Softmax(W)).

Step 4 is a mechanism for guarding the predictor network (APARENT) against spurious input; the continuous PWM could contain high-entropy nucleotides that, when directly passed as input to the network, would activate multiple orthogonal convolutional filters that in a 1-Hot setting never fire together. effectively creating activation artifacts that the network has never encountered. However, for all the objectives we tried when optimizing pA sequences, we found that this step was not necessary. Especially when optimizing the PWM for a low entropy in the objective function, the PWM is slowly pushed to an almost-1-Hot-coded pattern anyway.

Objective Functions

The forward pass can be summarized as:

$$y_p, y_1, \ldots, y_{186} = Net(Sample(Softmax(W) * Mask + Template))$$

where Net is the Neural network forward pass
Sample is the (optional) multinomial 1-Hot sequence sampler
Mask is the matrix that zeros non-optimized positions
Template is the matrix that injects sequence at non-optimized positions
W is the weight matrix to be optimized

The backward pass can be summarized as:

$$Calculate\ Loss = Objective + Entropy + Constraints$$
$$Calculate\ dLoss/dW$$
$$W^{(t+1)} = W^{(t)} + \eta * dLoss/dW$$

The 'Objective' is defined in terms of isoform or cleavage divergence against the target:

$$Isoform\ Objective = KL(y_p \mid t_p)$$

$$Cleavage\ Objective\ = KL(y_{1,\dots,186} \mid t_{1,\dots,186})$$

The 'Entropy' is controlled by a Sum-of-squares error between the PWM Shannon entropy and a target entropy T (T is a vector of target entropies per nucleotide):

$$Entropy\ = \sum_{i=1}^{186}(-\sum_{j=1}^{4} Softmax(W)_{ij}\ *\ log(Softmax(W))_{ij}) - T_i)^2$$

The 'Constraints' (which refers to 'soft' sequence constraints) are controlled by repeated shifted element-wise multiplications of the PWM. For example, to punish the generation of GGGG in the 60 to 90 nt region, the formula is defined as (where we denote S = Softmax(W) for brevity):

$$Constraint\ = \sum_{i=60}^{90-3} S_{i,2} * S_{i+1,2} * S_{i+2,2} * S_{i+3,2}$$

By repeatedly shifting S by one position and multiplying the corresponding channels (G = channel 2), we effectively get a large positive number for $S_{i,2} * S_{i+1,2} * S_{i+2,2} * S_{i+3,2}$ if and only if the PWM at position i to i+3 has a high frequency of GGGG.

Generated Target-Isoform Sequences

We generated and synthesized pA sequences for 6 different target isoform expression levels: 0%, 25%, 50%, 75% and 100% (we minimized KL-divergence against the proportions 0, 0.25, 0.5, 0.75, 1.0). The sequences were also optimized for minimal PWM entropy. The objective loss function was defined as (where y_p and t_p are predicted and target proximal proportions respectively):

$$Loss\ = KL(y_p \mid t_p)\ - \sum_{i=1}^{186}\sum_{j=1}^{4} Softmax(W)_{ij}\ *\ log(Softmax(W))_{ij}$$

10 randomly initialized sequences were generated per target proportion. The generation procedure was repeated 6 times, each time changing the non-optimized background sequence to one of the trained-on UTR library sequence contexts (as 'hard' sequence constraints). An example generation for TOMM5 is shown in Figure S4D.

We noted that SeqProp could generate much higher predicted expression levels (compared to the 100% KL-divergence objective) if we directly maximized the proximal class score just before the sigmoid output of APARENT. We optimized 20-50 PWMs per library background sequence for maximal score (and target entropy = 1.8 bits) and sampled up to 10 sequences from each PWM. The objective loss function was defined as (where s_p is the logit score of y_p):

$$Loss\ = -s_p\ + \sum_{i=1}^{186}(-\sum_{j=1}^{4} Softmax(W)_{ij}\ *\ log(Softmax(W))_{ij}) - 1.8)^2$$

An example generation for 6 libraries is shown in Figure S4E. We also optimized sequences for maximal preference with previously unseen UTR library backgrounds (contexts not trained on).

Generated Target-Cleavage Sequences
We generated and synthesized pA sequences for 9 different target cut positions: +5, +10, +15, +20, +25, +30, +35, +40, +45 nt downstream of the last nucleotide of the CSE. The objective was to direct and maximize specific cleavage to those positions (implicitly suppressing cleavage at other positions). The sequence were optimized with 1.8 bit target entropy. The objective loss function as defined as (entropy terms left out for brevity):

$$Loss = KL(y_1, \ldots, y_{186} \mid t_1, \ldots, t_{186})$$

The target vector t_1, …, t_{186} is set to all-zeros except for the target cut position, which is set to 1.

5-10 PWMs were generated at each target cut position, using the Alien1 sequence template as background. 6 sequences were sampled and synthesized for each PWM. During generation, we noted that APARENT favors cleavage over poly-A runs. Due to the 3' sequencing protocol, exactly where cleavage occurs is ambiguous (we cannot't observe at which of consecutive A-nucleotides polyadenylation happened). To show that we can optimize cleavage unambiguously, we repeated the entire experiment 2 times with additional constraints: (1) The initial sequence has a hard-coded T downstream of the target cut, and (2) the objective function punishes downstream poly-A runs (using a 'soft' constraint as previously described). A selection of optimized sequences for each of the three repeated experiments are shown in Figure S6I.

APARENT Fitted To Native APADB and Leslie Data

APADB is a publicly available database of APA site annotations in the human genome, determined experimentally from 3' mRNA sequencing (Müller et al., 2014). The data also includes RNA-Seq read counts mapped to each APA isoform for a number of different tissues. Similarly, the Leslie APA atlas contains 3' RNA-Seq data mapped to the genome for different human cell types (Lianoglou et al., 2013). We used these two datasets to develop and train an extension of APARENT capable of predicting the isoform proportions between adjacent APA sites (in Figure 5).

APADB Data
We downloaded the latest version of the pooled-tissue human APADB dataset (Homo sapiens v2) from (http://tools.genxpro.net/apadb/download/), which contains both an APA site annotation in genome coordinates and isoform read counts aggregated over all tissues. To obtain the same data but separated by individual tissues, we wrote a web scraping script that downloaded and extracted isoform read counts for all tissue types from the HTML-pages rooted at (http://tools.genxpro.net/apadb/browse/). We extracted 400 nt long sequences centered around the cut site from the hg19 reference genome for each annotated APA event, obtaining the sequence regions containing the complete PASs. The sequences were aligned such that the most canonical CSE hexamer upstream of the cut site starts at position 50. We used the tissue-specific and pooled read counts to estimate relative isoform proportions for every pair of adjacent APA sites. The data was filtered to remove events with CSEs differing by more than 2 bases from the canonical hexamer AATAAA.

The data was further filtered by throwing out APA pairs with low total read count. The exact count threshold varied per tissue in order to strike a balance between sufficient quality and sufficient data size. For the pooled APA pairs, a cutoff of 1,000 reads was used, retaining ~2,400 pairs. For tissue types 'full blood' and 'HLF', a cutoff of 500 reads was used, retaining

~2,500 pairs. A read cutoff of 20 was used for the remaining tissues, and the number of retained pairs varied from ~700-5,000.

Leslie Data
The cell type specific APA dataset from (Lianoglou et al., 2013) was downloaded from (https://cbio.mskcc.org/leslielab/ApA/atlas/, 'unified-atlas', final version). The read alignments were mapped to the APADB site annotation in order to assign isoform counts for each APA site and cell type. The PAS sequence of each APA event was extracted from hg19 as previously described. The relative isoform proportion between neighboring APA sites was estimated from the mapped RNA-Seq read counts. We aggregated pooled estimates from all the cell type read counts.

Similar to APADB, the Lianoglou data was filtered by throwing out low-count pairs. For the pooled APA pairs, a cutoff of 1,000 reads was used, resulting in ~4,000 pairs. A cutoff of 50 reads was used for all individual cell types, and the number of retained pairs varied significantly (Figure S5D).

Architecture
Native genomic APA sites contain a larger degree of complexity and variation than what is captured by the base APARENT model. Specifically, in the UTR training libraries the sequence of the distal PAS is fixed and in relatively close proximity to the proximal PAS. In a native context, both distal site composition and distance varies greatly. However, we hypothesized that individual PASs are independent when signals are well-separated, and that site distance adds only a prior towards proximal site selection. If we further assume that the same sequence-specific regulation governs both proximal and distal sites, APARENT can be used as a "PAS scoring function", outputting a scalar score (the isoform proportion logit) that reflects the strength of a PAS for both the proximal and distal sequences. This reduces the native APA prediction task to learning a linear function that maps the predicted score of each PAS, and their log-distance, to the measured APA isoform proportions (architecture illustrated in Figure 5A).

We used the isoform APARENT model to predict the isoform log odds of both the proximal and distal sequences for every native APA pair. These two scalar-valued scores were used together with the natural log of the site distance in a linear regression model to predict proximal vs. distal isoform log odds. No regularization was needed because of the low dimensionality (3) of the feature space compared to the data size (~2,500 APA pairs).

Training
The extended model was trained on the pooled-tissue APADB events. All data was used for training (we used leave-one-out cross-validation in the evaluation, see further below). The model was trained by minimizing the Sum-of-squares error between predicted and observed isoform log odds.

Evaluation
The extended model was evaluated with leave-one-out cross-validation, where one APA pair was left out of the training data and all other events were used for training. This procedure was repeated for all of APA pairs. Each trained model was then tasked with predicting the isoform log odds of the held-out event (in Figure 5B). We noticed that the predictions agree better and better with increasing read depth (Figure S5A).

The trained pooled-tissue model predictions were compared against the tissue and cell type-specific isoform proportions of both APADB and the Leslie data with no additional re-fitting of the model (in Figure 5D-E).

A network identical to APARENT, and a simpler 6-mer regression model (as previously described), was trained exclusively on the APADB pooled-tissue dataset and used for comparison by testing each model's accuracy on the task of classifying preferential isoform on all APA pairs. For this test, the read count filter on the APADB dataset was increased to 1,500 (as it seemed to improve training of the baseline comparison models), resulting in a set of 1,040 pairs. 75% (780 pairs) were randomly chosen as training data and the remaining 25% (260 pairs) were used as test data. The ADADB-only network was trained until it was predicting the isoform log odds with zero error. Interestingly, this did not lead to overfitting (Figure S5B), as the test set error never started to increase again.

Human Variant Predictions

We synthesized human wildtype PAS sequences and annotated variants from ClinVar and HGMD to study the effects of APA misregulation and pathogenicity (in Figure 6 and 7). We also performed saturation mutagenesis of disease-implicated PASs. This section describes the computational method for predicting variant effects using APARENT. See STAR Methods subsection "Human Wildtype, Variant & Engineered Sequence Array" for a detailed description of the construction, synthesis and experimental measurement of the variant data.

3' UTRs and Variants

We collected and synthesized a total of 1,085 wildtype human PAS sequences, and a total of 12,348 variants corresponding to SNVs or InDels within those native PASs. 1,811 of the SNVs and Indels are variants that occur in ClinVar or HGMD either 50 nt upstream or 100 nt downstream of an annotated PAS in APADB. The remaining set of variants were the result of either full saturation mutagenesis of ~20 disease-implicated PASs or random (not full) saturation mutagenesis of native human (not disease-implicated) PASs. The saturation mutagenesis was performed in the region -25 to +75 nt up- and downstream of the CSE hexamer.

The full saturation mutagenesis was performed on the disease-implicated PASs of the following genes: HBB, HBA2, TP53, INS, ARSA, FOXC1, FOXP3, F2, BMP2. We also performed full saturation mutagenesis of PASs of ACMG genes BRCA1, BRCA2, PTEN and TPMT.

Variant Effect Estimation & Significance Testing

Given a variant PAS sequence and its corresponding wildtype PAS sequence, we calculated the measured variant effect size as the log fold change, or log odds ratio, between variant and wildtype isoform proportions p_{var} and p_{ref}:

$$\Delta_{true} = logit(p_{var}) - logit(p_{ref})$$

In some of the analyses carried out in Figure 6 and 7, we filtered the variant set on only those with a statistically significant isoform fold change (for example to evaluate prediction accuracy on high-impact variants). Under the hypothesis that a variant isoform proportion is different from its wildtype proportion, we calculated the empirical p-value using a two-sided proportion difference test:

$$H_0: p_{var} - p_{ref} = 0$$

$$\hat{p} = (prox_{var} + prox_{ref} \; / \; n_{var} + n_{ref})$$
$$z = (p_{var} - p_{ref}) \; / \; \sqrt{\hat{p}(1-\hat{p})(1/n_{var} + 1/n_{ref})}$$
$$p - value = 2 \; * \; P(Z \; > \; z)$$

Where p_{var} and p_{ref} are the measured proximal proportions of the variant and wildtype PAS
$prox_{var}$ and $prox_{ref}$ are the measured proximal counts.
n_{var} and n_{ref} are the measured proximal and distal counts.
P(Z > z) is the survival function of the Normal distribution.

In the analyses of Figure 6 and 7, we used a p-value of 0.0001 to filter variants. This value was chosen as it gives a sufficiently low expected false positive rate (12,348 variants x 0.0001 ~= 10 false positives). The exception is in Figure 6C (and Figure S6C), where we used a more stringent p-value of 0.00001 as we wanted close to 0 false positives in the search for complex non-linear variants.

Prediction Method
We used the isoform APARENT model to predict the effects of variants, by separately passing the wildtype and variant PAS sequences as input and predicting the wildtype and variant isoform proportions y_{var} and y_{ref}. The predicted log fold change was calculated as:

$$\Delta_{pred} = logit(y_{var}) \; - \; logit(y_{ref})$$

Δ_{pred} could then be directly compared to the measured log fold change Δ_{true} for each variant.

In Figure 6C and S6C we use both the APARENT model and a 6-mer linear regression model (as described in STAR methods subsection “Linear Logistic 6-mer Regression Baseline”) to search for variants whose predicted direction of fold change (the sign of Δ_{pred}) is significantly different between the two models. Specifically, we filtered variants on the condition that the sign of Δ_{pred} is different between model predictions, and that $|\Delta_{pred}| > 0.1$for both models. The idea behind this is that the linear model can only account for regulatory variant effects by independent exchanges of 6-mers (between the variant and wildtype sequence), while APARENT has the ability to capture any type of non-linear effect between components anywhere within the PAS. Thus, if their predictions significantly differ, and furthermore if APARENT calls the direction of the variant correctly compared to measurements, it is indicative of the presence of non-linear regulatory effects.

GEUVADIS Variant Prediction

We validated APARENT’s ability to predict the effect of genomic APA variants on the GEUVADIS dataset (in Figure S6B), which consists of RNA-Seq data for a subset of individuals from the 1000 Genomes project.

Data & Processing
The RNA-Seq .bam-files from the GEUVADIS accession E-GEUV-1, the genotype variant call files of the 1000 Genomes project (vol 1, phase 1, v3, accession 20101123), and the Human Tandem 3’ UTR .gff annotation from the MISO project were downloaded. The Tandem 3’ UTR annotation was used to identify coordinate regions within 50 bp of annotated polyadenylation sites (cut sites) in the genome. The 1000 Genomes .vcf-files were filtered for variants occurring

within these regions. The reference and variant sequences of the filtered set of variants were extracted from the hg19 reference genome. Finally, the RNA-Seq data from GEUVADIS were processed using MISO, which estimates the reference and variant isoform ratios per human sample. The set of variants were further filtered such that the isoform ratios had a narrow confidence interval according to MISO (at most 25% difference between upper- and lower confidence bounds). The number of human reference samples of each variant that passed the confidence filter had to be greater than 10.

Finally, we averaged APA isoform abundance over individuals with a given variant and over individuals not carrying the variant, while keeping heterozygous and homozygous samples separate, obtaining a mean log odds ratio (fold change) estimate per variant.

Prediction Method

Since the entire UTR sequence is identical between wild type and variant samples except for the SNV, the shift in isoform log odds predicted by APARENT (which only receives the proximal PAS sequence as input) should be directly comparable to the observed shift in isoform log odds estimated by MISO from the RNA-Seq data. Hence, we let APARENT predict the isoform log odds of the variant and reference sequences and subtracted these two quantities to obtain a log odds ratio prediction per SNV. When predicting, we set the training library intercept terms of APARENT's regression layer to 0, however they do not matter since the effect of intercept terms are cancelled when subtracting reference from variant log odds predictions:

$$\text{logit}(\text{sigmoid}(\text{net}(S_{var} + I_1 + \ldots + I_K))) - \text{logit}(\text{sigmoid}(\text{net}(S_{ref} + I_1 + \ldots + I_K)))$$
$$= \text{logit}(\text{sigmoid}(\text{net}(S_{var}))) - \text{logit}(\text{sigmoid}(\text{net}(S_{ref})))$$

where S_{var} and S_{ref} are the variant and reference 1-Hot-coded input sequences, net(...) is the neural network model up to the final dense layer, $I_1 + \ldots + I_K$ are the library intercept terms, and sigmoid(...) is the final output prediction of APARENT.

QUANTIFICATION AND STATISTICAL ANALYSIS

Throughout the paper, R^2 is reported as the standard evaluation metric. In Figure 2B and Figure 3B, R^2 is defined as 1 - SSE/SST, where SSE is the sum-of-squares error and SST is the data variance. In the remainder of analyses, R^2 is defined as the square of Pearson's r. See STAR methods subsection "APARENT Model (Isoform Prediction)" for a detailed description of why we calculate R^2 according to two definitions.

In the variant prediction sections (Figure 6 and 7), a two-sided proportion difference test is used to filter variants that are significantly different from the wildtype isoform proportions. See STAR methods subsection "Human Variant Predictions" for a detailed description of this significance test.

Throughout the paper, whenever we evaluate prediction accuracy or correlation against measurements, we always present results on data we have not trained the machine learning model (APARENT) on. In Figure 2, we present evaluation results on test sets of trained-on UTR libraries (Figure 2B) or on completely held-out UTR libraries (Figure 2C-D). In Figure 3, we present evaluation results on test sets of trained-on libraries (Figure 2B-C). In Figure 5, we present evaluation results using a leave-one-out cross-validation test (Figure 5B) or on held-out test sets (Figure 5C-F). In Figure 6 and 7 (the human variant analysis), we did not re-train or

fine-tune the APARENT models on any of the variant data, but used the APARENT model trained only on random UTR libraries to infer variant effects.

DATA AND SOFTWARE AVAILABILITY

The raw sequencing data for the random 3' UTR plasmid libraries and the designed array library (human variants and engineered pA sequences) are available at GEO accession GSE113849. All processed datasets used for training, evaluation and analysis, including the source code, are available on GitHub (https://github.com/johli/aparent).