LURE PSB Supplemental Methods

David Haan, Ruikang Tao, Verena Friedl, Ioannis Nikolaos Anastopoulos, Christopher K Wong,
Alana S Weinstein, Joshua M Stuart[†]

Dept. of Biomolecular Engineering and UC Santa Cruz Genomics Institute
University Of California, Santa Cruz, Santa Cruz, CA 95064, USA
[†]*E-mail: jstuart@ucsc.edu*

1. Supplemental Methods

1.1. *Method Comparison*

There are several existing methods that predict driver from passenger mutations.[1] EPoC uses network modeling of the transcriptional effects of copy number aberrations to identify driver mutations in glioblastoma multiforme (GBM).[2] DriverNet employs a probabilistic model to locate driver mutations using transcriptional networks.[3] PARADIGM-Shift uses an integrative pathway inference method to predict a mutation's impact based on the discrepancy between a gene's upstream input and its downstream output.[4] These methods can predict novel drivers given a set of SNVs or copy number alterations and the corresponding mRNA gene expression data. In addition, there are methods that identify modules of driver genes based on mutual exclusivity in certain tumor types, such as CoMEt[5] and MEMo, the latter of which incorporates prior knowledge such as pathway data into driver gene module discovery.[6]

Similar to LURE, REVEALER is a computational method that identifies combinations of genomic alterations correlated with functional phenotypes, such as the activation or gene dependency of oncogenic pathways or sensitivity to a drug treatment.[7] While the concept of REVEALER is similar to LURE, LURE has several advantages. First, at every iteration LURE produces a new classification model slightly more accurate than the previous because the newly discovered events are no longer considered false positives and now aide in determining features relevant to the signature. Second, REVEALER requires a mutually exclusive relationship between new events which may limit results as mutation calls are not 100% accurate. By allowing some overlap between predicted events, LURE can account for possible mutation call errors and identify modules containing co-mutated events.

Another similar method, Onco-GPS, deconvolutes gene expression signatures to delineate oncogenic cellular states using gene expression, mutation, and pathway data.[8] Using these oncogenic cellular states, the authors were able to generate maps of divergent cell states and correlate these maps with drug sensitivity.. LURE, on the other hand, does not consider the specific features of the individual expression signatures but rather uses the signature as a whole to relate similar mutation events or oncogenes. This provides an advantage in situations where the relationship between the gene expression and the oncogenic effect are unknown.

Rykunov et al. use a novel machine learning method to identify expression signatures of altered oncogenic pathways in tumor samples and discover new driver mutations among samples harboring similar pathway signatures.[9] LURE works in a similar fashion by using a machine learning model to identify new driver mutations using signatures of known drivers, however, LURE does not restrict to known pathway gene sets and thus may detect a wider variety of mutations. In addition, LURE uses the area under the precision-recall curve (PR AUC) to estimate the hyperparameters of the classification models while Rykunov et al. use the ROC metric. By using PR AUC, LURE allows for an imbalance in class ratios which enables discovery of rare driver events.

1.2. *Classification Model Comparison*

We tested three different types of binary classification models: random forest, neural network, and logistic regression. The intersection of the top 100 baits of each method are shown in Supp. Figure 3.

We trained the random forest classification model using the "ranger" R package[10] and used 10-fold cross validation to obtain the test Precision-Recall Area under the Curve (PR AUC). The tuneRF function in the "randomForest" R package[11] was used to determine the number of number of candidate variables at each split, i.e. the value of the "mtry" parameter, that resulted in the lowest out-of-bag (OOB) error. The sample size, "sample.fraction," was calculated according to class proportions in order to increase the diversity and the correlation of trees for extremely unbalanced cases. Weight vectors were also calculated according to class proportions to give higher weight to the minority class.

We used the Keras[12] API to train our 10-fold cross-validated neural network classifier. The ReLU activation function was used for the input layer and the sigmoid activation function was used for the output layer. We used the Adam optimization algorithm[13] and the categorical cross-entropy loss function as implemented in Keras. The models were trained for 400 epochs in 32 batches.

LURE's logistic regression classification model was built using R's glmnet package.[14] Up to 10-fold cross validation can be performed for each iteration of LURE, but this is a customizable parameter. Each fold is stratified, meaning there is at least one positive (or negative) member in every fold. The number of folds is dependent on the number of mutations. For example, if there are only 5 mutations, there can only be a maximum of 5 folds. In addition to stratifying the folds to account for imbalanced datasets, LURE adds extra weight penalties per class according to class proportions. To reduce the numbers of features and speed up the calculation, LURE uses LASSO regularization with $\alpha = 1$. To measure the accuracy of each model and to correctly choose the regularization parameter λ, LURE uses PR AUC, which has been shown to be more informative than ROC on imbalanced datasets.[15] Within each fold, the λ value which maximizes the PR AUC is chosen. The λ values from all folds are averaged to produce the final λ for that classification model.

1.3. *SSEA (Sample Set Enrichment Analysis)*

We developed Sample Set Enrichment Analysis (SSEA) based on the GSEA Preranked Tool v2.2.2.[16] For our SSEA implementation, the GSEA Preranked tool takes two files, the first being a gmt file which contains one mutation per line and the samples which harbor this mutation. The second file contains a continuous value between 0 and 1 for each sample, which is the sample score from the mutation classification model. SSEA uses the default GSEA PreRanked parameters found in the command-line implementation. LURE considers events significant when they have a GSEA p-value greater than 0.05, FDR value less than 0.25, and are mutated in at least 4 samples, however these are customizable parameters.

1.4. *Positive Controls*

The IDH1 Positive controls were created using the IDH1 mutated samples in the Lower Grade Glioma (LGG) TCGA dataset. Of the 210 LGG samples with an IDH1 missense mutation, we created an initial bait with 150 samples and three sets of 20 samples as potential catch events. The SF3B1 positive controls were created in a similar manner using the SF3B1 missense mutated samples in the Uveal Melanoma (UVM) TCGA dataset. We created an initial bait using 8 SF3B1-mutated samples and left out two sets of 5 SF3B1-mutated samples for discovery. The gmt files for each of these positive controls are provided as Supplemental Data: 'positive_control_SF3B1_missense.gmt'; 'positive_control_IDH1_missense.gmt'. The gene expression data for the positive controls and the other LURE analyses are not provided here, but can be downloaded from the UCSC Xena datahub[17] at 'https://xena.ucsc.edu/'.

1.5. *Parameters for the TCGA Pan-Cancer Dataset Analysis*

The 723 COSMIC cancer consensus genes were downloaded from 'https://cancer.sanger.ac.uk/census' on June 9, 2019.[18] This list is provided as Supplemental Data: 'COSMIC_Census_allSun_Jun_9_07_12_56_2019.tsv'. The cross validation test scores of the baits which passed our restrictions can found in 'TCGA_Classification_Model_Scores.txt'. LURE was run with its default parameters except for limiting the maximum tree length to 3 and number of events at each tree leaf to 3. These restrictions were used to prevent long compute time for highly mutated tumor types. These parameters and the other default parameters are further described in LURE's Github repository.

1.6. *Parameters for the ALT Analysis*

The bait events for the ALT analysis were ATRX truncating mutations in the TCGA Sarcoma (SARC) and LGG data sets. The catch gene list used for the ALT driver analysis was downloaded from the TelNet database in May 2018.[19] The list used in this specific analysis can be found in 'TelNet_Genes.txt'. The default LURE parameters were used with no restriction on the tree length or number of mutations.

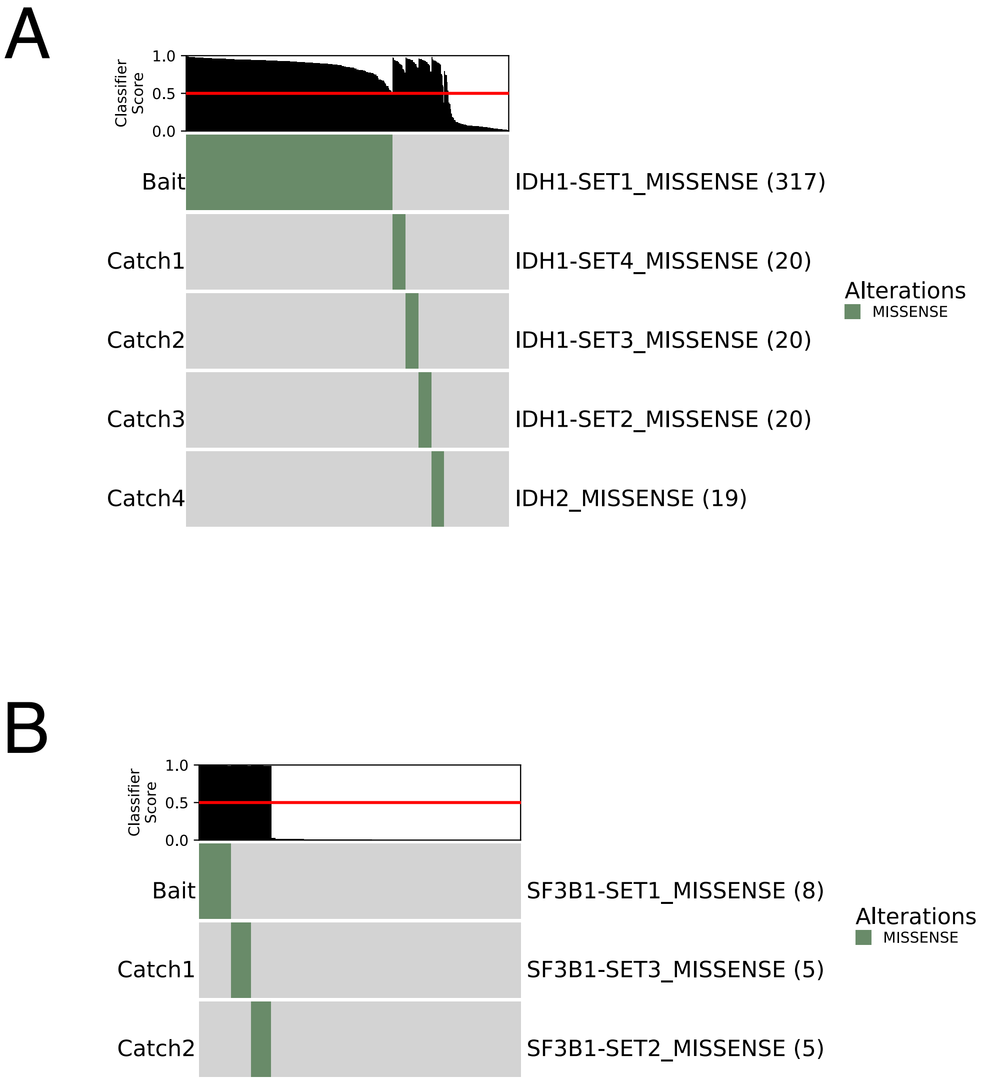
1.7. *Parameters for the MAPK/RTK Analysis*

The possible bait gene list for the MAPK/RTK analysis was restricted to a curated list of genes associated with the MAPK/RTK pathway as described in Sanchez-Vega.[20] This set of genes can be found in 'MAPK_RTK_Pathway_genes.txt'. The cross validation test scores of the bait classification models which had a PR AUC $>$ 0.5 can be found in 'TCGA_Classification_Model_Scores.txt'. All genes and mutations were considered as catch events in this analysis. The LURE 'percent_overlap' parameter's default value is set at 0.5 to allow for overlap in results, but for this analysis we set the parameter to only 0.1 in order to find nearly mutually exclusive events.

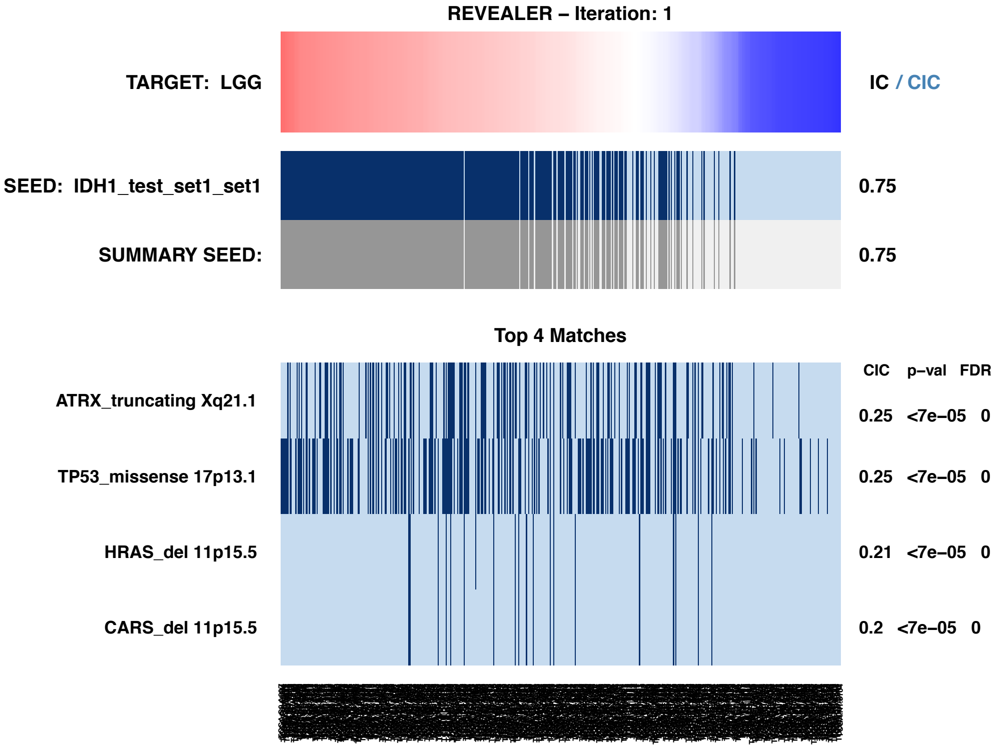
1.8. *LURE Parameters*

These parameters are used by both the LURE R function and the LURE wrapper script. The defaults shown here are set to run the SF3B1 UVM positive control.

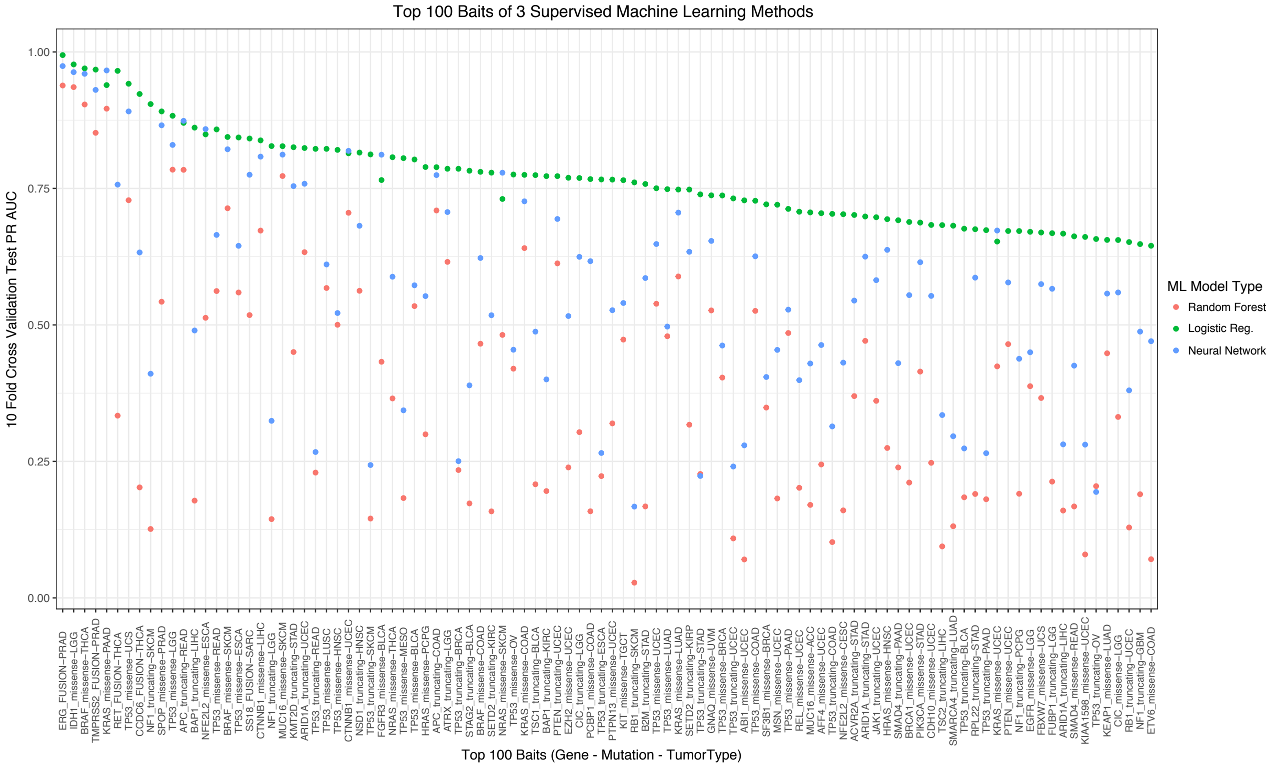
Parameter	Default Value	Description
–folds	10	Number of Cross Validation Folds.
–num_permutations	5	Number of iterations performed for each model. The more iterations, the more accurate the model
–min_gene_set_size	4	Catch event minimum size: parameter for SSEA; only events with min_gene_set_size or more mutated samples are considered.
–percent_overlap	0.5	If percent_overlap of the samples harboring the potential catch event are in the existing bait sample set then we skip it. A smaller number is more restrictive.
–max_tree_length	5	Sets the max length of the Event Discovery Tree (EDT). A longer EDT will result in longer run times.
–bait_gene	“SF3B1-SET1_MISSENSE”	Bait gene name. Must be present in the provided gmt file. Multiple baits are allowed separated by a semicolon.
–gmt_file	“positive_control_SF3B1_missense.gmt”	Gene Matrix Transposed (gmt) formatted file. Each line in the file lists a mutation and the samples harboring the mutation. See positive control files for examples. Must be located in input directory.
–gsea_fdr_threshold	.25	FDR value threshold for GSEA step.
–gsea_pvalue_threshold	.05	P value threshold for GSEA step.
–LURE_pvalue_threshold	.05	P value threshold for LURE PR AUC score step.
–max_num_events	5	Used to limit the number of catch events found by GSEA and considered for LURE’s classifier AUC score step. The events are sorted by GSEA NES score so the top events will be chosen. The larger this parameter the longer the runtime.
–feature_data_file	“pancan_RNAexp_UVM”	Feature Data File. File must be located in input directory.
–target_gmt_file	“”	This argument only pertains when LURE is run with enrichment only. It is the gmt file for the test/target dataset. The original gmt_file argument is used to identify the bait event samples.
–target_feature_file	“”	This argument only pertains when LURE is run with enrichment only. It is the feature file for the test/target dataset. File must be located in the input directory.
–output_file_prefix	“V10”	This is the file prefix assigned to all the output files. For multiple runs it helps keep track of each run.
–tissue	“”	Tissue or Tumor Type, used for additional filename prefix for large pancan runs.



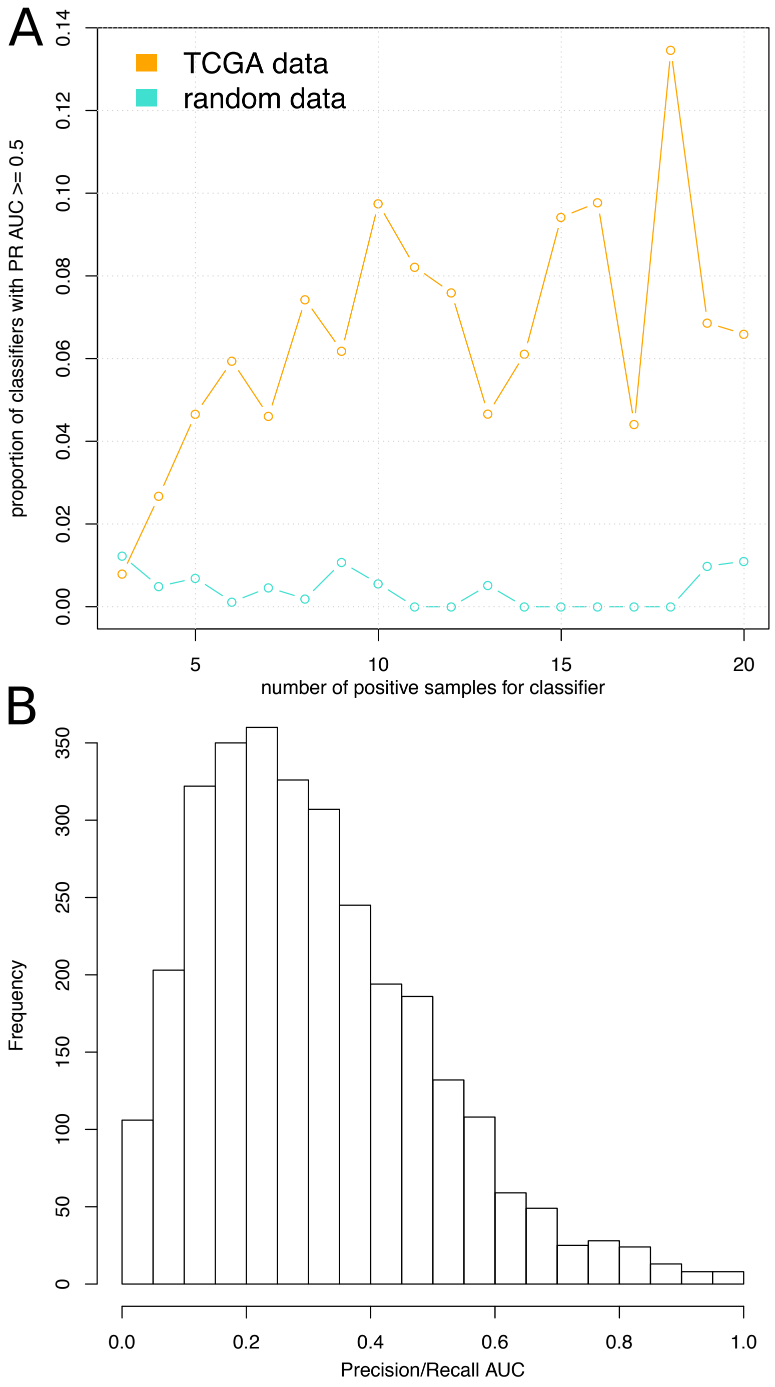
Supp. Fig. 1. **LURE Positive Controls. (A) IDH1 Positive Control Graph.** Graph showing IDH1 positive control results in Lower Grade Glioma (LGG). Approximately 500 LGG samples are represented by gray tick marks. The mutant samples are shown in green. The initial LURE bait was set to 150 of those samples, with 3 sets of 20 held out and left for discovery. LURE found all 3 held-out sets and in addition finds mutant IDH2 missense events, a known association. **(B) SF3B1 Positive Control Graph.** Graph showing SF3B1 positive control results in Uveal Melanoma (UVM). 80 UVM samples are represented by grey tick marks. The 18 SF3B1 mutant samples are shown in green. The initial LURE bait was set to 8 of those samples, with two sets of 5 held out and left for discovery. LURE found the two held-out sets, successfully identifying all SF3B1 mutants.



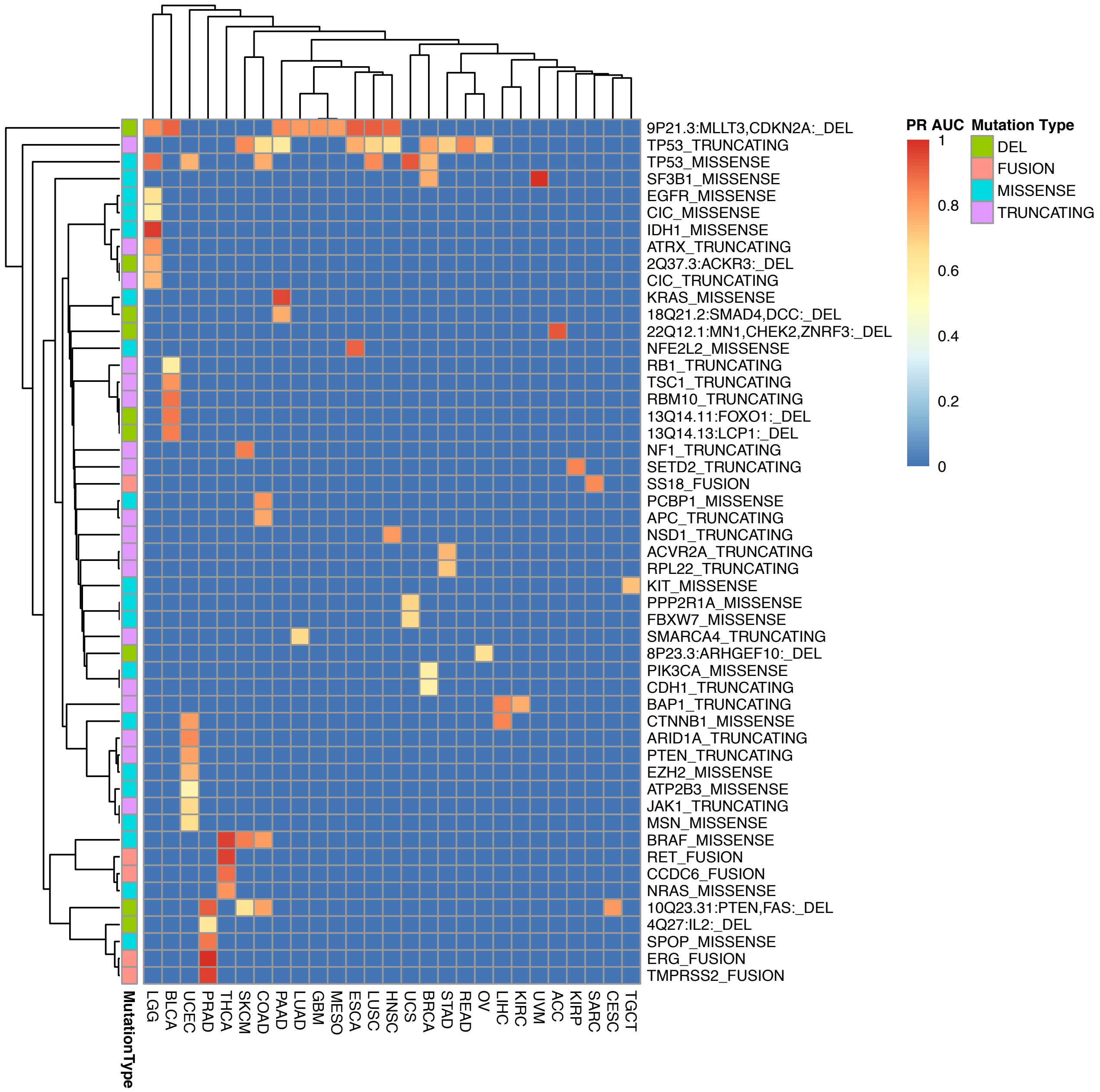
Supp. Fig. 2. **REVEALER Results on IDH1 Positive Control.** We ran the REVEALER method using our IDH1 positive controls. We first trained a logistic regression model on IDH1 'SET1' mutants (see Supp. Figure 1) in LGG and ran the model back on our training data to obtain scores for each sample. We used the response variable as a continuous input variable for REVEALER and set the seed to the IDH1 Positive Control SET1 and it was unable to find the other 3 sets.



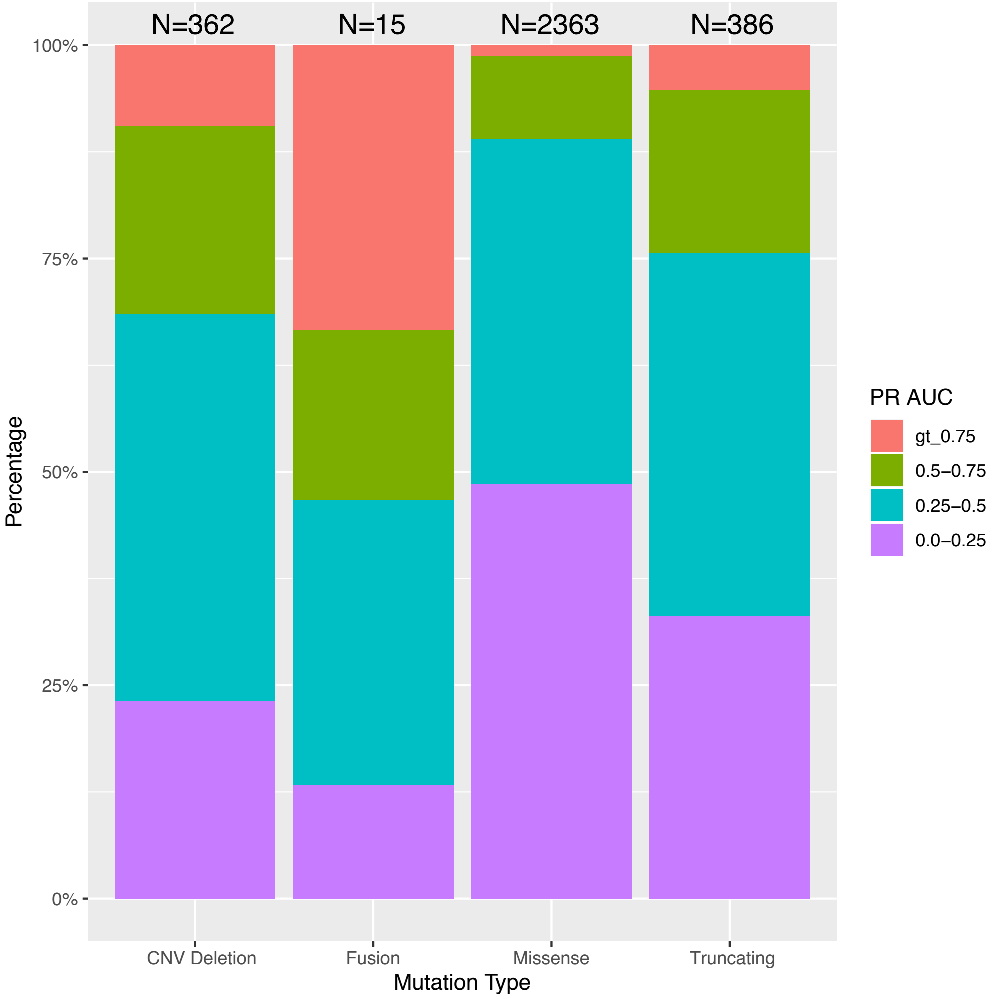
Supp. Fig. 3. **LURE Bait Classification Model Accuracies.** Plot shows the F1 score for three different supervised machine learning techniques. Each model was trained on tissue-specific mutation alterations as denoted on the x-axis. The scores from the random forest model were obtained using out-of-bag (OOB) estimate. The linear model scores were from 10-fold cross validation of a binary classification model using logistic regression. The neural network was built with one hidden layer with 1,000 nodes and also cross validated to obtain an F1 score.



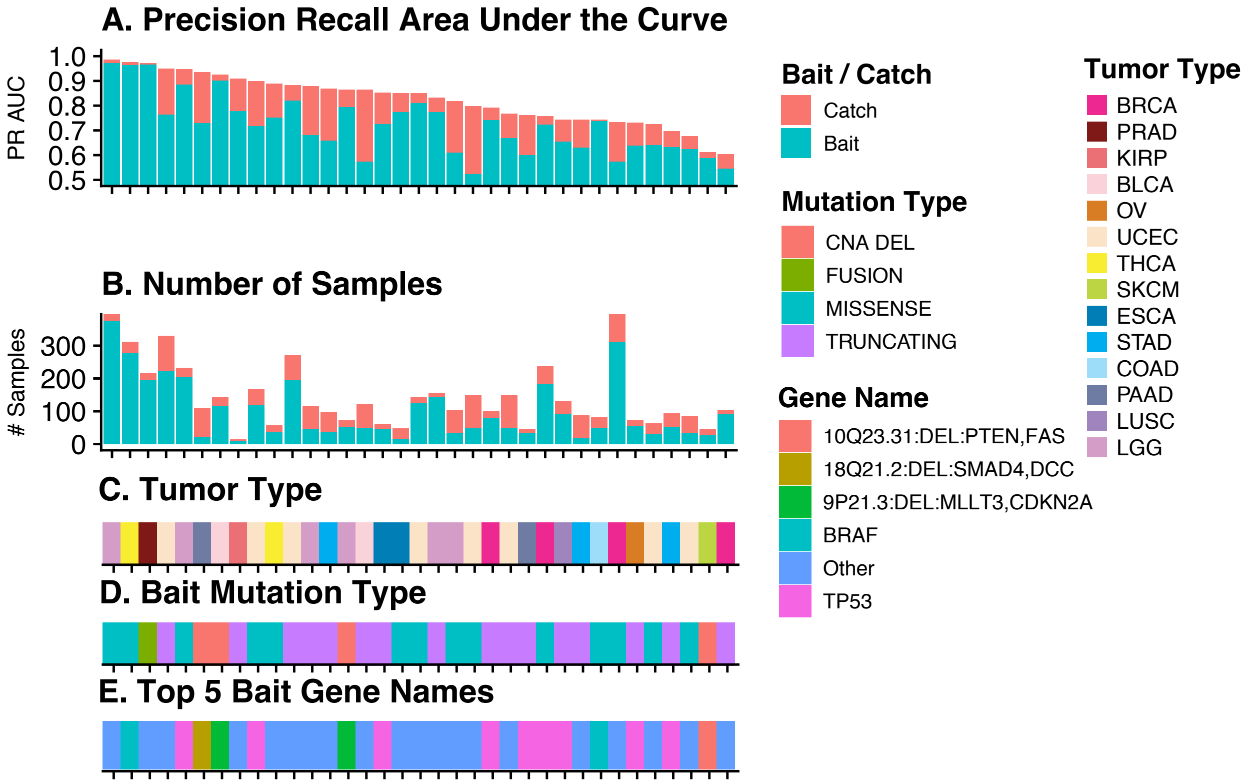
Supp. Fig. 4. **LURE TCGA Bait Classifiers (A)** Proportion of bait classifiers with Precision-Recall Area Under the Curve (PR AUC) $\geq$ 0.5 at increasing number of positive samples that have the bait event. The random data simulates the same amount of classifiers with the same number of positive samples as the TCGA bait events, but with random positive samples. **(B) LURE TCGA Potential Bait Histogram.** Histogram shows the PR AUC for 3,053 bait event/tumor type combinations with greater than 10 positive samples. PR AUC was averaged across 10 folds using cross validation.



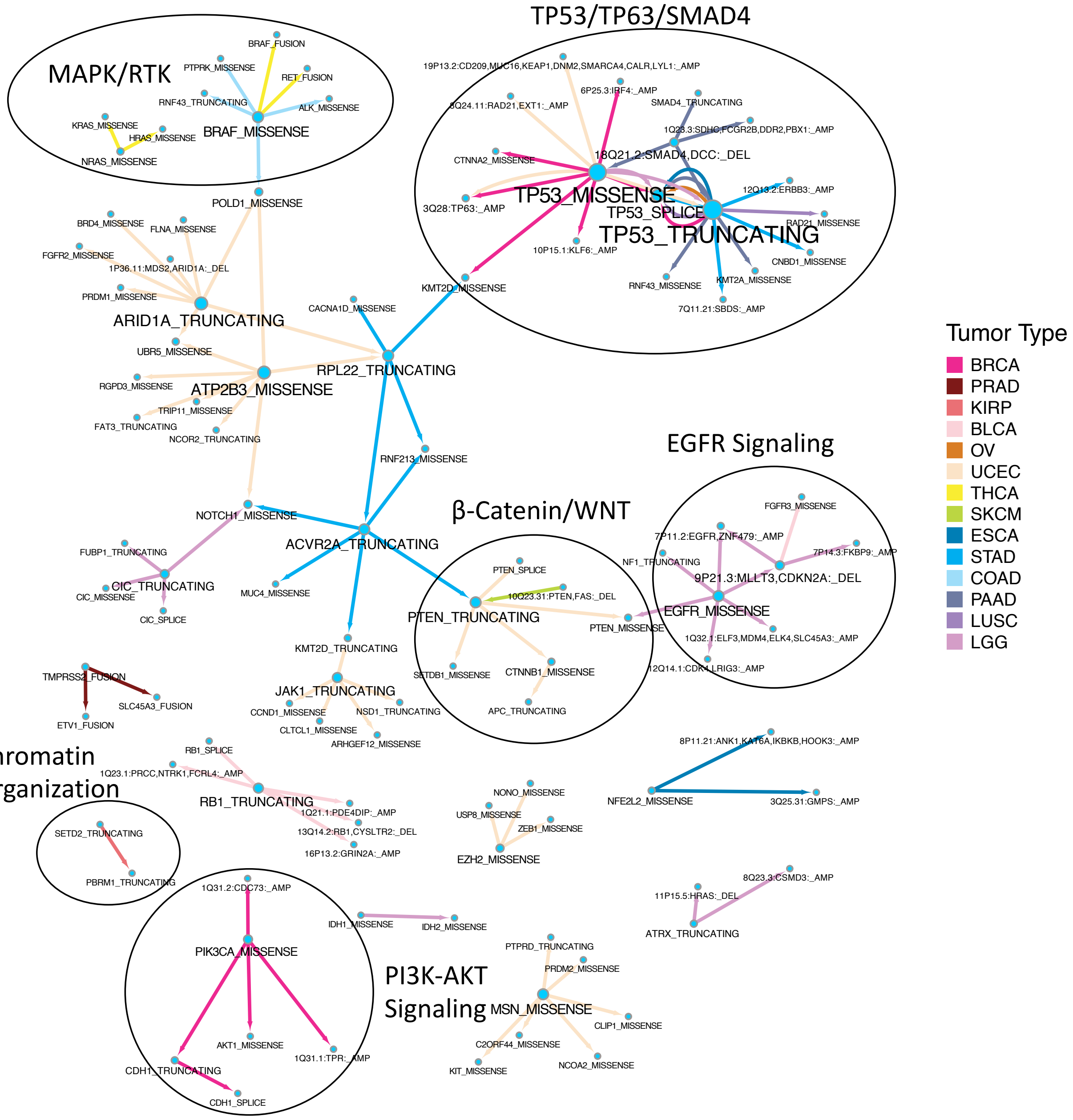
Supp. Fig. 5. **LURE TCGA Bait PR AUC Heatmap.** Heatmap shows the baits after filtering for only high-scoring potential baits (PR AUC> 0.4, precision> 0.3, recall> 0.75). The tumor type of each bait is shown as columns and each event as a row. The mutation type of each event is represented by color in the annotation bar on the left. The PR AUC of each bait is represented by color intensity in the heatmap.



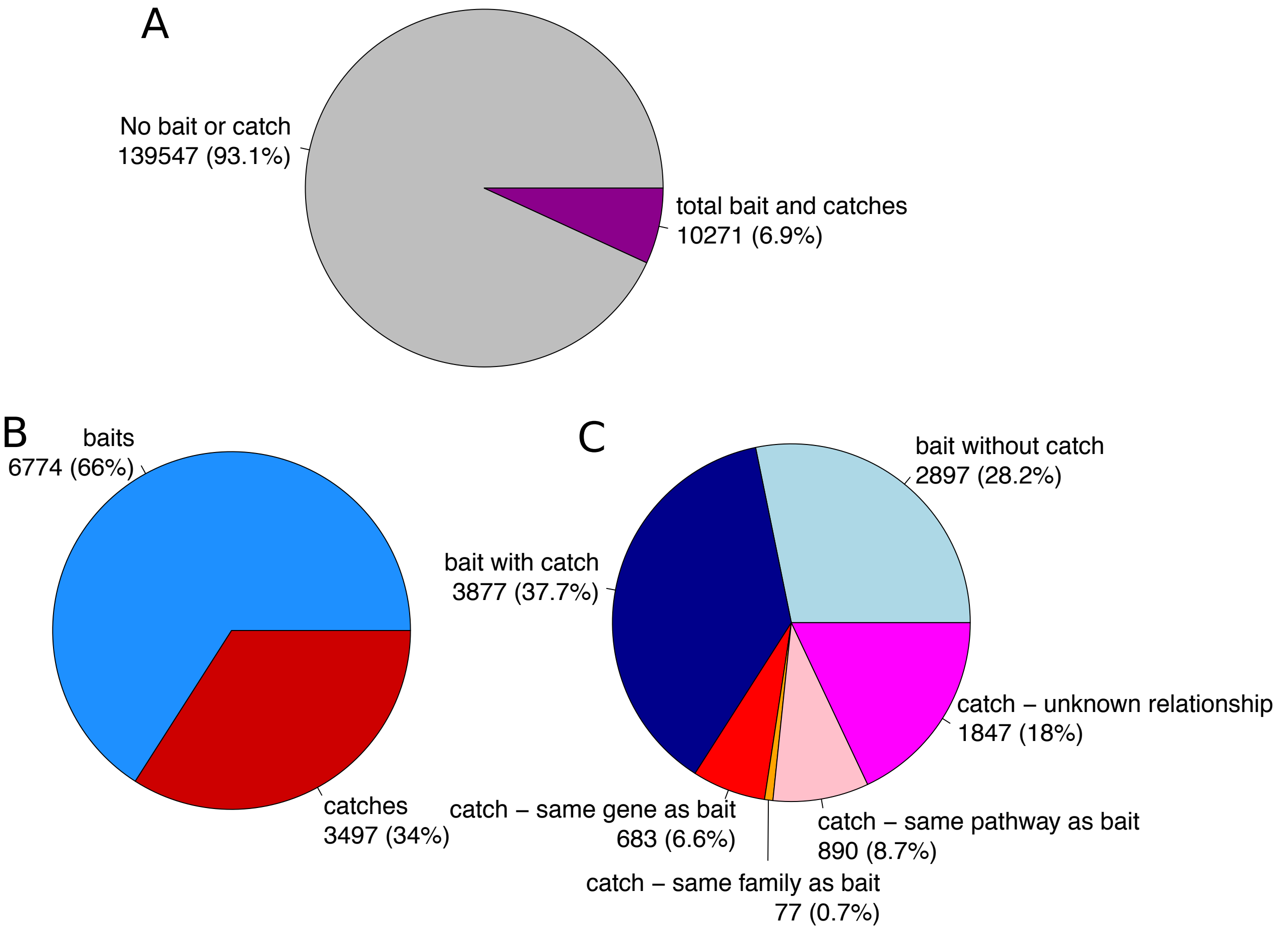
Supp. Fig. 6. **LURE TCGA Potential Bait Mutation Types.** Barplot shows the LURE bait classifiers for each mutation type (homozygous focal point copy number deletions, gene fusions, missense mutations, truncating mutations) binned by their accuracy measured in Precision Recall Area Under the Curve (PR AUC). The two top bins of accurate bait classifiers (PR AUC ≥ 0.75 and $0.5 \leq$ PR AUC < 0.75) contain baits promising for further consideration in LURE.



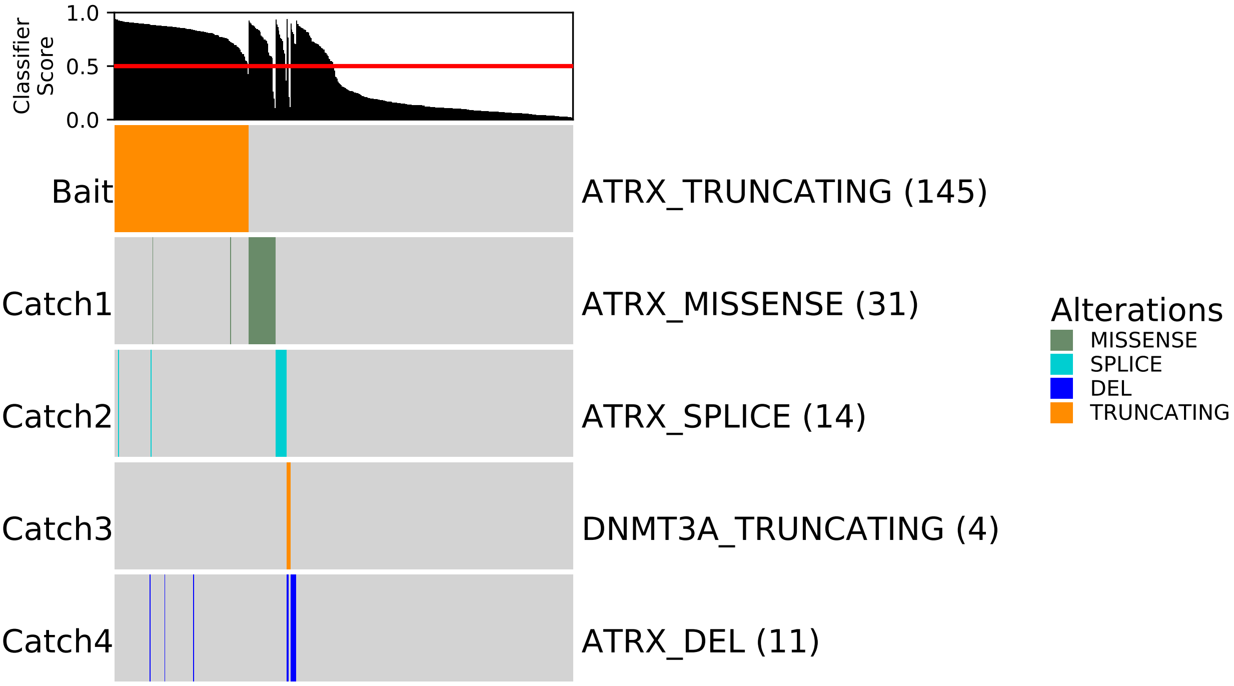
Supp. Fig. 7. **35 TCGA Pan-Cancer Atlas Bait-Catch Associations.** (A) Precision Recall Area Under the Curve Barplot. Barplot shows the bait (blue) and catch (red) PR AUC for each bait which found a catch. **(B) Number of Samples Barplot.** Barplot shows the number of samples in the bait and catch. **(C) Tumor Type Annotation Bar.** Each color of the annotation bar represents the tumor type of the bait and catch in which the association was found. **(D) Bait Mutation Type Annotation Bar.** Colors show the mutation type of the bait. **(E) Bait Gene Name Annotation Bar.** Colors show the top 5 most recurrent bait genes. ‘Other’ (blue) represents the other 30 bait mutation genes.



Supp. Fig. 8. **TCGA Event Net.** LURE Event Net shows all associations found in the TCGA dataset. Nodes represent events and each directed edge is a bait-catch association. The direction of the edge represents the bait-catch relationship (arrows pointing from bait to catch). The color of each edge is the tumor type in which the association was found. Pathway findings are circled and annotated by pathway name.



Supp. Fig. 9. **LURE genomic events in TCGA Pan-Cancer Samples.** Pie charts showing the mutational events being present in TCGA samples and how they were used in LURE. **(A)** All genomic events used in LURE: missense mutations, truncating mutations, homozygous focal point copy number deletions, and gene fusions in 732 COSMIC genes for bait events, and additionally splice site mutations and focal point copy number amplifications in catch events. The pie chart is divided into events that were neither bait or catch in LURE (no transcriptional signal detected), and the events that were either bait or catch or both (transcriptional signal detected). **(B)** Pie chart including all events from 'total bait and catches' in (A), divided into events that were used as bait by LURE and events that were used as catch and never as bait. **(C)** Same as (B), but baits are further divided by if LURE was successful in associating the bait event with at least one catch (bait with catch), or not (bait without catch). Catches are further divided by their relationship with the bait they were associated with by LURE: bait and catch are from the same gene, bait and catch genes are in the same gene family, bait and catch genes are in the same biological pathway (excluding pathway gene sets with > 1000 genes),[21] or the bait and catch genes did not have any of those connections (unknown relationship).



Supp. Fig. 10. **LURE Graph of ALT LGG Results.** LURE graph showing results using ATRX truncating mutations in LGG as bait. Four Catch Events are identified: ATRX missense and splice mutations, DNMT3A truncating mutations, and ATRX copy number deletions.

References

1. A. Gonzalez-Perez et al., Computational approaches to identify functional genetic variants in cancer genomes, *Nature Methods* **10**, p. 723 (2013).
2. T. Abenius, R. Jörnsten, T. Kling, L. Schmidt, J. Sánchez, and S. Nelander, System-scale network modeling of cancer using epoc, *Advances in Systems Biology* , p. 617 (Springer, 2012).
3. A. Bashashati et al., DriverNet: uncovering the impact of somatic driver mutations on transcriptional networks in cancer, *Genome Biology* **13**, p. R124 (2012).
4. S. Ng et al., Paradigm-shift predicts the function of mutations in multiple cancers using pathway impact analysis, *Bioinformatics* **28**, p. i640 (2012).
5. M. D. Leiserson, H.-T. Wu, F. Vandin, and B. J. Raphael, CoMEt: a statistical approach to identify combinations of mutually exclusive alterations in cancer, *Genome Biology* **16**, p. 160 (2015).
6. G. Ciriello, E. Cerami, B. A. Aksoy, C. Sander, and N. Schultz, Using MEMo to Discover Mutual Exclusivity Modules in Cancer, *Current Protocols in Cioinformatics* **41**, p. 8 (2013).
7. J. W. Kim et al., Characterizing genomic alterations in cancer by complementary functional associations, *Nature Biotechnology* **34**, p. 539 (2016).
8. J. W. Kim et al., Decomposing Oncogenic Transcriptional Signatures to Generate Maps of Divergent Cellular States, *Cell Systems* **5**, 105 (2017).
9. D. Rykunov, N. D. Beckmann, H. Li, A. Uzilov, E. E. Schadt, and B. Reva, A new molecular signature method for prediction of driver cancer pathways from transcriptional data, *Nucleic Acids Research* **44**, p. e110 (June 2016).
10. M. N. Wright and A. Ziegler, ranger: A Fast Implementation of Random Forests for High Dimensional Data in C++ and R, *Journal of Statistical Software* **77**, p. 1 (March 2017).
11. A. Liaw and M. Wiener, Classification and regression by randomForest, *R News* **2**, p. 18 (2002).
12. F. Chollet *et al.*, Keras `https://github.com/fchollet/keras`, (2015), original-date: 2015-03-28T00:35:42Z.
13. D. P. Kingma and J. Ba, Adam: A Method for Stochastic Optimization, *arXiv:1412.6980 [cs]* (December 2014).
14. J. H. Friedman, T. Hastie, and R. Tibshirani, Regularization Paths for Generalized Linear Models via Coordinate Descent, *Journal of Statistical Software* **33**, p. 1 (February 2010).
15. T. Saito and M. Rehmsmeier, The Precision-Recall Plot Is More Informative than the ROC Plot When Evaluating Binary Classifiers on Imbalanced Datasets, *PLOS One* **10**, p. e0118432 (2015).
16. A. Subramanian et al., Gene set enrichment analysis: A knowledge-based approach for interpreting genome-wide expression profiles, *Proceedings of the National Academy of Sciences* **102**, p. 15545 (2005).
17. M. Goldman et al., The UCSC Xena platform for public and private cancer genomics data visualization and interpretation, *bioRxiv 326470* (January 2019).
18. J. G. Tate et al., COSMIC: the Catalogue Of Somatic Mutations In Cancer, *Nucleic Acids Research* **47**, p. D941 (2018).
19. D. M. Braun, I. Chung, N. Kepper, K. I. Deeg, and K. Rippe, TelNet - a database for human and yeast genes involved in telomere maintenance, *BMC Genetics* **19**, p. 32 (2018).
20. F. Sanchez-Vega et al., Oncogenic Signaling Pathways in The Cancer Genome Atlas, *Cell* **173**, p. 321 (2018).
21. D. Merico, R. Isserlin, O. Stueker, A. Emili, and G. D. Bader, Enrichment map: a network-based method for gene-set enrichment visualization and interpretation, *PLOS One* **5**, p. e13984 (2010).