*All sub-sections below are associated with R scripts hosted on* [*GitHub*](https://github.com/rreggiar/KRAS-AALE-Paper/blob/master/KRAS-AALE-CODE/) *and organized in the “aale.2019.analysis.notebook.Rmd”*

RNA-seq psuedoalignment and quantification

All *fastq* files were trimmed with *Trimmomatic 2 (0.38)* [] using the Illumina NextSeq PE adapters. The resulting trimmed files were assessed with *FastQC* [] and then passed through the following analytical pipeline:

Salmon (0.14.1): pseudoalignment of RNA-seq reads performed with *Salmon* [] using the following arguments:

--validateMappings --rangeFactorizationBins 4 --gcBias --numBootstraps 10

with the ouput directories being processed in the following manners.

Sleuth: transcript differential expression was performed using *Sleuth* []. Upon completion, the transcripts with q-values below 0.05 in the likelihood-ratio test were used to filter salmon output from which log2fc was manually calculated and paired to the sleuth output.

DESeq2: *Salmon* output was imported to DESeq object using *tximport* [] and differential expression analysis was performed with standard arguments.

Transposable Element Content Analysis

5’/3’ UTR: mRNA sequences for all relevant genes were retrieved from the UCSC Genome Browser using the Table Browser functionality. Promoter sequences were defined as all genomic content 2000 nucleotides upstream of the transcription start site. FASTA files containing either mRNA or promoter sequences were cleared of duplicates and renamed using a custom *Python* script. All resulting FASTA files were processed using *RepeatMasker* [] with the following option flags:

*-no\_is -nolow -s -species Human -pa 8*

Parsing of *RepeatMasker* output was performed with bash command line tools and custom *Python* scripts. *SalmonTE* [] with default settings was used to quantify Transposable Element transcript expression from aggregate bulk RNA sequencing data and output was parsed using *R*. Analysis of all data was performed and visualized in *R* using custom scripts implementing the *Tidyverse* package.

Exon Overlap: a whole genome *.gtf* file was downloaded from the UCSC genome browser Table browser utility. This file was parsed and merged with the GENCODE v.29 reference transcriptome. This modified *.gtf* (now a *.bed* file) was passed to *bedtools* [] where the overlap function was used with the following arguments:

-a modified.gtf.bed -b all.ucsc.rmsk.genes.bed -wao -s > retained.overlap.bed

alongside a whole genome *.gtf* retrieved as described above except generated from the repeat-masked browser track. The resulting overlapped bed file was processed and visualized using custom *R* scripts.

Zinc Finger Protein Analysis

ChIP-exo data and supplementary information were extracted from supplementary data provided by *Imbeault et al* []. ZNF genes were cross referenced with *DESeq2* and *RepeatMasker* outputs to extract relevant differential expression data of ZNF proteins and Transposable Element transcripts using *R*. *RepeatMasker* output from promoter analyses was cross referenced with ChIP-exo target data to identify potential regulatory targets of differentially expressed KZNFs. Only KZNF targets with ‘score’ [see Imbeault *et al*] >= 75 were kept for analysis. Analysis of all data was performed and visualized in *R* using custom scripts.

Gene Set Enrichment Analysis

Genes determined to be significantly differentially expressed in *DESeq2* output were first ‘pre-ranked’ in *R* by the following metric:

Score metric = sin(log2FoldChange) \* -log10(p-value)

The resulting ranked files objects were processed using the *R* package *fgsea* [] alongside gene set files downloaded from msigdb [] using the *R* package *msigdbr* []. Additional code was written for select vizualizations.

Gene Ontology Analysis

Upregulated gene names were extracted from *DESeq2* output using bash command line tools. Name lists were pasted into the *Gene Ontology Consortium*’s *Enrichment Analysis* tool powered by *PANTHER*. Output data was exported as *.txt* files and parsed using bash command line tools. Parsed data was visualized using custom *R* scripts.

Single Cell Analysis:

*All single cell analyses are associated with R scripts hosted on* [*GitHub*](https://github.com/rreggiar/KRAS-AALE-Paper/blob/master/KRAS-AALE-CODE/) *and organized in the “aale.2019.single.cell.analysis.Rmd”*

10x Processing: Single cell output data was processed using 10x pipeline *CellRanger* [The *mkfastq* functionality was used to generate *fastq* files for further downstream analysis. Output was also aggregated and quantified using the *aggr* and *count* functionalities, respectively. This output was visualized using the 10x Loupe browser.

Downstream Analysis: *fastq* files generated above were passed to Salmn *alevin* [] with the following arguments:

--libtype A –chromium –dumpCsvCounts -p 16.

*alevin* was used to psuedoalign the libraries to both the GENCODE v.29 reference transcriptome as well as a composite transcriptome reference built by combining the GENCODE v.29 reference with one built from the GRCh38\_rmsk\_TE.gtf hosted by the Hammel lab at <http://labshare.cshl.edu/shares/mhammelllab/www-data/TEToolkit/TE_GTF/> . A salmon index was built from this reference with standard arguments. These alevin output matrices were imported into R using *tximport*. GSEA/ cluster correlations were calculated using the *R corr()* function. Normalization and clustering were performed with *Seurat* [] and additional code was written to handle select visualizations.

TRIMM -- Bolger, A. M., Lohse, M., & Usadel, B. (2014). Trimmomatic: A flexible trimmer for Illumina Sequence Data. *Bioinformatics*, btu170.

FASTQC -- *Andrews S. (2010). FastQC: a quality control tool for high throughput sequence data. Available online at:*[*http://www.bioinformatics.babraham.ac.uk/projects/fastqc*](http://www.bioinformatics.babraham.ac.uk/projects/fastqc)

Smit, AFA, Hubley, R & Green, P. *RepeatMasker Open-4.0*.  
2013-2015 <http://www.repeatmasker.org>.

SALMON -- Patro, R., Duggal, G., Love, M. I., Irizarry, R. A. & Kingsford, C. Salmon provides fast and bias-aware quantification of transcript expression. Nature Methods 14, 417 (2017).

SLEUTH -- Harold J. Pimentel, Nicolas Bray, Suzette Puente, Páll Melsted and Lior Pachter, [Differential analysis of RNA-Seq incorporating quantification uncertainty](http://www.nature.com/nmeth/journal/vaop/ncurrent/full/nmeth.4324.html), Nature Methods (2017), advanced access http://dx.doi.org/10.1038/nmeth.4324.

DESEQ2 -- Love, M.I., Huber, W., Anders, S. Moderated estimation of fold change and dispersion for RNA-seq

data with DESeq2 Genome Biology 15(12):550 (2014)

TXIMPORT -- Charlotte Soneson, Michael I. Love, Mark D. Robinson (2015): Differential analyses for RNA-seq: transcript-level estimates improve gene-level inferences. F1000Research

SalmonTE -- Guo, C., Jeong, H.-H., Hsieh, Y.-C., Klein, H.-U., Bennett, D.A., Jager, P.L.D., Liu, Z., and Shulman, J.M. (2018). Tau Activates Transposable Elements in Alzheimer’s Disease. Cell Reports *23*, 2874–2880.

Imbeault, M., Helleboid, P.-Y., and Trono, D. (2017). KRAB zinc-finger proteins contribute to the evolution of gene regulatory networks. Nature *543*, 550.

R Core Team (2018). R: A language and environment for statistical computing. R

Foundation for Statistical Computing, Vienna, Austria. URL <https://www.R-project.org/>.

Hadley Wickham (2017). tidyverse: Easily Install and Load the 'Tidyverse'. R package

version 1.2.1. <https://CRAN.R-project.org/package=tidyverse>

GSEA -- Sergushichev A (2016). “An algorithm for fast preranked gene set enrichment analysis using cumulative statistic calculation.” *bioRxiv*. doi: [10.1101/060012](https://doi.org/10.1101/060012), <http://biorxiv.org/content/early/2016/06/20/060012>.

MSigDB -- Liberzon et al. 2011 Bioinformatics 27(12):1739-40

ALL GO REFS:

Ashburner et al. Gene ontology: tool for the unification of biology (2000) *Nat Genet* **25(1)**:25-9. [Online at Nature Genetics](https://www.ncbi.nlm.nih.gov/pubmed/10802651).

GO Consortium, Nucleic Acids Res., 2017

Mi et al., Nucleic Acids Res., 2017

CELLRANGER -- Jennifer Bryan (2016). cellranger: Translate Spreadsheet Cell Ranges to Rows and Columns. R package version 1.1.0. <https://CRAN.R-project.org/package=cellranger>

SEURAT -- Stuart and Butler et al. Comprehensive integration of single cell data. bioRxiv (2018).