SAND2021-8385PE

MLDL

Machine Learning and Deep Learning Conference 2021

Data Science for Characterization of Genome Noise/Mutation

- Stephen J. Verzi/08722
- Raga Krishnakumar/08623, Drew Levin/08721
- Callie Boskin/08623, Kelly Williams/08623,
- Dan Krofcheck/08722

- LDRD

Abstract

Detection of genome editing is a challenging and broad area of research, and this research investigates whether data science techniques are an appropriate solution methodology.

We have gathered example data, for both edit and non-edit (or control) situations, and we have developed a data processing and analysis pipeline which includes genomic noise counting as well as machine learning (Random Forest and Deep Neural Network) and anomaly detection models.

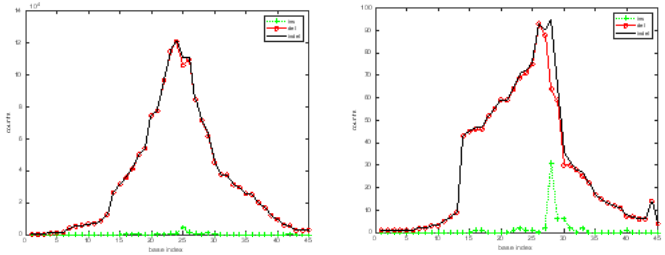
We will present results on genomic noise characterization as well as edit detection.

Problem you are trying to solve

In this work we are attempting to determine if either intended or unintended edits to the human genome, using the homology directed repair (HDR) mechanism, can be distinguished from normal mutation and/or machine error from “typical” deep sequence reads. The figure below shows that edit detection is possible when targeted sequence (i.e., Amplicon) reads are used, as is expected.

Verification of learnability

Wang et al., 2018 Cho et al., 2014



Here we verify the ML learnability signature in Amplicon data at multiple edit target sites (CCR5 and WDR5) from different experiments (Cho et al., 2014 and Wang et al., 2018).

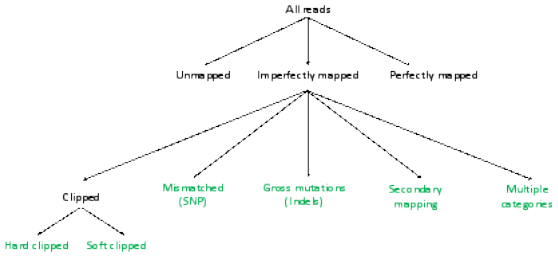
Algorithmic approach of your solution

We construct a data processing pipeline (see figure below) to facilitate distinguishing amongst:

- normal DNA sequence variations,
- sequencer machine error,
- penetration of editing: failure of editing in many cells in a tissue, and occasional NHEJ (with different outcomes in different cells in a tissue), even when attempting HDR.

We apply machine learning (ML) to deep sequence reads to characterize signals in them that allow distinction of normal sequence variations (either due to mutation or machine error) from genome editing.

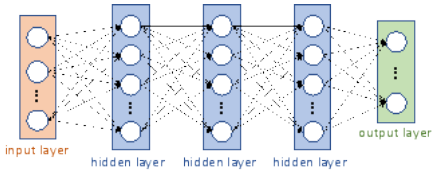
Data Processing Pipeline



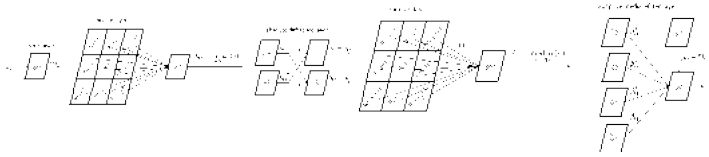
As part of the data pipeline, we focus upon and investigate imperfectly mapped reads.

Random forest

Deep CNN



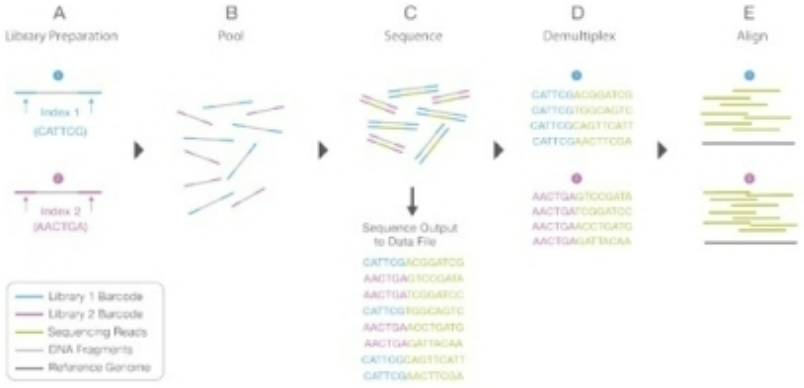
Deep spiking anomaly detection



Description of the data used

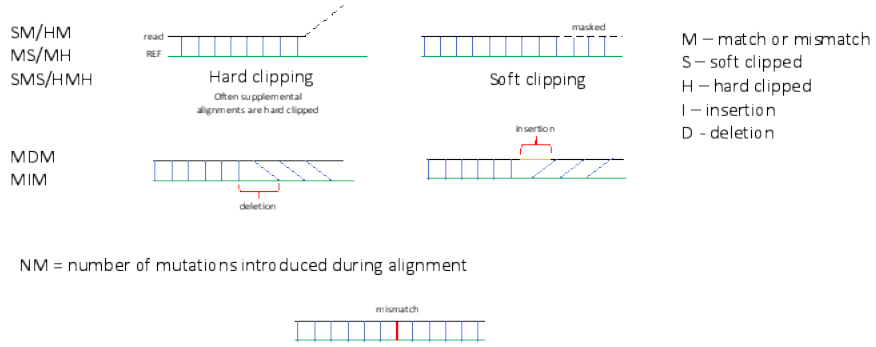
Sequence read data is available through various sources (including NCBI), but the formats vary significantly, specifically with respected to the experiment under study, result in the need for proper pre-processing, including alignment, filtering and match/mismatch counting.

Alignment (using experimental sequence reads)



Non perfect matching after alignment

SAM Flag 2048 = supplemental alignment (alignment at a secondary location, eg. chimeric reads)



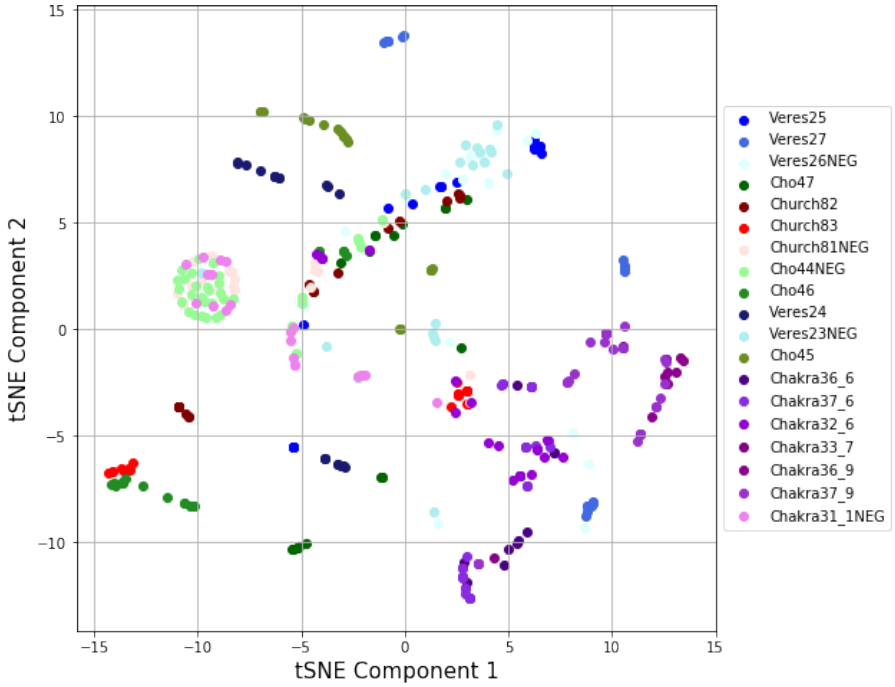
NM = number of mutations introduced during alignment

- Feature set for ML algorithms
  - Non-matches
  - Total reads
  - Matches
  - Deletions
  - Insertions
  - Clips
    - soft
    - hard
  - Nucleotides: A,C,G,T
  - others

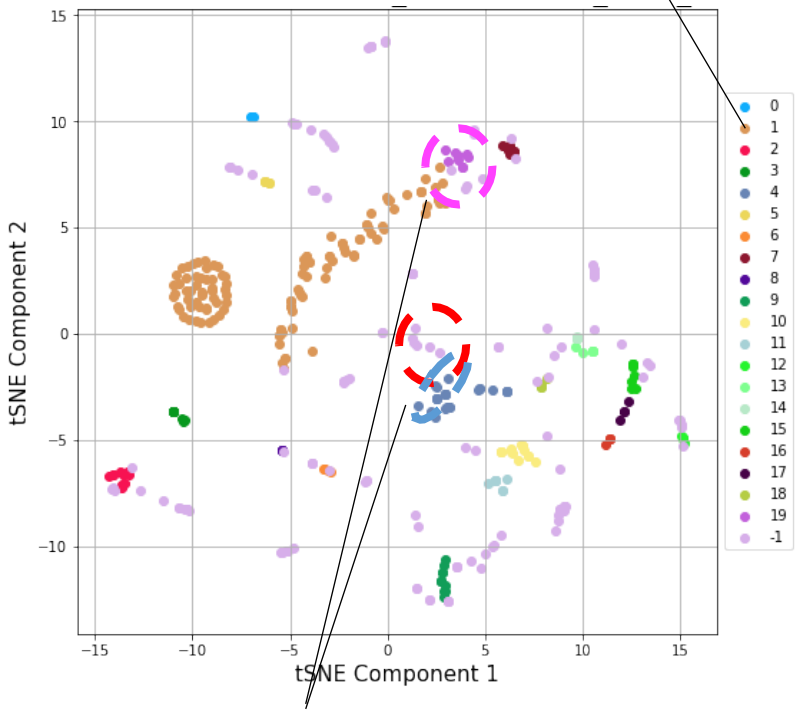
Results

tSNE (t-distributed Stochastic Neighbor Embedding) visualization

DBSCAN clustering



Most negatives are in cluster 1



Non-cluster 1 negatives are either in cluster 4, 19 or singletons (not in any cluster)

Results

Tree-based ensembles – use SMOTE to amplify the amount of data we have

	pre-SMOTE	post-SMOTE
Matches	-0.254619	-0.292882
random	0.012406	0.014812
SD_Softclips	0.097822	0.097472
SD_Insertions	0.142293	0.161023
Soft clips	0.149396	0.165603
Insertions	0.150198	0.172695
Deletions	0.209405	0.248075
Hard clips	0.235041	0.277336
SD_Deletions	0.235192	0.277489
SD_Hardclips	0.24263	0.285641
SD_Matches	0.244244	0.276878
Sum indels clips	0.257533	0.295573
delPeaks	0.280862	0.330604
insPeaks	0.292268	0.327164
crossIns	0.295258	0.330344
crossDel	0.381553	0.440137
crossHard	0.384375	0.443334
hPeaks	0.404201	0.465554
sPeaks	0.406744	0.442842
allPeaks	0.43573	0.475255
crossSoft	0.43661	0.462243
crossAll	0.545992	0.597058
Class	1	1

Random Forest vs XGBoost (hyperparameter

	Xgboost	RF
1	0.9756	0.9982
2	0.9774	0.9982
3	0.9737	0.9826
4	0.9765	0.9982
5	0.9759	0.9982
6	0.9765	0.9466
7	0.9744	0.9982
8	0.9755	0.8698
9	0.9756	0.997
10	0.9788	0.9436

Random Forest vs XGBoost (test data set metrics)

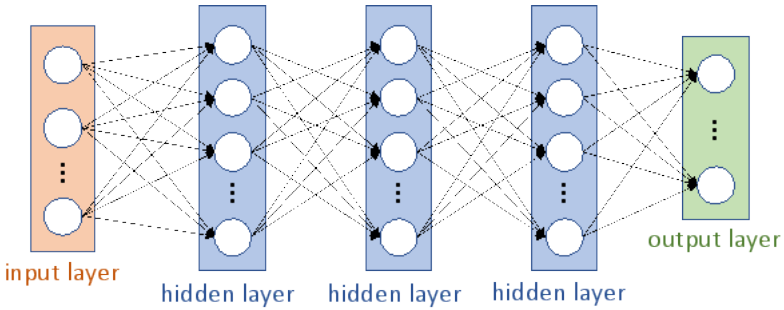
Test set	Algorithm	Accuracy	Precision	Recall	Specificity	F1
Data set 1	RF	0.995	0.5	0.1	0.9995	0.1666
Data set 1	XGB	0.995	0.4444	0.4	0.9983	0.421
Data set 2	RF	0.97	0.96226	0.51	0.99875	0.66667
Data set 2	XGB	0.985	0.93023	1	0.98125	0.96386

Conclusions:

- Overall high specificity and accuracy
- Variability in precision and recall depending on data
- XGB > RF for recall

Results

Deep convolutional neural network (CNN)



- 1D CNNs in each hidden layer
- Trained over
  - All edit sites (positive)
  - Each control file (negative)
    - Cho
    - Church
    - Veres
    - 1000 genomes

- Edit detection performance
  - Using window
    - size = 500
    - step = 250
  - ~75 positive hits across each control file

Results

Anomaly (edit) detection using deep spiking neural network

- Bio-inspired simplification of CRISPR edit precision prediction [Chakrabarti, 2019]
  - Using spiking adaptive median-filtering [Verzi, 2018]

$$o_{ij} = \begin{cases} \hat{\rho}_{ij}^{1}, & \text{if } \exists m, \hat{\rho}_{ij}^{m} > \theta^{m} \\ x_{ij}, & \text{otherwise} \end{cases}$$

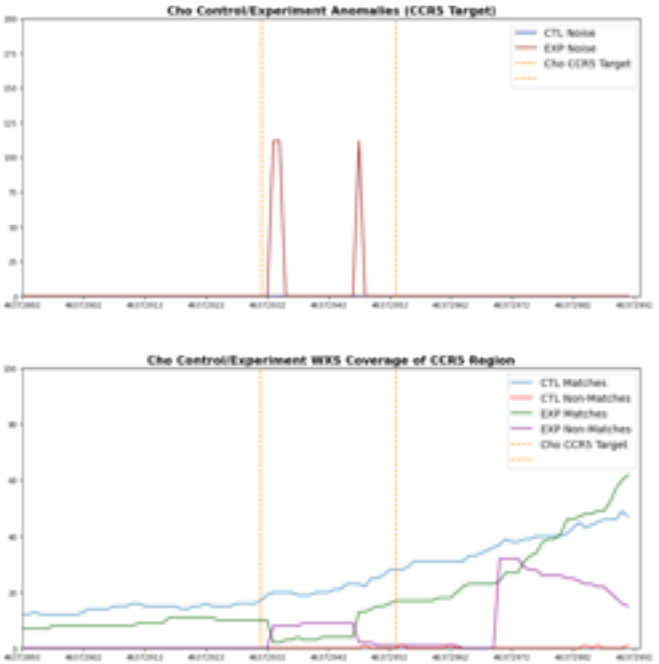
$$\hat{\rho}_{ij}^{m} = \underset{x \in \Omega_{ij}^{m}}{\text{median}}\{x\}$$

$$\Omega_{ij}^{m} = \{x_{lr} \mid i - m \le l \le i + m, j - m \le r \le j + m\}$$

$$\theta^{m} = s \cdot \underset{x \in \Omega_{ij}^{m}}{\text{median}}\left\{\hat{\rho}_{ij}^{m} - x\right\} + \delta^{m}$$

$$\delta^{m} = \frac{\left((2m+1)^{2}-1\right)}{2}$$

- Edit detection at CCR5 (in chr 3)



detecting edges of edit region

Conclusions

What is the one-sentence summary of your work that you would want a technical person to remember?

We are exploring the utility of data science and machine learning at the forensic end of genome edit detection, with an eye towards the potential for future use in pre-forensic application as well as in synthetic bioinformatics modeling.

What is the one-sentence summary of your work that you would want a manager or program developer to remember?

In this research, we are bringing together expertise from both bioinformatics and machine learning at Sandia to develop a capability vital to national security in genomics.

Indicate presentation time between 15 and 30 minutes or a 5-minute spotlight. We will try to accommodate.

We can present in any available time slot between 15 and 30 minutes in length.