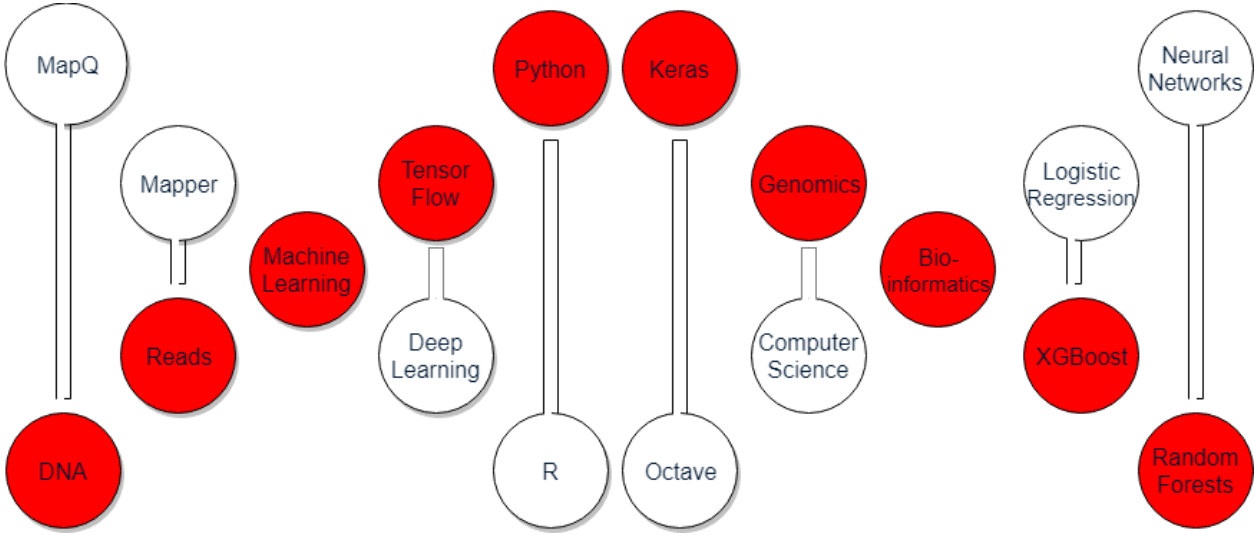
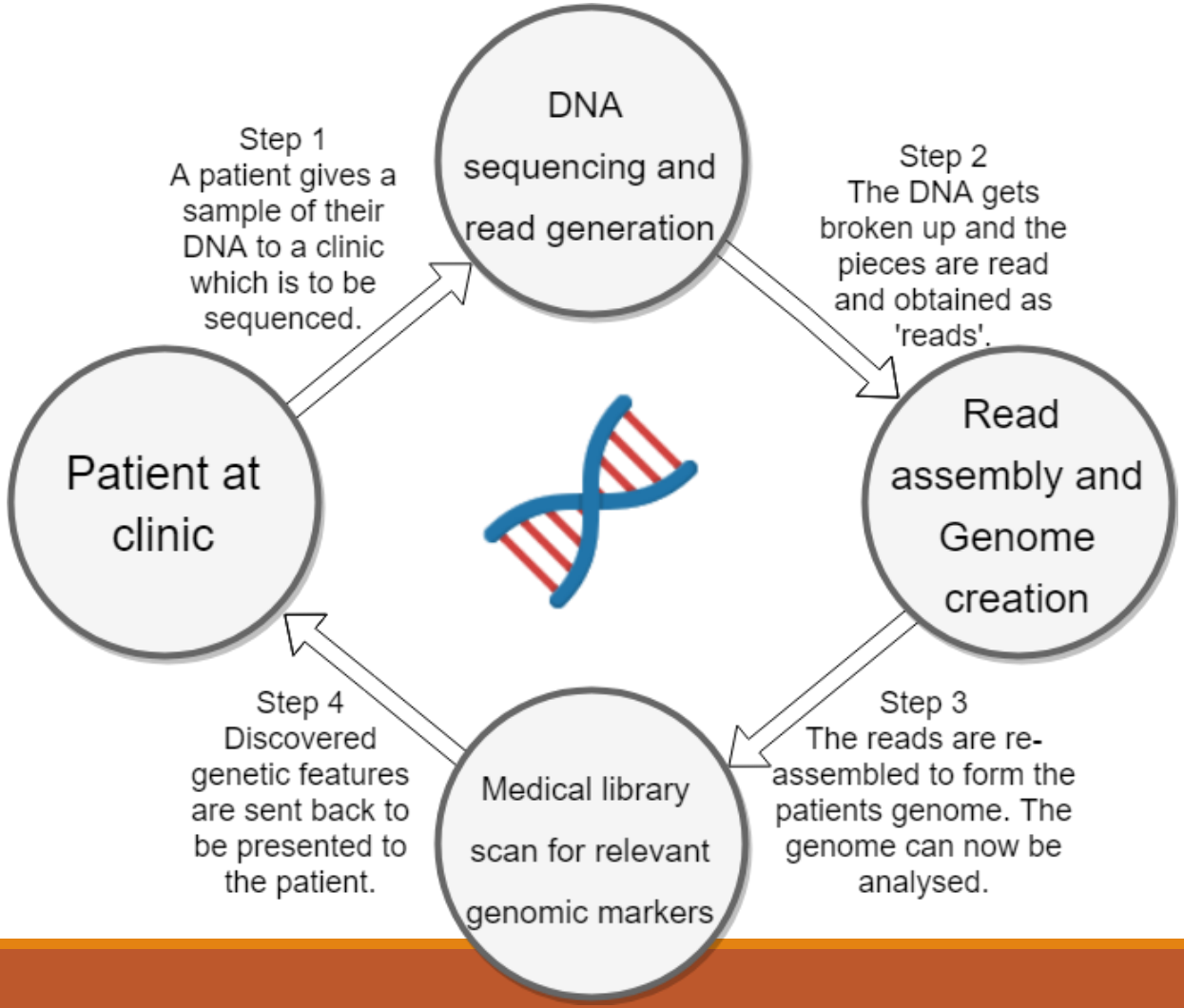
Adaptive calibration of MAPQ scores using machine learning techniques

Soroush H.



Contents


DNA sequencing and read generation
Step 1
A patient gives a sample of their DNA to a clinic which is to be sequenced.
Step 2
The DNA gets broken up and the pieces are read and obtained as 'reads'.
Patient at clinic
Read assembly and Genome creation
Step 4
Discovered genetic features are sent back to be presented to the patient.
Step 3
The reads are re-assembled to form the patients genome. The genome can now be analysed.
Medical library scan for relevant genomic markers

Why is this process so important?

-Single Nucleotide Polymorphism occurs at a rate of 1 in 1000 in humans. There's a lot that can be missed.

-A wrong read placement can change a diagnosis.
Go from potential 'Cancer Marker' to 'Not at risk individual'.

-A doctor would rather not make assumptions in regards to a patients health. They would rather tell a few extra people to get tested for cancer because they're unsure, than completely miss it.

What are MapQ Scores?

-MapQ score is a measurement of confidence. How confident the tool is that it has placed the piece of DNA (read) in the correct location.

-Formula: -10log10(1-p), where p is the probability it's correct.
Ex: Score of 20 = 99%. 60 = 99.9999%.

-These are generated for each read along with the presumed position in the genome by the mapping tools.

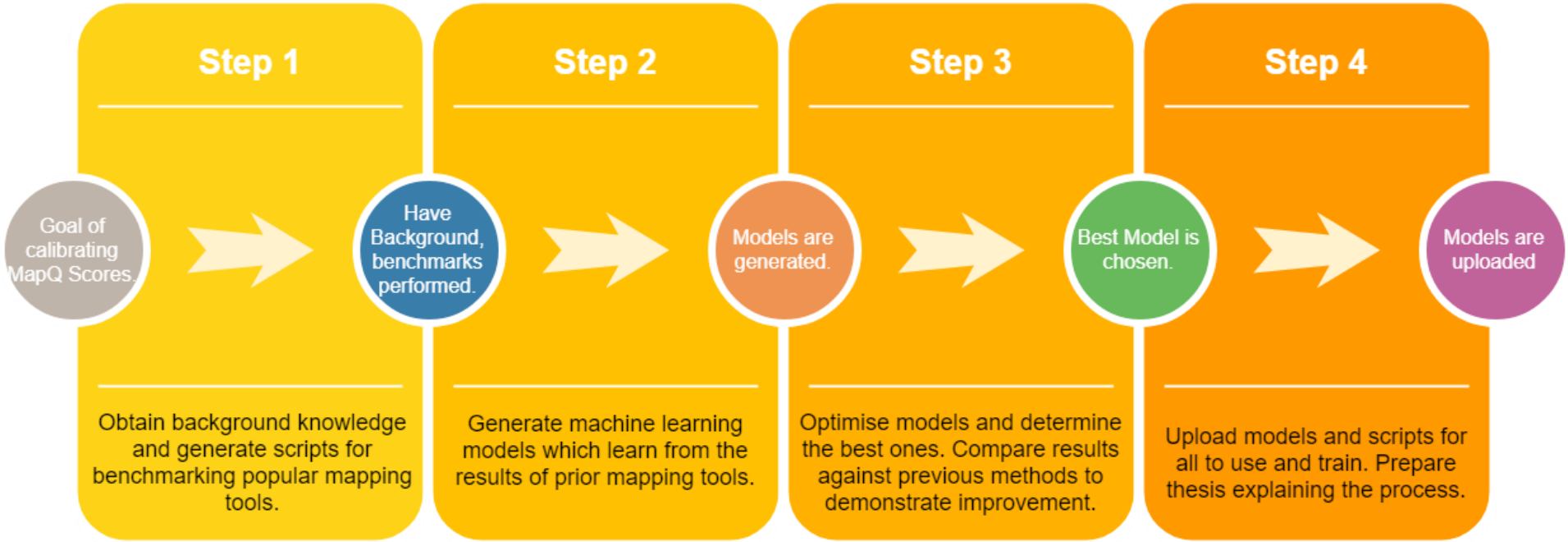
Current MapQ Accuracies

-Many mappers have difficulty maintaining any score accurately.

-A lot of tools such as BWA and MiniMapper2 will place ones they're uncertain of into lower score regions (0-20), resulting in a more true positive lower region and damaging overall read accuracy.

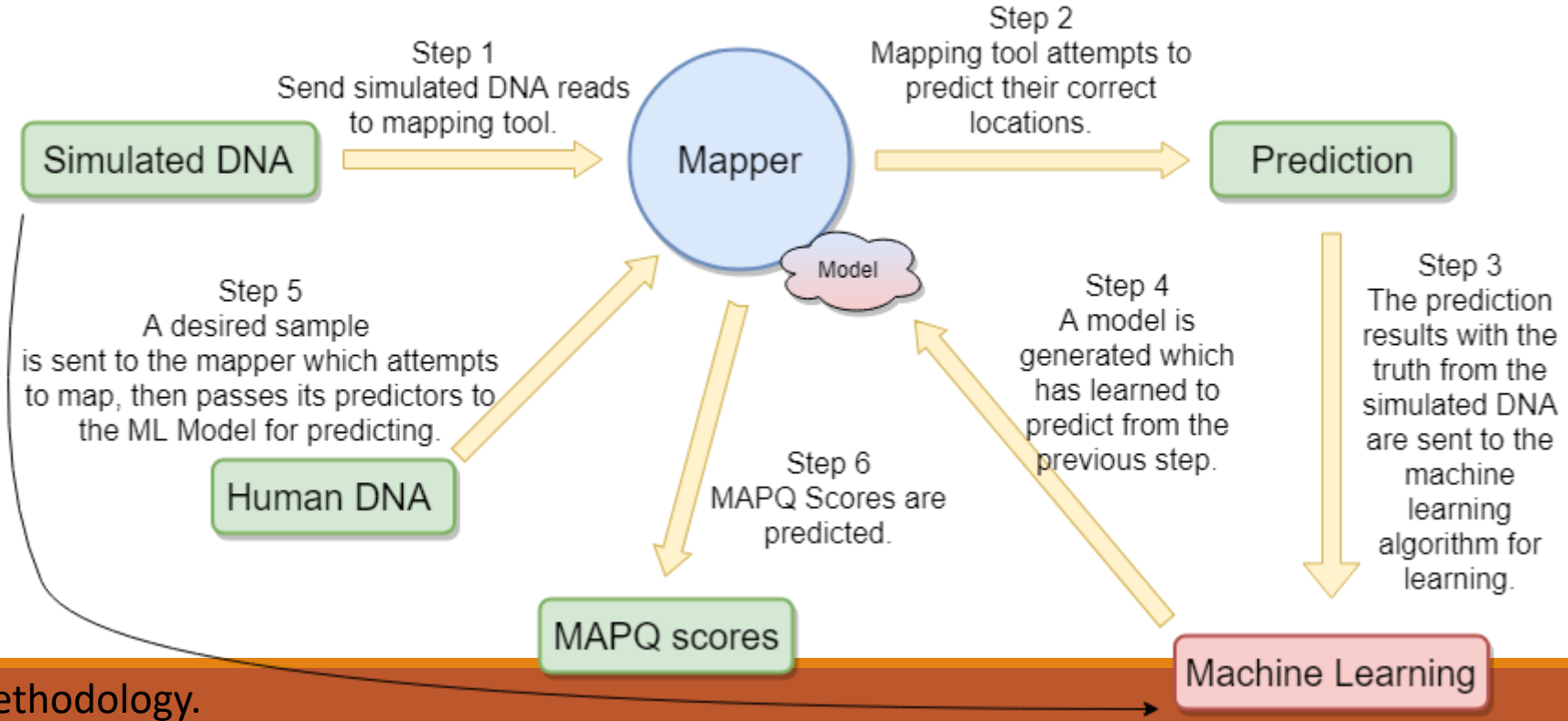
-Can we obtain better MapQ accuracy than the mappers themselves?

Workflow Process



2. How it was approached.

Detailed Process



Utilized Machine Learning Methods

-Logistic Regression
Simple, easy and old. (1940s)

-Random Forests
Flexible, fast to implement. Different method of learning than L.R.

-XGBoost
Similar to Random Forests. Has been winning contests left and right.

-Support Vector Machines
Shown potential, ran in downtime while Neural networks were being run.

-Neural Networks (Deep Learning)
Shown the biggest potential. Is being used for everything now a days.

Required Work

-All machine learning methods have their own background, and only by actually understanding them can you properly tune the parameters to suit your data. (I would recommend the coursera course and the many audio podcasts on machine learning).

-Learning octave was useful for implementing Logistic Regression. It handled large data better than the python version I wrote for Logistic Regression.

-Learning how the TensorFlow framework works helped with Deep Learning.

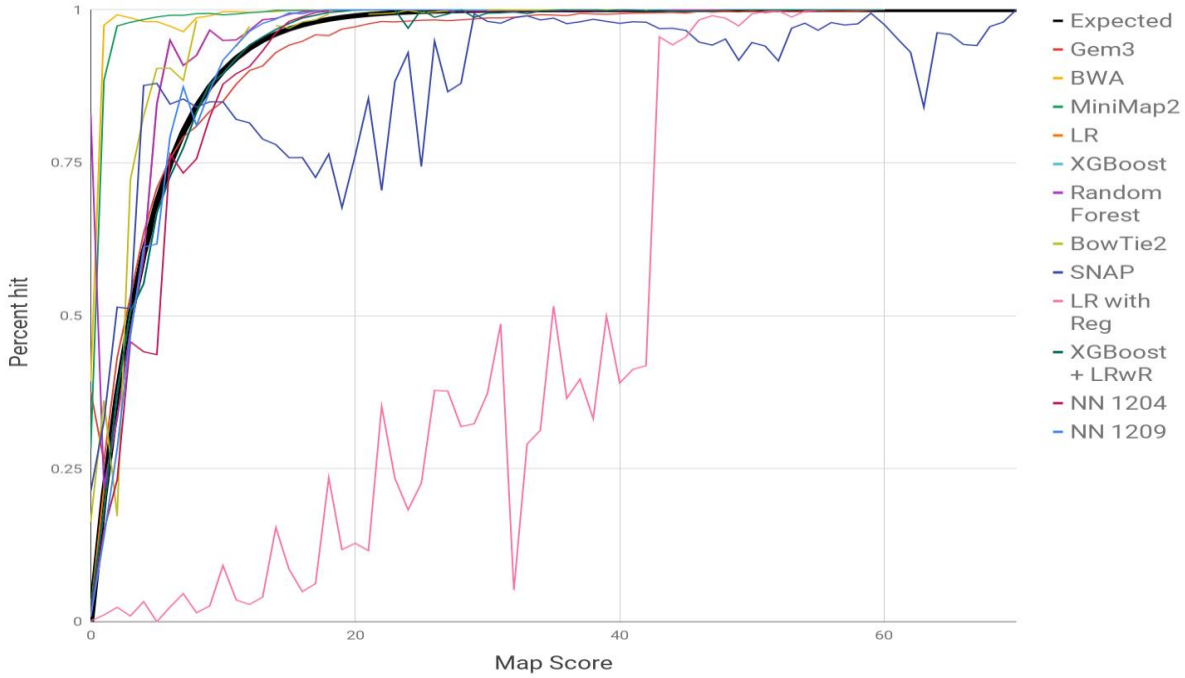
-Over 2000 Unique Machine Learning models were generated each with their own code. (hundreds of millions of rows of DNA data with over a dozen columns were used to learn.)

-Simultaneously ran on clusters, laptop, and later a Nvidia 1080Ti GPU for months non stop (days, nights, weekends). Some models would take days to generate, most a few hours.

-Benchmarking scripts were written which were ran on any remaining threads (when memory allowed) to analyze the results and prepare the next set of scripts.

Results

Overall MapQ Score Accuracy Visualized



5. Improved MapQ Score Accuracy?

Overall MapQ Score Accuracy Quantified

Popular Tools	
Model	AUC vs Expected
BowTie2	9.05749
SNAP	5.71882
BWA	3.85854
MiniMapper2	3.74596
Gem3	1.11743

Machine Learning Methods	
Model	AUC vs Expected
Logistic Regression with Regularization (LRwR)	29.81349
Random Forest	1.83424
Logistic Regression	**1.04164**
Neural Network (1204)	**1.02618**
Neural Network (1209)	**0.56146**
XGBoost + (LRwR)	**0.286947**
XGBoost	**0.24996**

20:60. Why neglect MapQ accuracy?

-Scientists will discard lower quality regions, to make sure they're not getting false reads (any region below a confidence of 20, aka 99%.)

-This is to lower the number of false positives of what someone will actually be using.

-Can we have a better 20:60 region while maintaining our better MapQ score accuracy?

Desired Region. 20 : 60 results

Popular Tools			
Method	TP 20 to 60	FP 20 to 60	% Accuracy
SNAP	8810864	26893	0.996957
BWA*	9040397	2751	0.999697
BowTie2	8508834	2478	0.999709
Gem3*	9217282	1612	0.999825
MiniMapper 2*	8714557	692	0.999921

Machine Learning Methods			
Method	TP 20 to 60	FP 20 to 60	Accuracy
LRwR	9257446	667335	0.932761
Random Forest	8635614	1719	0.999801
Logistic Regression	7399193	904	0.999878
XGBoost*	9201165	976	0.999895
XGB + LRwR	**8713879**	**429**	**0.999951**
Neural Network 1204*	**8788038**	**65**	**0.999992**
Neural Network 1209*	**8374355**	**55**	**0.999993**

6. Improved 20 through 60 region?

Out of 10 million 150bp reads. *Results averaged and rounded from 6 samples. **SNAP is 20:70

Great, but what really matters?

-Mappers have difficulty ever reaching a score of 60, even though they ‘confidently’ place most of their predictions at that score.

-If their priority is having the least false positives, and they sacrifice MapQ accuracy, what is their best capability, for someone who wants the most sensitive results possible?

Best possible range. Single Digit FP's attained!

Popular Tools				
Method	TP	FP	% Accuracy	Theoretical MapQ Score
BWA*	8687646	2296	0.9997368	35.5
SNAP	8267170	676	0.9999182	40.9
MiniMapper2 *	8061615	563	0.9999308	41.6
Gem3*	8928359	54	0.9999940	52.2

Machine Learning Methods				
Method	TP	FP	% Accuracy	Theoretical MapQ Score
SVM	8942055	105192	0.9883730	19.3
Random Forest	8055968	749	0.9999070	40.3
LRwR	8015468	46	0.9999943	52.4
XGBoost + LRwR	7876320	28	0.9999965	54.6
Neural Network (1204)*	**7798136**	**3**	**0.9999996**	**64.0**
Neural Network (1209)*	**6345552**	**1 (0.8)**	**0.9999999**	**>70**

7. Improved best possible Accuracy?

Out of 10 million 150bp reads. *Results averaged and rounded from 6 samples. **SNAP is 20:70

Other Measurement methods. Total Accuracy, F1 Score

-Overall Accuracy can be potentially important for someone looking to just take all mapped positions. Max hits possible, with a high accuracy goal.

-F1 Score is a commonly used method of measuring tool performances. It's the harmonic average of Precision & Recall. 2(P*R)/(P+R).

-Precision: number of TP from the number retrieved TP/(TP +FP).
-Recall: number of TP from total taken. TP/(TP+FN).

Overall Accuracy and F1 Score

Popular Tools (Sorted by F1)			
Model	Total TP	% Accuracy	F1 Score
MiniMapper2*	9344965	0.977317	0.954361
SNAP	9181727	0.948769	0.957341
BowTie2	9220455	0.926781	0.959442
BWA*	9529593	0.973362	0.961868
Gem3*	**9675271**	0.959546	0.962752

Machine Learning Methods (Sorted by F1)			
Model	Total TP	% Accuracy	F1 Score
Random Forest	8409599	0.858343	0.913610
XGB + LRwR	9254563	0.939813	0.961285
Logistic Regression	9255631	0.937215	0.961343
LRwR	9257895	0.932653	0.961465
NN 1204v14**	9257987	0.931899	0.961470
XGBoost*	**9634493**	**0.973097**	**0.979089**
Neural Network 1204*	**9675271**	**0.979401**	**0.983346**
Neural Network 1209*	**9675271**	**0.979056**	**0.983372**

8. Best Overall Accuracy and F1 Score?

Out of 10 million 150bp reads. *Results averaged and rounded from 6 samples.
**NN 1204v14 places all its hits at a MapQ Score of 0, 10, 11

Performance Results

Model	File Size
Logistic Regression	66 byte Text file
XGBoost	118 kb
NN model	44 kb

Model	Time to run (seconds)
Gem3	177
MiniMapper2	1312
BWA	5946
SNAP	>6000
BowTie2	>6000

Require Gem3 to be run (Total time if merged)	
Model	Time to run (seconds)
Logistic Regression	~0 (177)
XGBoost	351 (528)
Neural Networks*	1418 (1595)

9. Performance metrics.

10 million 150bp reads mapped. *Neural Networks ran on slower laptop CPU, potential time to gain.

Summary of Results

Obtained through Neural Networks:

- -Best MapQ score accuracy
- -Best F1 score
- -Greatest overall accuracy (0:60)
- -Greatest 20:60 region accuracy
- -Greatest possible accuracy

Bonus:

- -XGBoost can be combined with the Logistic Regression with Regularization for great results, including the best MapQ accuracy. Models are also trained and generated much more quickly than Neural Networks.

Future work

Models can be further improved given:

-More or better predictors.

-More positions to place the read (already given by tools such as BWA).

-More time to run neural network parameters and samples.

-Potentially through the combining of multiple ML methods as seen with XGBoost + LRwR.

Models are flexible:

-Can be generated for specific parts of the genome (ex: Chr1).

-For each popular read length (for 25bp, 50bp, 100bp, etc).

-For different mapping tools if the creators so desire to use their predictors (MiniMapper2, BWA, etc).

-The code and models are all open source and available now, entirely reproduceable.

-Publication is in the works.

Questions?

Thank you for your time.