Ensemble machine learning modeling for the prediction of artemisinin resistance in malaria

Colby T. Ford[1,2,*] **and Daniel Janies**[1]

[1]**Department of Bioinformatics and Genomics, University of North Carolina at Charlotte, Charlotte, North Carolina, 28223, USA**
[2]**School of Data Science, University of North Carolina at Charlotte, Charlotte, North Carolina, 28223, USA**
[*]**Corresponding author: Colby T Ford (colby.ford@uncc.edu)**

Abstract

Resistance in malaria is a growing concern affecting many areas of Sub-Saharan Africa and Southeast Asia. Since the emergence of artemisinin resistance in the late 2000s in Cambodia, research into the underlying mechanisms has been underway.
The 2019 Malaria Challenge posited the task of developing computational models that address important problems in advancing the fight against malaria. The first goal was to accurately predict artemisinin drug resistance levels of *Plasmodium falciparum* isolates, as quantified by the IC_{50}. The second goal was to predict the parasite clearance rate of malaria parasite isolates based on *in vitro* transcriptional profiles.
In this work, we develop machine learning models using novel methods for transforming isolate data and handling the tens of thousands of variables that result from these data transformation exercises. This is demonstrated by using massively parallel processing of the data vectorization for use in scalable machine learning. In addition, we show the utility of ensemble machine learning modeling for highly effective predictions of both goals of this challenge. This is demonstrated by the use of multiple machine learning algorithms combined with various scaling and normalization preprocessing steps. Then, using a voting ensemble, multiple models are combined to generate a final model prediction.

Keywords

malaria, *Plasmodium falciparum*, machine learning, parallel computing, Apache Spark, big data, artemisinin, DREAM Competition

Introduction

Malaria is a serious disease caused by parasites belonging to the genus *Plasmodium* which are transmitted by *Anopheles* mosquitoes in the genus. The World Health Organization (WHO) reports that there were 219 million cases of malaria in 2017 across 87 countries [1]. *Plasmodium falciparum* poses one of greatest health threats in Southeast Asia, being responsible for 62.8% of malaria cases in the region in 2017 [1].

Artemisinin-based therapies are among the best treatment options for malaria caused by *P. falciparum* [2]. The use of artemisinin in combination with other drugs, called artemisinin combination therapies, are the best treatment options today against malaria infections.

However, emergence of artemisinin resistance in Thailand and Cambodia in 2007 has been cause for research [3]. While there are polymorphisms in the kelch domain–carrying protein K13 in *P. falciparum* that are known to be associated with artemisinin resistance, many of the underlying molecular mechanisms that confer resistance remains unknown [4]. In early 2020, Birnbaum et al. discovered that the highly-conserved gene *kelch13* is associated with a molecular mechanism that allows the parasite to feed on host erythrocytes by endocytosis of hemoglobin [5]. Given that artemisinin is activated by hemoglobin degradation products, these mutations can confer resistance to artemisinin. The established pharmacodynamics benchmark for *P. falciparum* sensitivity to artemisinin-based therapy is the parasite clearance rate [6, 7]. Resistance to artemisinin-based therapy is considered to be present with a parasite clearance rate greater than five hours[8]. By understanding the genetic factors that affect resistance in malaria, targeted development can occur in an effort to abate further resistance or infections of resistant strains.

Previous research has shown success in applying similar machine learning methods in the explanation of genetic differences in plants [9], fungi [10], and even humans [11]. Previous work in machine learning-based tropical disease research, including malaria and other diseases, has shown effective in drug discovery [12, 13] and in the understanding of degradomes [14]. Also, other machine learning work in malaria has focused on the identification and diagnosis of malaria using image classification [15, 16, 17].

In this work, we create multiple machine learning-based models to address these issues around artemisinin resistance and parasite clearance. Given that the interpretation and analysis of many genes and their effects on resistance may be tedious, machine learning allows for a more power investigation into this relationship. Plus, we employ model explainability methods to help rank particular genes of interest in the malaria genome.

Prediction of artemisinin IC_{50}

First, we created a machine learning model to predict the IC_{50} of malaria parasites based on transcription profiles of experimentally-tested isolates. IC_{50}, also known as the half maximal inhibitory concentration, is the drug concentration at which 50% of parasites die. This value indicates a population of parasites' ability to withstand various doses of antimalarial drugs, such as artemisinin.

Methods

Training data was obtained from the 2019 DREAM Malaria Challenge [18, 19]. The training data consists of gene expression data of 5,540 genes of 30 isolates from the malaria parasite, *Plasmodium falciparum*. For each malaria parasite isolate, transcription data was collected at two time points [6 hours post invasion (hpi) and 24 hpi], with and without treatment of dihydroartemisinin (the metabolically active form of artemisinin), each with a biological replicate. This yields a total of at eight data points for each isolate. The initial form of the training dataset contains 272 rows and 5,546 columns, as shown in Table 1.

The transcription data was collected as described in Table 2. The transcription data set consists of 92 non-coding RNAs (denoted by gene IDs that begins with 'MAL'), while the rest are protein coding genes (denoted by gene IDs that start with 'PF3D7'). The feature to predict is DHA_IC50.

Data preparation

We used Apache Spark [21] to pivot the dataset such that each isolate was its own row and each of the transcription values for each gene and attributes (i.e. timepoint, treatment, biological replicate) combination was its own column. This exercise transformed the training dataset from 272 rows and 5,546 columns to 30 rows and 44,343 columns, as shown in Table 3.

We completed this pivot by slicing the data by each of the eight combinations of timepoint, treatment, and biological replicate, dynamically renaming the variables (genes) for each slice, and then joining all eight slices back together.

By using the massively parallel architecture of Spark, this transformation can be completed in a minimal amount of time on a relatively small cluster environment (e.g., <10 minutes using a 8-worker/36-core cluster with PySpark on Apache Spark 2.4.3).

Sample_Name	Isolate	Timepoint	Treatment	BioRep	$Gene_1$	...	$Gene_{5540}$	DHA_IC50
isolate_01.24HR.DHA.BRep1	isolate_01	24HR	DHA	BRep1	0.008286	...	-2.48653	2.177
isolate_01.24HR.DHA.BRep2	isolate_01	24HR	DHA	BRep2	-0.87203	...	-1.79457	2.177
isolate_01.24HR.UT.BRep1	isolate_01	24HR	UT	BRep1	0.03948	...	-2.49517	2.177
isolate_01.24HR.UT.BRep2	isolate_01	24HR	UT	BRep2	0.125177	...	-1.73531	2.177
isolate_01.6HR.DHA.BRep1	isolate_01	6HR	DHA	BRep1	1.354956	...	-0.82169	2.177
isolate_01.6HR.DHA.BRep2	isolate_01	6HR	DHA	BRep2	-0.21807	...	-1.61839	2.177
isolate_01.6HR.UT.BRep1	isolate_01	6HR	UT	BRep1	1.31135	...	-2.62262	2.177
isolate_01.6HR.UT.BRep2	isolate_01	6HR	UT	BRep2	0.997722	...	-2.24719	2.177
...	...	...	...	...	...	...	...	...
isolate_30.6HR.UT.BRep2	isolate_30	6HR	UT	BRep2	-0.26639	...	-1.72273	1.363

Table 1. Initial IC_{50} model training data format. Note that for Treatment, *UT* represents untreated samples and *DHA* represents samples treated with dihydroartemisinin.

	Training Set
Array	Bozdech
Platform	Printed
Plexes	1
Unique Probes	10159
Range of Probes per Exon	N/A
Average Probes per Gene	2
Genes Represented	5363
Transcript Isoform Profiling	No
ncRNAs	No
Channel Detection Method	Two Color
Scanner	PowerScanner
Data Extraction	GenePix Pro

Table 2. IC_{50} training data information. (Adapted from Turnbull et al., (2017) PLoS One[20])

Isolate	DHA_IC50	$hr24\_trDHA\_br1\_Gene_1$	$hr24\_trDHA\_br2\_Gene_1$	...	$hr6\_trUT\_br2\_Gene_{5540}$
isolate_01	2.177	0.008286	-0.87203	...	-2.24719
...	...	...	...	...	...
isolate_30	1.363	0.195032	0.031504	...	-1.72273

Table 3. Post-transformation format of the IC_{50} model training data.

Lastly, the dataset is then vectorized using the Spark `VectorAssembler`, and converted into a Numpy[22]-compatible array. Vectorization allows for highly scalable parallelization of the machine learning modeling in the next step.

Machine learning

We used the Microsoft Azure Machine Learning Service [23] as the tracking platform for retaining model performance metrics as the various models were generated. For this use case, 498 machine learning models were trained using various scaling techniques and algorithms. Scaling and normalization methods are shown in Table 14. We then created two ensemble models of the individual models using Stack Ensemble and Voting ensemble methods.

The Microsoft AutoML package [24] allows for the parallel creation and testing of various models, fitting based on a primary metric. For this use case, models were trained using Decision Tree, Elastic Net, Extreme Random Tree, Gradient Boosting, Lasso Lars, LightGBM, RandomForest, and Stochastic Gradient Decent algorithms along with various scaling methods from Maximum Absolute Scaler, Min/Max Scaler, Principal Component Analysis, Robust Scaler, Sparse Normalizer, Standard Scale Wrapper, Truncated Singular Value Decomposition Wrapper (as defined in Table 14). All of the machine learning algorithms are from the *scikit-learn* package[25] except for LightGBM, which is from the *LightGBM* package[26]. The settings for the model sweep are defined in Table 4. The 'Preprocess Data?' parameter enables the scaling and imputation of the features

in the data. Note that these models were evaluated using random sampling of the input training dataset provided by the DREAM Challenge, though the evaluation within the challenge was performed on an unlabelled testing dataset. The metrics in the Results section below reflect the evaluation on the sampled training data.

Parameter	Value
Task	Regression
Number of Iterations	500
Iteration Timeout (minutes)	20
Max Cores per Iteration	7
Primary Metric	Normalized Root Mean Squared Error
Preprocess Data?	True
k-Fold Cross-Validations	20 folds

Table 4. Model search parameter setting for the IC_{50} model search.

Once the 498 individual models were trained, two ensemble models (voting ensemble and stack ensemble) were then created and tested. The voting ensemble method makes a prediction based on the weighted average of the previous models' predicted regression outputs whereas the stacking ensemble method combines the previous models and trains a meta-model using the elastic net algorithm based on the output from the previous models. The model selection method used was the Caruana ensemble selection algorithm[27].

Results

The voting ensemble model (using soft voting) was selected as the best model, having the lowest normalized Root Mean Squared Error (RMSE), as shown in Table 5. The top 10 models trained are reported in Table 6. Having a normalized RMSE of only 0.1228 and a Mean Absolute Percentage Error (MAPE) of 24.27%, this model is expected to accurately predict IC_{50} in malaria isolates. See Figure 1 for a visualization of the experiment runs and Figure 2 for the distribution of residuals on the best model.

Metric	Value
Normalized Root Mean Squared Error	0.1228
Root Mean Squared Log Error	0.1336
Normalized Mean Absolute Error	0.1097
Mean Absolute Percentage Error	24.27
Normalized Median Absolute Error	0.1097
Root Mean Squared Error	0.3398
Explained Variance	-1.755
Normalized Root Mean Squared Log Error	0.1379
Median Absolute Error	0.3035
Mean Absolute Error	0.3035

Table 5. Model metrics of the final IC_{50} ensemble model.

Iteration	Preprocessor	Algorithm	Normalized RMSE
498		VotingEnsemble	0.12283293
370	SparseNormalizer	RandomForest	0.132003138
432	StandardScalerWrapper	LightGBM	0.133180215
240	SparseNormalizer	RandomForest	0.133779391
430	StandardScalerWrapper	RandomForest	0.137084337
65	SparseNormalizer	RandomForest	0.13884791
56	SparseNormalizer	RandomForest	0.14417843
68	MaxAbsScaler	ExtremeRandomTrees	0.151925822
470	StandardScalerWrapper	RandomForest	0.152262231
181	MinMaxScaler	LightGBM	0.15279075

Table 6. Top 10 training iterations of the IC_{50} model search, evaluated by Root Mean Squared Error. Note that the top performing model (VotingEnsemble) is the final IC_{50} model discussed in this paper.

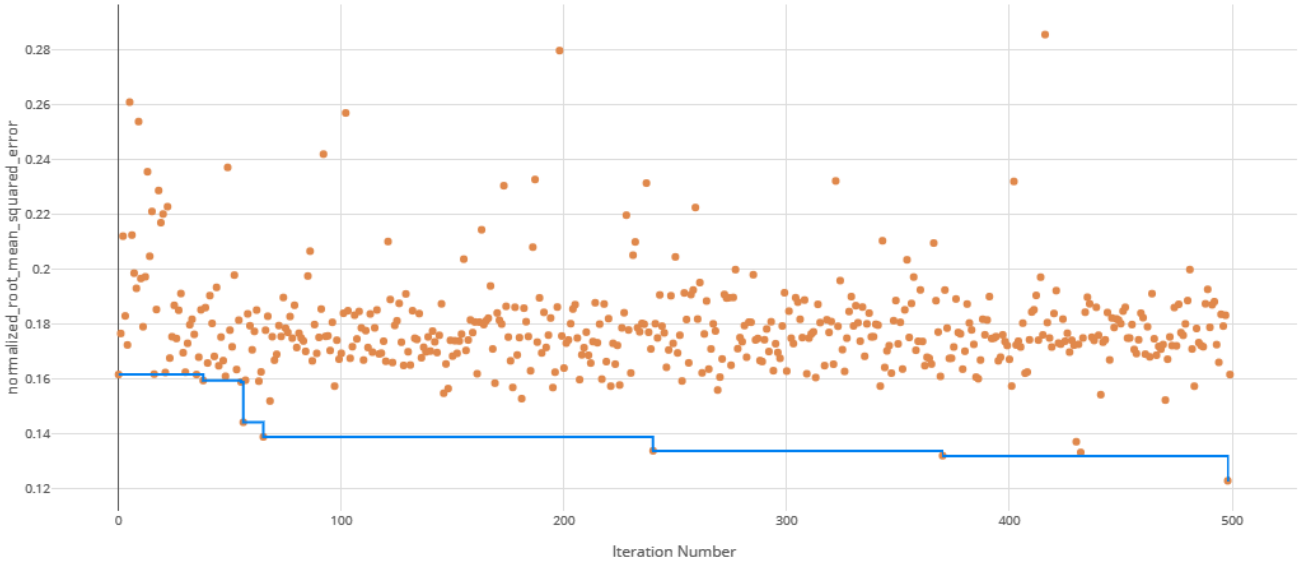

Figure 1. Root Mean Squared Error (RMSE) by iteration of the IC_{50} model search. Each orange dot is an iteration with the blue line representing the minimum RMSE up to that iteration.

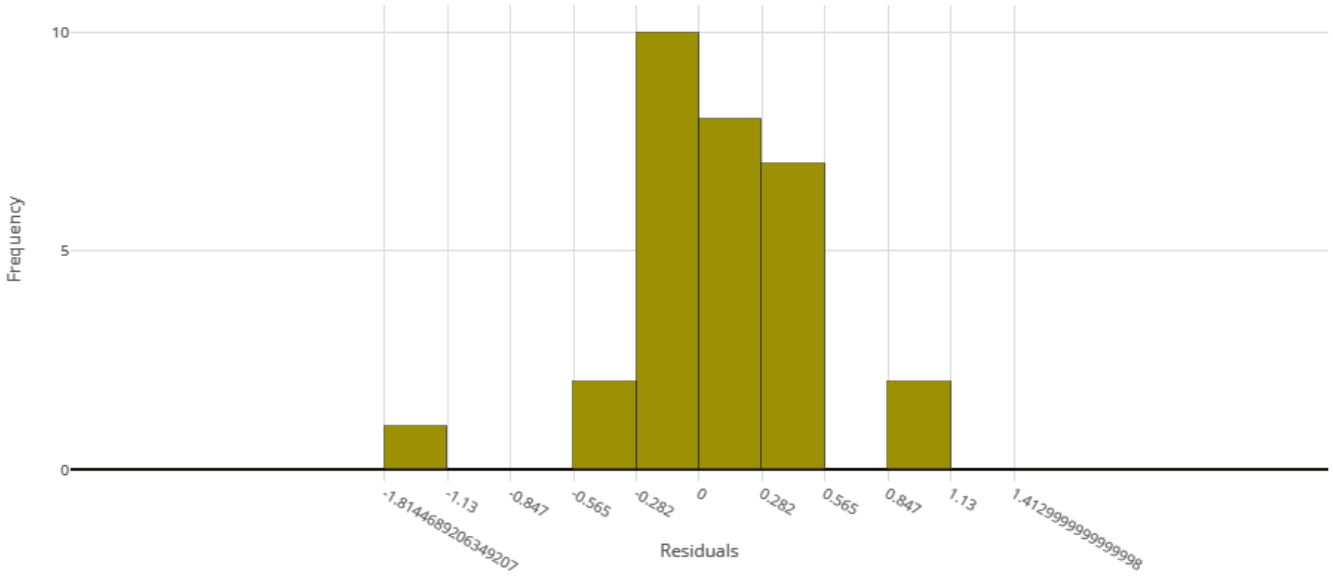

Figure 2. Model residuals of the final IC_{50} ensemble model.

Prediction of resistance status

The second task of this work was to create a machine learning model that can predict the parasite clearance rate (fast versus slow) of malaria isolates. When resistance rates change in a pathogen, it can be indicative of regulatory changes in the pathogen's genome. These changes can be exploited for the prevention of further resistance spread. Thus, a goal of this work is to understand genes important in the prediction of artemisinin resistance. The relationship of this use case to the first is that parasite clearance is a measure of the effectiveness of a treatment regimen. While the first use case looked at the drug concentration, this use case looks into the speed at which the parasites are cleared as a result of a standard treatment.

Methods

An *in vivo* transcription data set from Mok *et al.*, (2015) Science[28] was used to predict the parasite clearance rate of malaria parasite isolates based on *in vitro* transcriptional profiles (see Table 8).

The training data consists of 1,043 isolates with 4,952 genes from the malaria parasite *Plasmodium falciparum*. For each malaria parasite isolate, transcription data was collected for various *PF3D7* genes. The form of the training dataset contains 1,043 rows and 4,957 columns, as shown in Table 7. The feature to predict is `ClearanceRate`.

Sample_Names	Country	Asexual_stage__hpi_	Kmeans_Grp	PF3D7_0100100	...	PF3D7_1480100	ClearanceRate
GSM1427365	Bangladesh	20	B	0.226311	...	-0.64171	Fast
...	...	...	...	...	...	...	...
GSM1427537	Cambodia	12	C	0.81096	...	-1.72825	Slow
...	...	...	...	...	...	...	...
GSM1428407	Vietnam	8	A	0.999095	...	NaN	Fast

Table 7. Format of the clearance rate model training data.

Data preparation

The training data for this use case did not require the same pivoting transformations as in the last use case as each record describes a single isolate. Thus, only the vectorization of the data was necessary, which was performed using the Spark `VectorAssembler` and then converted into a Numpy-compatible array [22]. Note that this vectorization only kept the numerical columns, which excludes the `Country`, `Kmeans_Grp`, and `Asexual_stage__hpi_` attributes as they are either absent or contain non-matching factors (i.e. different set of countries) in the testing data.

	Training Set
Number of isolates	1043
Isolate collection site	Southeast Asia
Isolate collection years	2012-2014
Sample type	*in vivo*
Synchronized?	Not synchronized
Number of samples per isolate	1
Additional attributes	~18 hpi, Non-perturbed, No replicates

Table 8. Training dataset information from Mok *et al.*, 2015[28].

Machine learning

Once the 98 individual models were trained, two ensemble models (voting ensemble and stack ensemble) were then created and tested as before. Model search parameters are shown in Table 9.

Parameter	Value
Task	Regression
Number of iterations	100
Iteration timeout (minutes)	20
Max cores per iteration	14
Primary metric	weighted area under the receiver operating characteristic curve (AUC)
Preprocess data?	True
k-Fold cross-validations	10 folds

Table 9. Model search parameter settings for the clearance rate model search.

Results

The voting ensemble model (using soft voting) was selected as the best model, having the highest area under the receiver operating characteristic curve (AUC), as shown in Table 11. The top 10 of the 100 models trained are reported in Table 10. Having a weighted AUC of 0.87 and a weighted F1 score of 0.80, this model is expected to accurately predict isolate clearance rates. A confusion matrix of the predicted results versus actuals is shown in Table 12. See Figure 3 for a visualization of the experiment runs and see Figures 4 and 5 for the ROC and Precision-Recall curves on the best model. Note that these models were evaluated using random sampling of the input training dataset provided by the DREAM Challenge, though the evaluation within the challenge was performed on an unlabelled testing dataset. The metrics in the Results section below reflect the evaluation on the sampled training data.

Note that the averages reported in Figures 4 and 5 are defined as follows:

- 'micro': Computed globally by combining the true positives and false positives from each class at each cutoff.
- 'macro': The arithmetic mean for each class. This does not take class imbalance into account.
- 'weighted': The arithmetic mean of the score for each class, weighted by the number of true instances in each class (support).

Iteration	Preprocessor	Algorithm	Weighted AUC
98		VotingEnsemble	0.870471056
99		StackEnsemble	0.865215516
65	StandardScalerWrapper	LogisticRegression	0.86062304
33	StandardScalerWrapper	LogisticRegression	0.859881677
97	StandardScalerWrapper	LogisticRegression	0.858791006
44	StandardScalerWrapper	LogisticRegression	0.856105491
73	StandardScalerWrapper	LogisticRegression	0.855502817
17	RobustScaler	SVM	0.855452622
43	StandardScalerWrapper	LogisticRegression	0.855368394
61	RobustScaler	LogisticRegression	0.854357599

Table 10. Top 10 training iterations of the clearance rate model search.
Note that the top performing model (VotingEnsemble) is the clearance rate model discussed in this paper.

Metric	Accuracy
f1_score_macro	0.6084
AUC_micro	0.9445
AUC_macro	0.8475
recall_score_micro	0.8101
recall_score_weighted	0.8101
average_precision_score_weighted	0.8707
weighted_accuracy	0.8585
precision_score_macro	0.6217
precision_score_micro	0.8101
balanced_accuracy	0.6027
log_loss	0.4455
recall_score_macro	0.6027
precision_score_weighted	0.8
AUC_weighted	0.8705
average_precision_score_micro	0.8911
f1_score_weighted	0.8019
f1_score_micro	0.8101
norm_macro_recall	0.354
average_precision_score_macro	0.7344
accuracy	0.8101

Table 11. Model metrics of the final clearance rate ensemble model.

Feature importance

Feature importances were calculated using mimic-based model explanation of the ensemble model [29]. The mimic explainer works by training global surrogate models to mimic blackbox models (i.e. complex models that are difficult to explain). The surrogate model is an interpretable model, trained to approximate the predictions of a black box model as

	Class	Prediction		
		Fast (ID: 0)	Slow (ID: 1)	Null (ID: 2)
Actual	Fast (ID: 0)	661	74	0
	Slow (ID: 1)	115	184	0
	Null (ID: 2)	6	3	0

Table 12. Confusion matrix of clearance rate predictions versus actual.

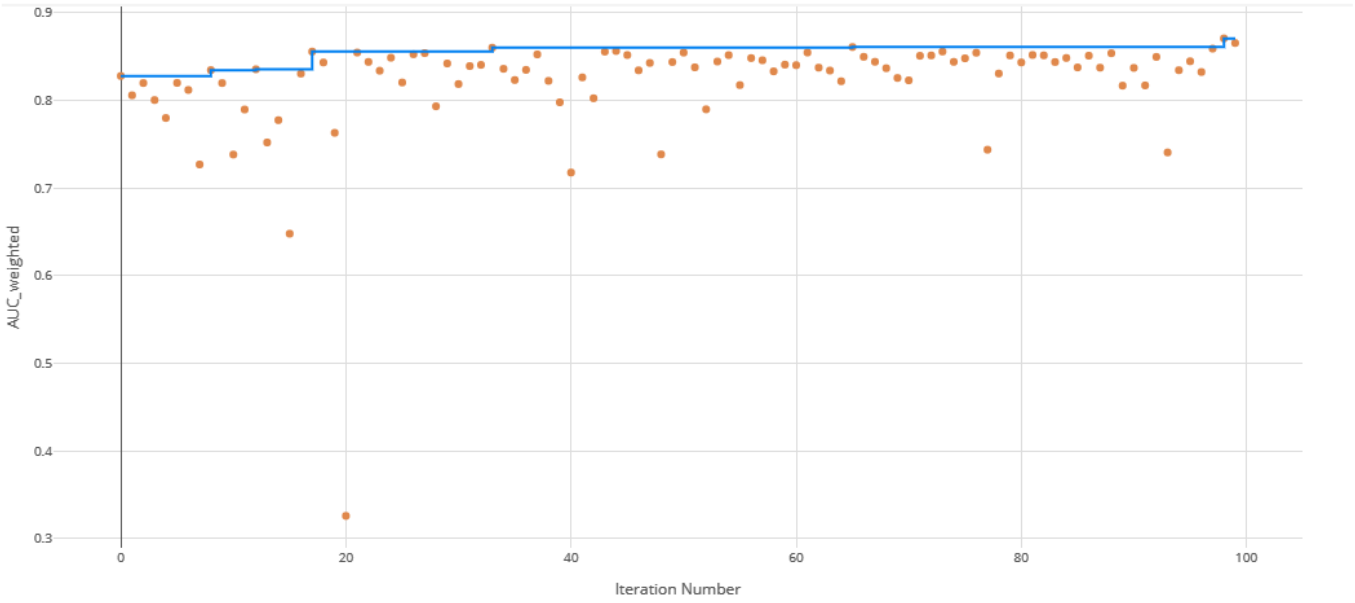

Figure 3. Area under the receiver operating characteristic curve (AUC) by iteration of the clearance rate model. Each orange dot is an iteration with the blue line representing the maximum AUC up to that iteration.

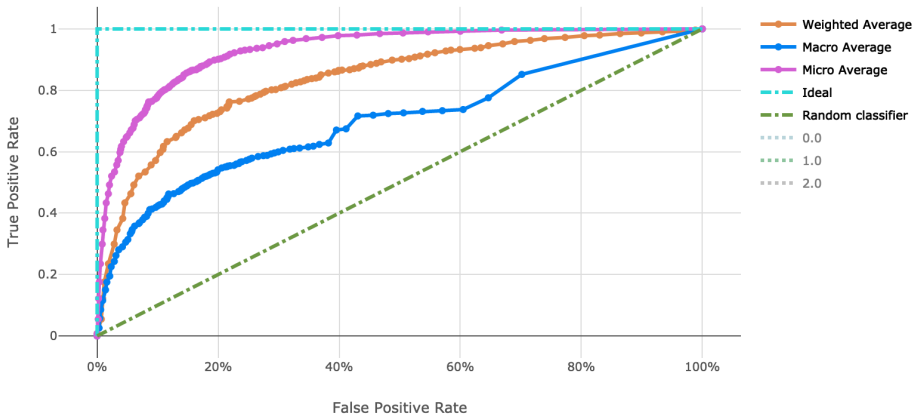

Figure 4. Receiver operating characteristic curve of the clearance rate model.

accurately as possible [30]. In Figure 6 and Table 13, the feature importance values for each class ("Slow", "Fast", and NULL) are shown. This shows which genes are important in the prediction of clearance rate.

The mimic explainer was opted over other traditional methods such as principal component analysis (PCA) because of its ability to provide clearer interpretations into the features' importance. PCA occludes the true values of individual features by summarising multiple features together. Given that insights into particular genes' importance on resistance were desired here, the mimic explainer provides this output in a more straightforward manner.

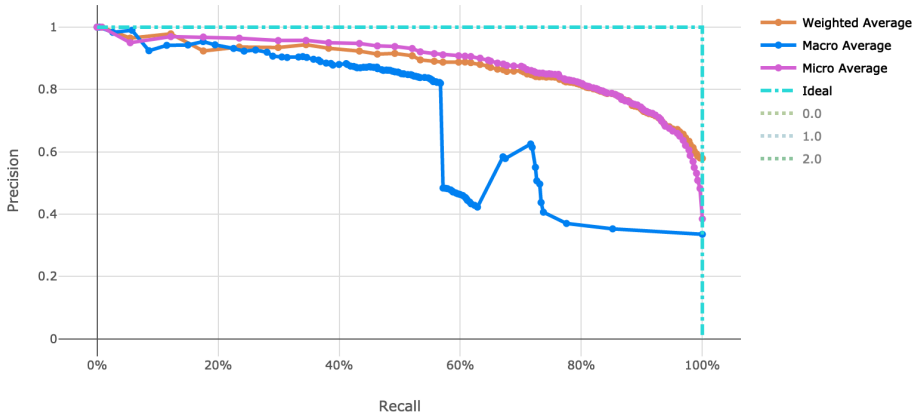

Figure 5. Precision-Recall curve of the clearance rate model.

Rank	PF3D7 Gene	Slow Importance	Fast Importance	NULL Importance	Overall Importance
1	PF3D7_1245300	0.292	0.118	0.000	0.410
2	PF3D7_1107700	0.020	0.274	0.000	0.294
3	PF3D7_1328400	0.154	0.123	0.000	0.277
4	PF3D7_1372000	0.172	0.095	0.000	0.267
5	PF3D7_1115600	0.083	0.179	0.000	0.262
6	PF3D7_0608100	0.000	0.000	0.243	0.243
7	PF3D7_0523000	0.154	0.087	0.000	0.241
8	PF3D7_1205300	0.000	0.002	0.197	0.199
9	PF3D7_1129100	0.008	0.191	0.000	0.199

Table 13. Top 10 PF3D7 genes (features) in predicting clearance rate.

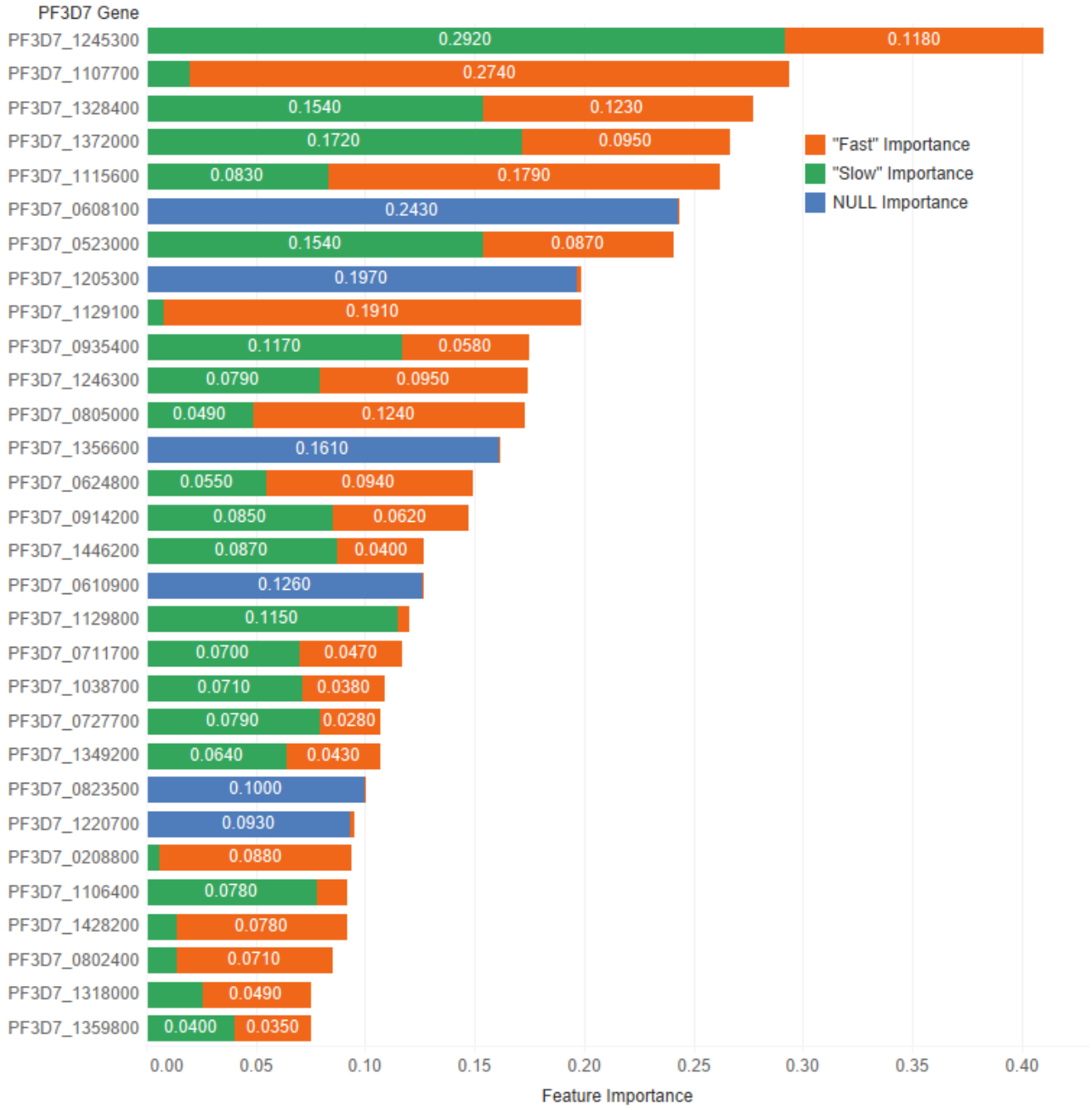

Figure 6. Derived feature importances using the black box mimic model explanation of the clearance rate model. (Shown: Top 30 genes.)

Discussion

By using distributed processing of the data preparation, we can successfully shape and manage large malaria datasets. We efficiently transformed a matrix of over 40,000 genetic attributes for the IC_{50} use case and over 4,000 genetic attributes for the resistance rate use case. This was completed with scalable vectorization of the training data, which allowed for many machine learning models to be generated. By tracking the individual performance results of each machine learning model, we can determine which model is most useful. In addition, ensemble modeling of the various singular models proved effective for both tasks in this work. While the number of training observations for each use case stand to be improved, the usage of adequate cross-validation can help to stabilize the risk of over fitting models to such a small dataset. Also note that there is an imbalance in the number of samples in each class in the clearance rate experiment, which stands to be remedied in future work. There are over double the number of “Fast" clearance rate isolates compared to “Slow". This can be seen in the variation in model performance as indicated by the macro average Precision-Recall curve (Figure 5).

The resulting model performance of both the IC_{50} model and the clearance rate model show relatively adequate fitting of the data for their respective predictions. While additional model tuning may provide a lift in model performance, we have demonstrated the utility of ensemble modeling in these predictive use cases in malaria. In both models, we shows that IC_{50} and clearance rate can be effectively predicted using transcriptomic analysis data with machine learning. By extension, this is also predicting the phenotypic result of the genetic variations among the samples as is relates to resistance.

In a broader sense for the field parasitology, this exercise helps to quantify the importance of genetic features, spotlighting potential genes that are significant in artemisinin resistance. The merit of this work showcases the utility of machine learning to assist in the understanding of the underlying genetic/transcriptomic mechanisms that affect drug performance.

Specific examples include PF3D7 1245300, the most important feature in predicting slow parasite clearance. PF3D7 1245300 is the gene that codes for the NEDD8-conjugating enzyme UBC12 (UniProt ID: Q8I4X8), a ligase used in the ubiquitin conjugating pathway. Another example, PF3D7 1107700 is the most important gene for fast clearance rate. PF3D7 1107700 (UniProt ID: Q8IIS5) is important in the regulation of the cell cycle, specifically in the maturation of ribosomal RNAs and in the formation of the large ribosomal subunit. Future *in vitro* experiments of this *in silico* work should be performed to validate these findings. While biological confirmations of these genetic factors are needed, this analysis helps to rank the most probable factors by importance, therefore reducing the *in vitro* work to be performed.

These two examples of important genes identified here along with the other may one day be the target for future drugs or may prove integral in the overall understanding of how resistance works in *P. falciparum*. The utility of these models will help in directing development of alternative treatments or coordination of combination therapies in resistant infections and provides an example of the usage of machine learning in the identification of important genetic feature in infectious disease research.

Scaling and Normalization	Description
StandardScaleWrapper	Standardize features by removing the mean and scaling to unit variance
MinMaxScalar	Transforms features by scaling each feature by that column’s minimum and maximum
MaxAbsScaler	Scale each feature by its maximum absolute value
RobustScalar	This Scaler features by their quantile range
PCA	Linear dimensionality reduction using singular value decomposition of the data to project it to a lower dimensional space
TruncatedSVDWrapper	This transformer performs linear dimensionality reduction by means of truncated singular value decomposition. Contrary to PCA, this estimator does not center the data before computing the singular value decomposition. This means it can efficiently work with sparse matrices.
SparseNormalizer	Each sample (each record of the data) with at least one non-zero component is re-scaled independently of other samples so that its norm (L1 or L2) equals one

Table 14. Scaling function information for machine learning model search [31].

Preprint

An earlier version of this article can be found on bioRxiv (doi:10.1101/856922).

Data availability

Underlying data

The challenge datasets are available from Synapse (`https://www.synapse.org/`; Synapse ID: syn18089524). Access to the data requires registration and agreement to the conditions for use at: `https://www.synapse.org/#!Synapse:syn18089524`.

Challenge documentation, including the detailed description of the Challenge design, data description, and overall results can be found at: `https://www.synapse.org/#!Synapse:syn16924919/wiki/583955`.

Whole genome expression profiling of artemsinin-resistant Plasmodium falciparum field isolates, Accession number GSE59099: `https://www.ncbi.nlm.nih.gov/geo/query/acc.cgi?acc=GSE59099`.

Zenodo: colbyford/malaria_DREAM2019: Ensemble Machine Learning Modeling for the Prediction of Artemisinin Resistance in Malaria - Initial Code Release for Research Publication (F1000). `https://doi.org/10.5281/zenodo.3590459`. [32]

This project contains the following underlying data:

- /SubChallenge1/data/sc1_X_train.pkl (Pickle file of the SubChallenge 1 independent variables, pivoted by `Timepoint`, `Treatment`, and `BioRep`.)
- /SubChallenge1/data/sc1_y_train.pkl (Pickle file of the SubChallenge 1 dependent variable, `DHA_IC50`.)
- /SubChallenge2/data/sc2_X_train.pkl (Pickle file of the SubChallenge 2 independent variables.)
- /SubChallenge2/data/sc2_y_train.pkl (Pickle file of the SubChallenge 2 dependent variable, `ClearanceRate`.)

Software availability

- Source code available from: `https://github.com/colbyford/malaria_DREAM2019`
- Archived source code at time of publication: `https://doi.org/10.5281/zenodo.3590459` [32]
- License: GPL-3.0

Competing interests

No competing interests were disclosed.

Grant information

This work was supported by the University of North Carolina at Charlotte Department of Bioinformatics and Genomics and the School of Data Science.

References

[1] Fact sheet about malaria. *World Health Organization*, Mar 2019. URL https://www.who.int/news-room/fact-sheets/detail/malaria.

[2] *Guidelines for the treatment of malaria*. World Health Organization, 2015.

[3] Arjen M. Dondorp, François Nosten, Poravuth Yi, Debashish Das, Aung Phae Phyo, Joel Tarning, Khin Maung Lwin, Frederic Ariey, Warunee Hanpithakpong, Sue J. Lee, Pascal Ringwald, Kamolrat Silamut, Mallika Imwong, Kesinee Chotivanich, Pharath Lim, Trent Herdman, Sen Sam An, Shunmay Yeung, Pratap Singhasivanon, Nicholas P.J. Day, Niklas Lindegardh, Duong Socheat, and Nicholas J. White. Artemisinin resistance in plasmodium falciparum malaria. *New England Journal of Medicine*, 361(5):455–467, 2009. doi: 10.1056/NEJMoa0808859. URL https://doi.org/10.1056/NEJMoa0808859. PMID: 19641202.

[4] Amed Ouattara, Aminatou Kone, Matthew Adams, Bakary Fofana, Amelia Walling Maiga, Shay Hampton, Drissa Coulibaly, Mahamadou A. Thera, Nouhoum Diallo, Antoine Dara, Issaka Sagara, Jose Pedro Gil, Anders Bjorkman, Shannon Takala-Harrison, Ogobara K. Doumbo, Christopher V. Plowe, and Abdoulaye A. Djimde. Polymorphisms in the k13-propeller gene in artemisinin-susceptible plasmodium falciparum parasites from bougoula-hameau and bandiagara, mali. *The American Journal of Tropical Medicine and Hygiene*, 92(6):1202–1206, 2015. ISSN 0002-9637. doi: https://doi.org/10.4269/ajtmh.14-0605. URL http://www.ajtmh.org/content/journals/10.4269/ajtmh.14-0605.

[5] Jakob Birnbaum, Sarah Scharf, Sabine Schmidt, Ernst Jonscher, Wieteke Anna Maria Hoeijmakers, Sven Flemming, Christa Geeke Toenhake, Marius Schmitt, Ricarda Sabitzki, Bärbel Bergmann, Ulrike Fröhlke, Paolo Mesén-Ramírez, Alexandra Blancke Soares, Hendrik Herrmann, Richárd Bártfai, and Tobias Spielmann. A kelch13-defined endocytosis pathway mediates artemisinin resistance in malaria parasites. *Science*, 367(6473):51–59, 2020. ISSN 0036-8075. doi: 10.1126/science.aax4735. URL https://science.sciencemag.org/content/367/6473/51.

[6] Sompob Saralamba, Wirichada Pan-Ngum, Richard J. Maude, Sue J. Lee, Joel Tarning, Niklas Lindegårdh, Kesinee Chotivanich, François Nosten, Nicholas P. J. Day, Duong Socheat, Nicholas J. White, Arjen M. Dondorp, and Lisa J. White. Intrahost modeling of artemisinin resistance in plasmodium falciparum. *Proceedings of the National Academy of Sciences*, 108(1):397–402, 2011. ISSN 0027-8424. doi: 10.1073/pnas.1006113108. URL https://www.pnas.org/content/108/1/397.

[7] Nicholas James White. The parasite clearance curve. In *Malaria Journal*, 2011.

[8] Elizabeth A. Ashley, Mehul Dhorda, Rick M. Fairhurst, Chanaki Amaratunga, Parath Lim, Seila Suon, Sokunthea Sreng, Jennifer M. Anderson, Sivanna Mao, Baramey Sam, Chantha Sopha, Char Meng Chuor, Chea Nguon, Siv Sovannaroth, Sasithon Pukrittayakamee, Podjanee Jittamala, Kesinee Chotivanich, Kitipumi Chutasmit, Chaiyaporn Suchatsoonthorn, Ratchadaporn Runcharoen, Tran Tinh Hien, Nguyen Thanh Thuy-Nhien, Ngo Viet Thanh, Nguyen Hoan Phu, Ye Htut, Kay-Thwe Han, Kyin Hla Aye, Olugbenga A. Mokuolu, Rasaq R. Olaosebikan, Olaleke O. Folaranmi, Mayfong Mayxay, Maniphone Khanthavong, Bouasy Hongvanthong, Paul N. Newton, Marie A. Onyamboko, Caterina I. Fanello, Antoinette K. Tshefu, Neelima Mishra, Neena Valecha, Aung Pyae Phyo, Francois Nosten, Poravuth Yi, Rupam Tripura, Steffen Borrmann, Mahfudh Bashraheil, Judy Peshu, M. Abul Faiz, Aniruddha Ghose, M. Amir Hossain, Rasheda Samad, M. Ridwanur Rahman, M. Mahtabuddin Hasan, Akhterul Islam, Olivo Miotto, Roberto Amato, Bronwyn MacInnis, Jim Stalker, Dominic P. Kwiatkowski, Zbynek Bozdech, Atthanee Jeeyapant, Phaik Yeong Cheah, Tharisara Sakulthaew, Jeremy Chalk, Benjamas Intharabut, Kamolrat Silamut, Sue J. Lee, Benchawan Vihokhern, Chanon Kunasol, Mallika Imwong, Joel Tarning, Walter J. Taylor, Shunmay Yeung, Charles J. Woodrow, Jennifer A. Flegg, Debashish Das, Jeffery Smith, Meera Venkatesan, Christopher V. Plowe, Kasia Stepniewska, Philippe J. Guerin, Arjen M. Dondorp, Nicholas P. Day, and Nicholas J. White. Spread of artemisinin resistance in plasmodium falciparum malaria. *New England Journal of Medicine*, 371(5):411–423, 2014. doi: 10.1056/NEJMoa1314981. URL https://doi.org/10.1056/NEJMoa1314981. PMID: 25075834.

[9] Douglas B Kell, Robert M Darby, and John Draper. Genomic computing. explanatory analysis of plant expression profiling data using machine learning. *Plant Physiology*, 126(3):943–951, 2001.

[10] Amanda Clare. *Machine learning and data mining for yeast functional genomics*. PhD thesis, University of Wales, Aberystwyth, 2003.

[11] Sangkyu Lee, Sarah Kerns, Harry Ostrer, Barry Rosenstein, Joseph O Deasy, and Jung Hun Oh. Machine learning on a genome-wide association study to predict late genitourinary toxicity after prostate radiation therapy. *International Journal of Radiation Oncology* Biology* Physics*, 101(1):128–135, 2018.

[12] Dmitry Grapov, Johannes Fahrmann, Kwanjeera Wanichthanarak, and Sakda Khoomrung. Rise of deep learning for genomic, proteomic, and metabolomic data integration in precision medicine. *Omics: a journal of integrative biology*, 22(10):630–636, 2018.

[13] Sean Ekins, Jair Lage de Siqueira-Neto, Laura-Isobel McCall, Malabika Sarker, Maneesh Yadav, Elizabeth L Ponder, E Adam Kallel, Danielle Kellar, Steven Chen, Michelle Arkin, et al. Machine learning models and pathway genome data base for trypanosoma cruzi drug discovery. *PLoS neglected tropical diseases*, 9(6), 2015.

[14] Rui Kuang, Jianying Gu, Hong Cai, and Yufeng Wang. Improved prediction of malaria degradomes by supervised learning with svm and profile kernel. *Genetica*, 136(1):189–209, 2009.

[15] Dev Kumar Das, Madhumala Ghosh, Mallika Pal, Asok K Maiti, and Chandan Chakraborty. Machine learning approach for automated screening of malaria parasite using light microscopic images. *Micron*, 45:97–106, 2013.

[16] Zhaohui Liang, Andrew Powell, Ilker Ersoy, Mahdieh Poostchi, Kamolrat Silamut, Kannappan Palaniappan, Peng Guo, Md Amir Hossain, Antani Sameer, Richard James Maude, et al. Cnn-based image analysis for malaria diagnosis. In *2016 IEEE International Conference on Bioinformatics and Biomedicine (BIBM)*, pages 493–496. IEEE, 2016.

[17] Mahdieh Poostchi, Kamolrat Silamut, Richard J Maude, Stefan Jaeger, and George Thoma. Image analysis and machine learning for detecting malaria. *Translational Research*, 194:36–55, 2018.

[18] Sage Davis, Katrina Button-Simons, Taoufik Bensellak, Eren Mehmet Ahsen, Lisa Checkley, Gabriel J. Foster, Xinzhuan Su, Ahmed Moussa, Darlington Mapiye, Sok Kean Khoo, Francois Nosten, Timothy J. C. Anderson, Katelyn Vendrely, Julie Bletz, Thomas Yu, Sumir Panji, Amel Ghouila, Nicola Mulder, Thea Norman, Steven Kern, Pablo Meyer, Gustavo Stolovitzky, Michael T. Ferdig, and Geoffrey H. Siwo. Leveraging crowdsourcing to accelerate global health solutions. *Nature Biotechnology*, 37(8):848–850, 2019. ISSN 1546-1696. doi: 10.1038/s41587-019-0180-5. URL `https://doi.org/10.1038/s41587-019-0180-5`.

[19] A. Ghouila, G. H. Siwo, J. D. Entfellner, S. Panji, K. A. Button-Simons, S. Z. Davis, F. M. Fadlelmola, M. T. Ferdig, N. Mulder, T. Bensellak, A. Ghansah, K. Ghedira, A. Gritzman, I. Isewon, A. Kishk, A. Moussa, C. Loucoubar, P. Musicha, M. Pore, D. M. Sengeh, D. S. Mapiye, P. K. Rallabandi, and M. Varughese. Hackathons as a means of accelerating scientific discoveries and knowledge transfer. *Genome Res.*, 28(5):759–765, 05 2018.

[20] Lindsey B. Turnbull, Geoffrey H. Siwo, Katrina A. Button-Simons, Asako Tan, Lisa A. Checkley, Heather J. Painter, Manuel Llinás, and Michael T. Ferdig. Simultaneous genome-wide gene expression and transcript isoform profiling in the human malaria parasite. *PLOS ONE*, 12(11):1–20, 11 2017. doi: 10.1371/journal.pone.0187595. URL `https://doi.org/10.1371/journal.pone.0187595`.

[21] Matei Zaharia, Reynold S. Xin, Patrick Wendell, Tathagata Das, Michael Armbrust, Ankur Dave, Xiangrui Meng, Josh Rosen, Shivaram Venkataraman, Michael J. Franklin, Ali Ghodsi, Joseph Gonzalez, Scott Shenker, and Ion Stoica. Apache spark: A unified engine for big data processing. *Commun. ACM*, 59(11):56–65, October 2016. ISSN 0001-0782. doi: 10.1145/2934664. URL `http://doi.acm.org/10.1145/2934664`.

[22] Stéfan van der Walt, S. Chris Colbert, and Gaël Varoquaux. The numpy array: A structure for efficient numerical computation. *Computing in Science & Engineering*, 13(2):22–30, 2011. doi: 10.1109/MCSE.2011.37. URL `https://aip.scitation.org/doi/abs/10.1109/MCSE.2011.37`.

[23] Microsoft Azure Machine Learning Service, 2019. URL `https://azure.microsoft.com/en-us/services/machine-learning/`.

[24] Microsoft. *Azure Machine Learning AutoML Core version 1.0.79*. 2019. URL `https://pypi.org/project/azureml-automl-core/`.

[25] F. Pedregosa, G. Varoquaux, A. Gramfort, V. Michel, B. Thirion, O. Grisel, M. Blondel, P. Prettenhofer, R. Weiss, V. Dubourg, J. Vanderplas, A. Passos, D. Cournapeau, M. Brucher, M. Perrot, and E. Duchesnay. Scikit-learn: Machine learning in Python. *Journal of Machine Learning Research*, 12:2825–2830, 2011.

[26] Guolin Ke, Qi Meng, Thomas Finley, Taifeng Wang, Wei Chen, Weidong Ma, Qiwei Ye, and Tie-Yan Liu. Lightgbm: A highly efficient gradient boosting decision tree. In I. Guyon, U. V. Luxburg, S. Bengio, H. Wallach, R. Fergus, S. Vishwanathan, and R. Garnett, editors, *Advances in Neural Information Processing Systems 30*, pages 3146–3154. Curran Associates, Inc., 2017. URL `http://papers.nips.cc/paper/6907-lightgbm-a-highly-efficient-gradient-boosting-decision-tree.pdf`.

[27] Rich Caruana, Alexandru Niculescu-Mizil, Geoff Crew, and Alex Ksikes. Ensemble selection from libraries of models. In *Proceedings of the Twenty-first International Conference on Machine Learning*, ICML '04, pages 18–, New York, NY, USA, 2004. ACM. ISBN 1-58113-838-5. doi: 10.1145/1015330.1015432. URL `http://doi.acm.org/10.1145/1015330.1015432`.

[28] Sachel Mok, Elizabeth A. Ashley, Pedro E. Ferreira, Lei Zhu, Zhaoting Lin, Tomas Yeo, Kesinee Chotivanich, Mallika Imwong, Sasithon Pukrittayakamee, Mehul Dhorda, Chea Nguon, Pharath Lim, Chanaki Amaratunga, Seila Suon, Tran Tinh Hien, Ye Htut, M. Abul Faiz, Marie A. Onyamboko, Mayfong Mayxay, Paul N. Newton, Rupam Tripura, Charles J. Woodrow, Olivo Miotto, Dominic P. Kwiatkowski, François Nosten, Nicholas P. J. Day, Peter R. Preiser, Nicholas J. White, Arjen M. Dondorp, Rick M. Fairhurst, and Zbynek Bozdech. Population transcriptomics of human malaria parasites reveals the mechanism of artemisinin resistance. *Science*, 347(6220):431–435, 2015. ISSN 0036-8075. doi: 10.1126/science.1260403. URL `https://science.sciencemag.org/content/347/6220/431`.

[29] Scott M Lundberg and Su-In Lee. A unified approach to interpreting model predictions. In I. Guyon, U. V. Luxburg, S. Bengio, H. Wallach, R. Fergus, S. Vishwanathan, and R. Garnett, editors, *Advances in Neural Information Processing Systems 30*, pages 4765–4774. Curran Associates, Inc., 2017. URL `http://papers.nips.cc/paper/7062-a-unified-approach-to-interpreting-model-predictions.pdf`.

[30] Christoph Molnar. *Interpretable Machine Learning*. 2019. https://christophm.github.io/interpretable-ml-book/.

[31] Microsoft. Microsoft Azure Machine Learning - AutoML Preprocessing, Nov 2019. URL https://docs.microsoft.com/en-us/azure/machine-learning/concept-automated-ml#automatic-preprocessing-standard.

[32] Colby Ford. colbyford/malaria_DREAM2019: Ensemble Machine Learning Modeling for the Prediction of Artemisinin Resistance in Malaria - Initial Code Release for Research Publication (F1000), December 2019. URL https://doi.org/10.5281/zenodo.3590459.