An Improved Dataset for Predicting Mammal Infecting Viruses from Genetic Sequence Information

Tyler Reddy[1*], Austin Schneider[2], Aaron R Hall[1], Adam Witmer[1], Nick Hengartner[3*]

[1]CAI-1: Applied Computer Science, Los Alamos National Laboratory, Los Alamos, New Mexico.
[2]P-2: Applied And Fundamental Physics, Los Alamos National Laboratory, Los Alamos, New Mexico.
[3]T-6: Theoretical Biology and Biophysics, Los Alamos National Laboratory, Los Alamos, New Mexico.

*Corresponding author(s). E-mail(s): treddy@lanl.gov; nickh@lanl.gov;

Abstract

There have been several attempts to develop machine learning (ML) models to identify human infecting viruses from their genomic sequences, with varying degrees of success. Direct comparison between models is problematic, because these models are typically trained and evaluated on different datasets having different baseline, outcome, and features. In this paper we present a standardized dataset of mammal infecting and non-infecting viral pathogens, refined from the previous work of Mollentze *et al.* to include the latest literature evidence and new host target labels, primate and mammal. On this dataset, we report the performance of eight machine learning models for predicting mammal-infecting viruses from their genomic sequences. We find that randomly assigning cases in our improved dataset to training/testing sets, when compared to the original assignments into training/testing in Mollentze *et al.*, increases the overall average ROC AUC of prediction of human infection from **0.663±0.070** to **0.784±0.013**, while the broadest host category of mammal infection can be predicted most reliably at **0.850 ± 0.020**. We share our improved dataset and code to enable standardized comparisons of machine learning methods to predict human host infections.

Keywords: zoonotic viruses, machine learning, bioinformatics

1 Introduction

Accurate prediction of viral spillover events and emerging threats is crucial to the development of early warning systems to mitigate future pandemics. Such systems are needed because pandemics from spillover events occur regularly. The Spanish flu (1918) – birds, swine flu (H1N1, 2009) – pigs, Zika virus outbreak (2015) – monkeys, and the recent SARS-CoV-2 (2019) – bats, are a few notable examples [1]. Early detection of spillover events is challenging due to the vast diversity of possible zoonotic pathogens that need to be surveilled. The recent efforts to rank pathogens at highest risk of initiating a viral spillover event, including machine-learning models, [2–5] can partially mitigate this challenge. Viral genomic data is generally the most readily available data, especially for newly discovered viruses, whereas other biological descriptors (features), such as protein structures, require a greater investment of time and resources [6, 7]. Therefore, machine learning (ML) models trained solely on nucleotide sequences offer a quick and cost-effective method to triage emerging viruses to identify those of greatest concern to human health.

Many ML classifiers have been developed to identify human infecting viruses from genetic sequence information. They differ significantly with regards to how, and what data, are used, and, how the ensuing results are reported (see Table 1). That heterogeneity and ensuing wide range of ML predictive performances, highlight the need for a benchmark dataset to enable model comparison. Indeed, ML researchers advocate developing ML methods using standardized datasets with agreed upon metrics and reporting standards [8, 9].

Several factors should be weighted when designing a standardized reference dataset, given that variation in data selection and reported metrics makes it difficult to compare models. For example, some models are fitted to large datasets composed of multiple isolates of each viral species [10, 11]; even though only around 300 distinct viral species are known to infect humans [12, 13]. Other models use a single reference sequence per species, leading to extremely small datasets [2, 14]. While having more isolates can increase the sample size and has the potential to reveal more features, it also risks introducing near duplicates in the test set of data in the training set, thereby artificially increasing the performance metrics [15]. On the other hand, using only reference sequences produces smaller datasets that limit the modeling techniques available to improve performance, and for which the fitted models may miss key features of zoonotic emergence [16]. ML methods risk to overfit in either case: in large datasets because of over-representation of frequently sequenced viruses, and in small datasets because noise in the data is not averaged out. The disparate model evaluation metrics and data splitting protocols used by the community, summarized in Table 1, reflect different priorities and can vary significantly across datasets, especially in class-imbalanced settings. They can be difficult to compare directly [17–19]. Without consistency in datasets, data splits, and reporting strategies, direct comparisons between models cannot be made, or are misleading [16, 20].

With this paper, we release an improved version of the dataset presented in Mollenzte *et al.* [2] by updating the assigned human infection potential and adding both potential mammal and primate infection labels. Using a train-test split scheme based on the train-holdout sets in [2], we evaluate model performance on the improved

Model Type	Dataset (Source)	% Human Infecting	Metric Type	Metric Value	Data Split	Citation
k-nearest neighbors	9,428 viral genomes (Virus-Host DB)	13.1	ROC AUC	0.92	cross-validated	[10]
deep neural network	9,496 viral genomes	13.7	balanced accuracy	91.7	80:10:10 train: test: validation	[21]
k-nearest neighbors	9,496 viral genomes	13.7	balanced accuracy	82.8	80:10:10 train: test: validation	[11]
support vector machine	158 viral genomes (Virus-Host DB)	50	ROC AUC	0.94	80:20 train: test split	[14]
gradient boosting	861 viral genomes	30.2	ROC AUC	0.77 (1), 0.73 (2)	(1) 70:15:15 train: calibration: test split over 100 iterations; (2) 0.73 on separate holdout dataset of 758 viruses (14.9 % human infecting)	[2]
Random Forest, Extra Trees, gradient boosting, SVM	849 viral genomes	33.0 (human) 36.2 (primate) 73.5 (mammal)	ROC AUC	0.784 ± 0.013 (human); 0.774 ± 0.015 (primate); 0.850 ± 0.020 (mammal)	holdout dataset of 736 viruses (19.0 % human infecting; 27.6 % primate infecting; 75.8 % mammal infecting)	This Work

Table 1 Heterogeneity of previously-described ML models that predict the likelihood of human infection based on viral genetic information. Our approach is shown in the bottom row, and cell colors are used to emphasize the similarities and differences between approaches.

dataset for common ML algorithms: random forest [22], extra-trees [22], gradient-boosted decision trees [23, 24], and support vector machines [22] (Figure 4). We found the original data split had relatively poor performance, possibly due to imbalanced viral family representation, and demonstrate that rebalancing the training and testing sets improves performance (sign test applied to 10 pairs of ROC AUC values 8 times

— 8 ML methods with 10 random seeds — yields both a minimum and maximum p-value $p = 1.95 \times 10^{-3}$) when predicting human pathogens (Figure 1). Furthermore, we see an expected increase in model performance when predicting mammal pathogens relative to primate (sign test, 10 pairs of ROC AUC values, 8 ML models, min and max p-value of $p = 1.95 \times 10^{-3}$) and human (sign test, same conditions, min and max p-value of $p = 1.95 \times 10^{-3}$) infecting pathogens. This improved dataset is a first step towards establishing a standard for determining human infectivity from genetic sequence data, and we suggest additional improvements that can be pursued towards this goal (see Discussion).

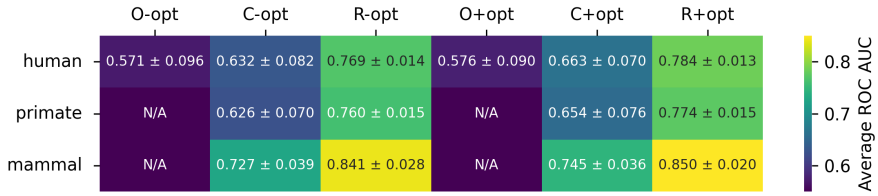


Fig. 1 Average estimator performances on test (ROC AUC across 10 seeds, with standard deviations) on different datasets in this work. Estimators averaged include Random Forest, Extra Trees, gradient boosted trees, and support vector machines. The datasets include the original (O) from Mollentze, a corrected version (C) of the Mollentze dataset, and a rebalanced (R) version of the corrected dataset. Results are further divided by optimization of hyperparameters (+) or lack thereof (-).

2 Results

2.1 Improvements to Mollentze et al. Dataset

Careful review of the original datasets from [2] (5th row in Table 1) revealed several inaccuracies. For example, twelve accession numbers (unique identifiers) corresponding to four viruses were present in both the train and test datasets, which we removed from the test dataset but kept in the train dataset. Additionally, we removed viral genomes from either dataset if any of their accessions contained the keyword "partial" in their genbank file on NCBI, indicating an incomplete genome. Finally, viral genomes that contained no coding sequences or only coding sequences not divisible by 3 were also removed, as amino acid based features could not be calculated for such coding sequences in [2] (see Discussion). The resulting corrected dataset now contains 849 (out of 861) genomes in the train set and 736 (out of 758) in the test set.

Since the holdout dataset used in [2] did not include review of the human infectivity of the viruses not isolated from humans, and because the knowledge of human infectivity of many viruses may have improved, we manually relabeled the viral dataset based on current literature (July 2025). While performing this review, we also determined primate and mammal infectivity for all viral species from the literature, as classifying against these targets may be useful for determining zoonotic emergence [25]. The updated abundance of human infecting viruses and the new abundances of primate

and mammal-infecting viruses are summarized in Figure 2A and Table 1. In particular, we found 2.8 % (train) and 4.1 % (test) increases in human infectivity relative to the original work.

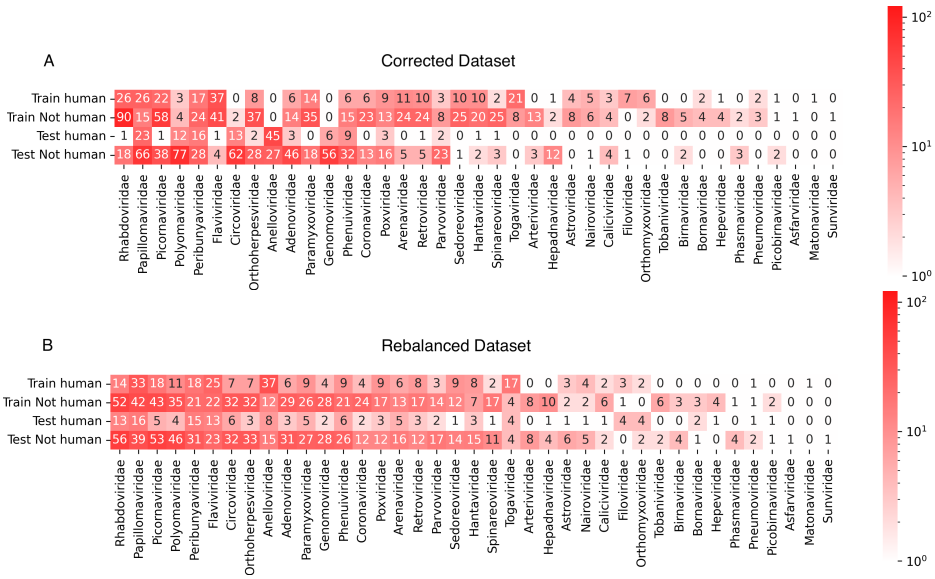


Fig. 2 Distribution of viral genomes in the datasets used in this work, categorized by human infectivity and training and test data split. The datasets are improved versions of the original dataset reported by Mollentze et al. [2], with specific improvements including removal of problematic genomes and updating known human infectivity (A), or additionally rebalancing the datasets by random shuffling with preservation of human infectivity ratios (B).

We also noticed that neither training nor test datasets had broad coverage over all viral families. This was a choice made by the authors as discussed in [2] and may account for their better performance observed on training than on the holdout set [2] (ROC AUC 0.77 vs 0.73, respectively). Although this may probe generalizability to a novel virus of unknown origin, a more likely scenario of zoonotic emergence would involve viruses in viral families known to contain species with the ability to infect humans or related vertebrates [26]. This suggests constraining training and testing sets to have similar representation across viral families. To achieve this, we reshuffled the train and holdout sets for each target host (human, primate, and mammal). The shuffle was unbiased with respect to viral family and preserved the total genomes and target label counts of the original datasets for comparison (Figure 2B). The additively smoothed ($\epsilon = 1 \times 10^{-9}$) Kullback-Leibler divergence (relative entropy) in viral family representation across training and test sets improved from 3.00 to 0.08 as a result of the shuffling, for human hosts. For primate and mammal host targets, the shuffling-based improvements in relative entropy were both 3.00 to 0.07.

2.2 Performance of ML Models on Dataset

Using our open source ML workflow available at https://github.com/lanl/ldrd_virus_work, we evaluated the following models on our dataset: random forest [22], extra trees [22], XGBoost [24], lightGBM (with both the boost and DART algorithms) [23, 27], and support vector machines using the linear, polynomial, and radial basis function kernels [22]. Models were trained on the training set and their performance evaluated on the test set. It is notable that our ML workflow leverages many of the same genomic data features previously reported by Mollentze *et al.* [2], in addition to leveraging overlapping peptide kmers translated from the viral genomes. The following sections present two major findings on how data splits impact classification performance.

2.2.1 Rebalancing Dataset for Viral Family Representation Improves Classification Performance

We observed that the models perform better (sign test, 10 pairs of ROC AUC values, 8 ML models, min and max $p = 1.95 \times 10^{-3}$) on the task of predicting viral human infectivity on the rebalanced (shuffled) dataset and have more similar ROC curves (Figure 3, Figure 4, Figure 1; mean ROC AUC 0.784 ± 0.013 across all models) and that the corrected dataset yields a slightly higher average ROC AUC (sign test, 10 pairs of ROC AUC values, 8 ML models, min and max $p = 1.95 \times 10^{-3}$) on the test set (mean ROC AUC 0.663 ± 0.070 across all models) than the original dataset (0.576 ± 0.090 across all models). Our XGBoost model underperforms the same estimator used by [2] on the original test set (0.657 ± 0.000; Figure 5, compared to 0.73) likely due to differences in our workflows (see Methods), one of which is the addition of peptide kmers in our array of features. Indeed, none of our estimators outperformed the original test set performance described by Mollentze *et al.* [2] (ROC AUC of 0.73), suggesting that the addition of peptide kmers to the array of features did not improve model performance on its own (Figure 5). Nonetheless, when combined with dataset rebalancing, our average estimator performance for prediction of viral human infectivity (0.784 ± 0.013) was greater than the originally published value (0.73).

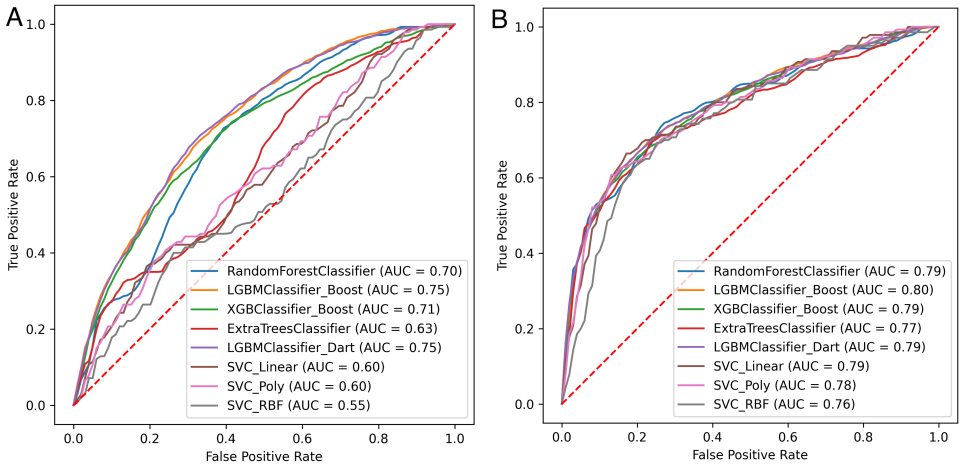
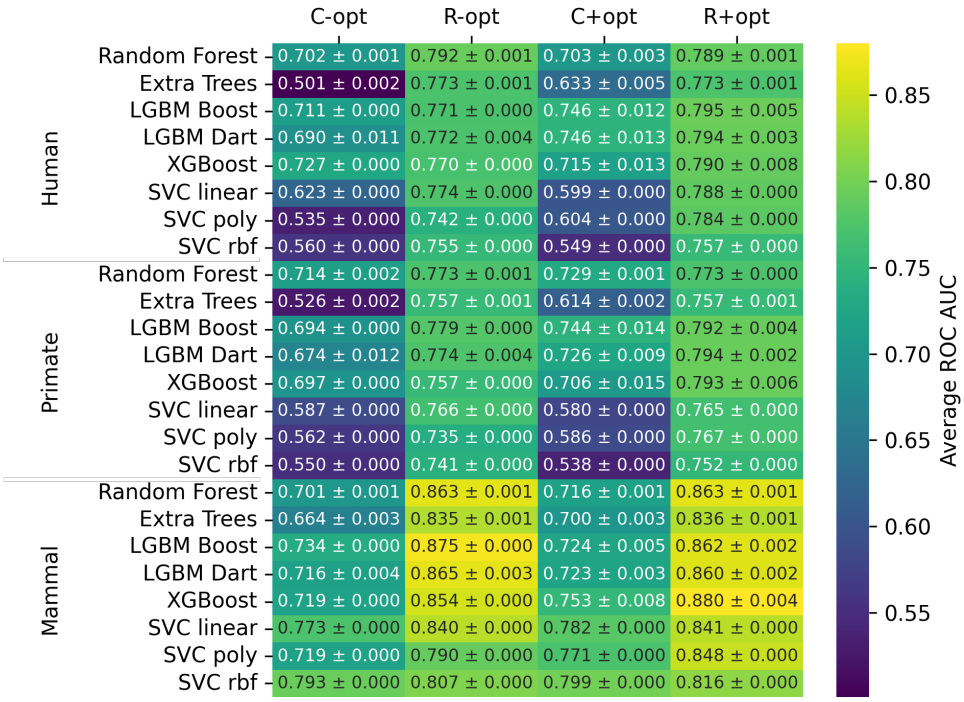


Fig. 3 The performance of common ML models was evaluated on the corrected (A) and rebalanced (B) datasets for prediction of viral human infectivity. Features calculated using our workflow were similar to those used by Mollentze et al. [2], except that we additionally included peptide kmers. Hyperparameter optimization was performed for each model before training (see Methods). Mean ROC curves were calculated from prediction scores on the test data across 10 random seeds, with ROC values shown in the legends and standard deviations (SD) summarized in Figure 4. The dashed red line represents an estimator that is no better than random chance.

Fig. 4 Mean ROC AUC on test broken down by estimator, dataset (C=corrected; R=rebalanced), hyperparameter optimization status (- = non-optimized; + = optimized), and host target. Results are averages and standard deviations across ten random seeds.



2.2.2 Changing Host Target Reveals Insights About Model Generalizability

To determine the ability of the model to generalize across broader host categories, we evaluated model performance for predicting primate and mammal infection (Figure 6). On the corrected dataset, the primate infection prediction achieved a mean ROC AUC of 0.654 ± 0.076. This performance is arguably distinguishable from the results for the human target on the same dataset (Figure 1; sign test, 10 pairs of ROC AUC values, 8 ML models, min and max p-values of $p = 1.95 \times 10^{-3}$ and 7.54×10^{-1}, respectively). On the rebalanced dataset, the primate target achieved a mean ROC AUC of 0.774 ± 0.015, again greater than the performance achieved for predicting human pathogens, but still with a substantial range of confidence between estimator types (sign test, 10 pairs of ROC AUC values, 8 ML models, min and max p-values of $p = 1.95 \times 10^{-3}$ and $p = 1.0$). ML predictions of mammal infecting viruses achieve a mean ROC AUC of 0.745 ± 0.036 on the corrected dataset, which is better than model predictions of human pathogens (sign test, 10 pairs of ROC AUC values, 8 ML models, min and max p-values of $p = 1.95 \times 10^{-3}$ and $p = 2.15 \times 10^{-2}$, respectively) and primate pathogens (sign test, 10 pairs of ROC AUC values, 8 ML models, min

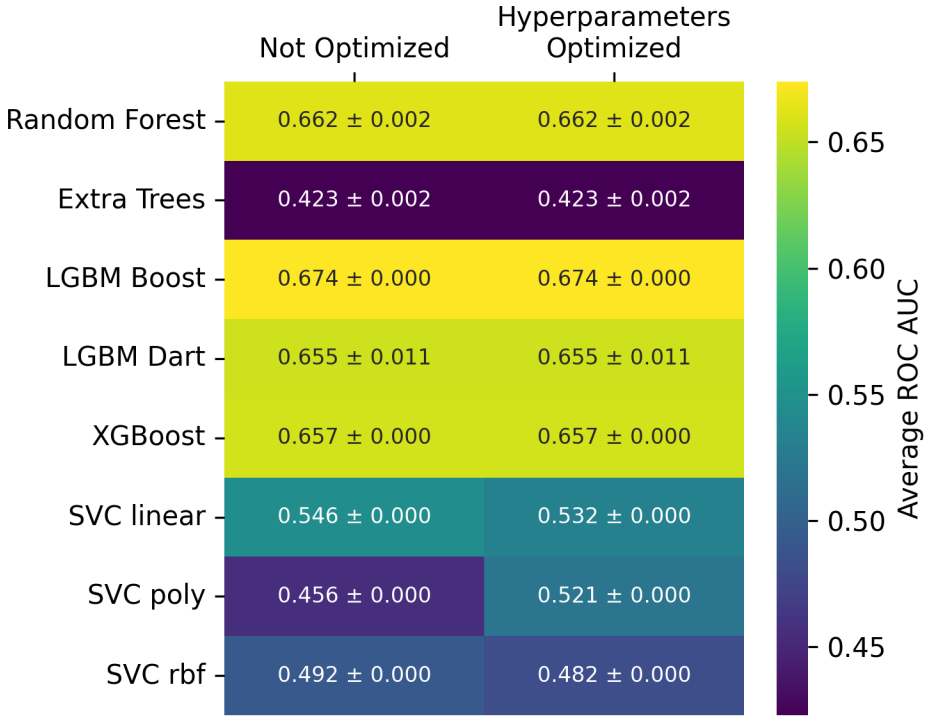


Fig. 5 Mean ROC AUC on test broken down by estimator and hyperparameter optimization status. These results are for a dataset closely matched to the one originally leveraged by Mollentze *et al.*, which only has human target labels. Results are averages and standard deviations across ten random seeds.

and max p-values of $p = 1.95 \times 10^{-3}$ and $p = 7.54 \times 10^{-1}$, respectively). On the rebalanced dataset, predictions for the mammal target achieved a mean ROC AUC of 0.850 ± 0.020, which is more confidently higher than predictions for human (sign test, 10 pairs of ROC AUC values, 8 ML models, min and max p-value of $p = 1.95 \times 10^{-3}$) and primate (sign test, 10 pairs of ROC AUC values, 8 ML models, min and max p-value of $p = 1.95 \times 10^{-3}$) pathogens. Taken together, these results indicate that models trained on a balanced dataset have better performance and capture important features for a variety of host categories, but some of the general trends are still preserved even without shuffling, albeit with lower confidence.

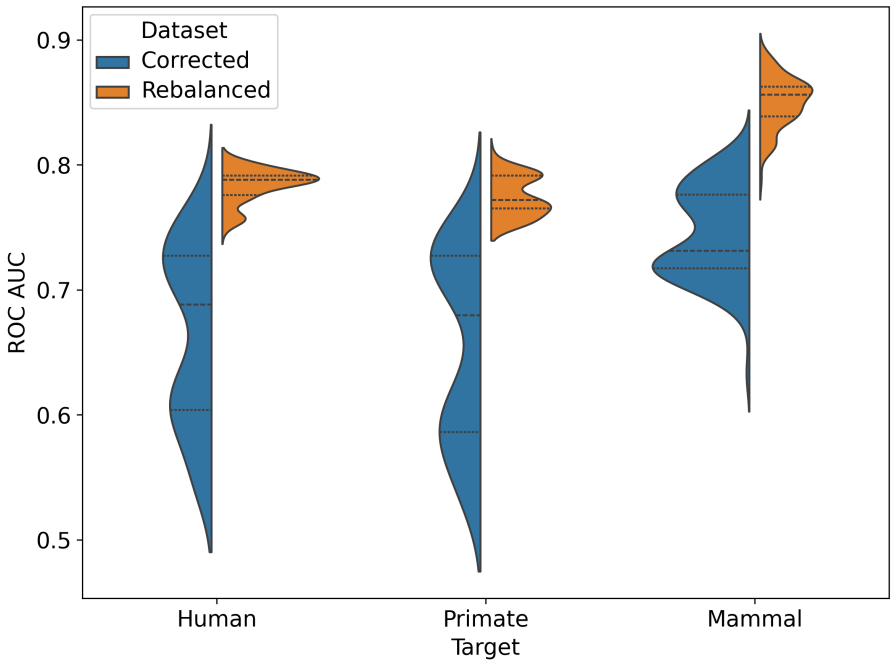


Fig. 6 The performance of common machine learning models was evaluated on the corrected and rebalanced datasets for human, primate, and mammal host targets. The ROC AUC for each model is reported for the predictions on test across 10 seeds. The violin plot displays the statistical distributions for ROC AUC across estimators, with dashed lines for the three quartiles–the 25th percentile, the median, and the 75th percentile.

3 Discussion

The need for classifiers to predict human infecting viruses from their genomic sequences alone is well motivated, yet progress towards their development has been slowed by inconsistencies in the datasets used, the metrics reported, and the data splitting practices [16, 20] (Table 1). We showed that shuffling the viral genome datasets, to improve their balance of viral families between training and test data, tends to improve ML model performance for human infection prediction (sign test applied to 10 pairs (seeds) of ROC AUC values, 8 ML models, minimum and maximum p-value $p = 1.95 \times 10^{-3}$). The datasets we share with the community here also include quality filtering sequences to avoid incomplete genomes, updating viral host infectivity to correct erroneous data points based on the latest literature, and expanding to include primate and mammal host infection labels—infection of primates may ultimately be the phylogenetic barrier crossing of greatest interest. We used the datasets presented here to train multiple ML models, providing the community with baseline estimator performances via a code base and set of data that are open and reproducible. It remains an open question as to whether it would be preferable to probe ML model performance on datasets that are

balanced on viral families or not—it depends, for example, on the interest in identifying truly novel pathogens relative to those that are quite similar to currently-known viruses.

We also found that models trained on mammalian infection show improved performance over those trained on human (sign test, 10 pairs of ROC AUC values, 8 ML models, min and max p-value of $p = 1.95 \times 10^{-3}$) or primate (sign test, 10 pairs of ROC AUC values, 8 ML models, min and max p-value of $p = 1.95 \times 10^{-3}$) host labels. This may seem like a somewhat obvious phenomenon, given the broader range of the mammalian label, however we also observed a narrower spread of ROC-AUC of models trained on the corrected dataset as compared to human and primate hosts, which suggests that certain features of the mammalian host target may contribute to improved classification relative to non-mammal host infection.

We recognize limitations of the work presented here which should be considered as these datasets develop. For example, we removed viruses if they did not contain any coding sequences divisible by 3 as these coding sequences are not used for feature calculation in [2]. However, viruses are known to abuse ribosomal frameshifting [28], and models with more robust feature calculation may be capable of making use of these genomes. Additionally, while we selected a train-test split for ease of comparison with results from [2], a smaller test set may be appropriate for other studies as we do not necessarily suggest a roughly even split between training and test data. It may have been possible for us to achieve greater impact by demonstrating that our rebalanced datasets enable superior generalization power on a *third* dataset of recently-emerged viral pathogens of interest, for example avian influenza infection of dairy cows and humans [29]. This may also highlight the need for specialized challenge datasets where a small number of nucleotide changes in viral genomes is known to confer a change in host range, to probe the specific strengths and weaknesses of predictive models.

Although these results are a step towards establishing a benchmark dataset for prediction of viral host infectivity from genomic information, open questions remain. It has not yet been established if a larger dataset with multiple isolates per species is preferable to a smaller dataset with distinct representative samples. Our work assumes that using a smaller dataset is an acceptable tradeoff to mitigate potential information leakage that arises when multiple isolates per species are included into both the training and test datasets. Perhaps what our community needs most is a committee that curates and updates datasets used for assessing viral host infectivity, and a spirit of competition to drive model excellence. Examples of competitions include those hosted on Kaggle, and there are also several machine learning benchmark working groups and committees: ML Commons maintains machine learning performance benchmarks like MLPerf [30], and the common objects in context (COCO) consortium maintains a standard dataset for the computer vision community [31].

4 Methods

4.1 Dataset Improvement

The original Mollentze et al. dataset classes include 'Human-host' vs 'Not human-host' labels for viral infection [2] indicating whether a certain virus is known to infect

a human or not. Initial results using these data classes show a large amount of false positives (Figure 7), which indicates that the true label may be 'human-host' but there may not be sufficient supporting data to make that claim. There is also potential for making phylogenetically informed classification decisions by relabeling the dataset to include classes that are aimed at different hosts. This modified target is based on the idea that whether a virus has spilled over into infecting primates may be a more informative indication of the zoonotic potential of the virus to infect humans. To this end, the viral dataset was relabeled from the original classes to include more broad categories using literature-based evidence. Each virus was relabeled as either "human," "Primate," "Mammal," "Avian," or "non-human" (i.e. insect, reptile, plant). Ground-truth relabeling decisions were made on a case-by-case basis by three experts using evidence of direct infection or presence of virus in host serum as positive confirmation, whereas evidence of infection in cell-culture was not considered sufficient. Additionally, mislabeled data-points were changed in certain cases based on new evidence. The precise justifications for host infection statuses are described in the `retarget.py` module in our open source code. Beyond straightforward fixes to remove problematic or duplicated viral records, the latest literature evidence allowed us to relabel 21 viral species from non-infecting to human infecting, while no viral species were switched from human infecting to not. The final datasets used in the machine learning workflow are available in CSV files stored in project path `viral_seq/data` (at the DOI listed below), and supporting genomic data at another DOI also specified below.

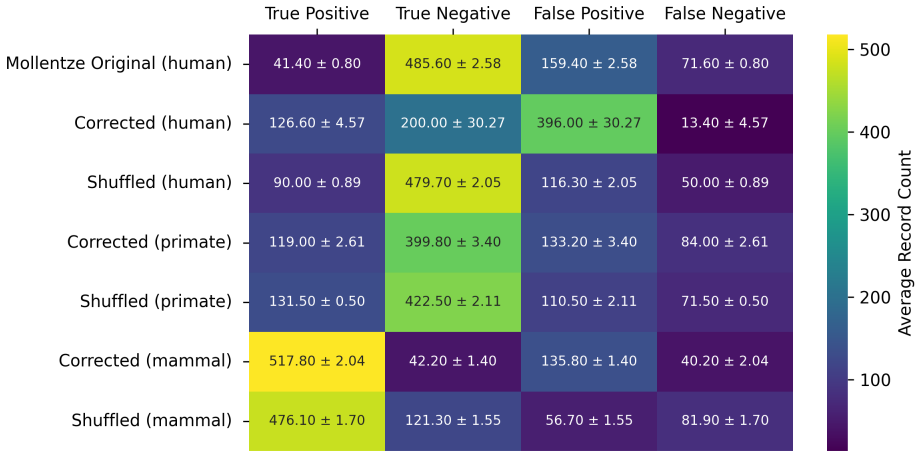


Fig. 7 Tabulation of confusion matrix data across hyperparameter optimized Random Forest estimators averaged across ten random seeds for their predictions on test set data. The confusion matrix data for other estimators may be reproduced using the open source code accompanying this work. Average and standard deviation values are shown for different datasets. The threshold used for prediction was the equal error rate (EER).

4.2 ML Workflow

Our open source machine learning workflow (https://github.com/lanl/ldrd_virus_work) may be used to fully reproduce the results of this work, at commit hash `c21dfd8e6`, `git` release tag `v0.1.0`, and Zenodo DOI https://doi.org/10.5281/zenodo.17074080. Features were calculated for each viral genome in a manner that was strongly influenced by the original workflow published by *Mollentze et al.* [2]. We evaluated the datasets using the following models: `RandomForestClassifier`, `ExtraTreesClassifier`, and SVC with the linear, polynomial, and radial basis function kernels as implemented in the scikit-learn package [22], `LGBMClassifier` using both the default gradient boosting decision trees and DART [27] algorithms from the LightGBM package [23], and `XGBClassifier` from the XGBoost package [24]. Each model's hyperparameters were optimized for each dataset using the tree-structured Parzen estimator algorithm [32] for the `OptunaSearch` subpackage [33] in the Ray Tune [34] Python package. The number of steps of optimization was chosen for each model such that the performance of five independent seeds appeared to roughly plateau as measured by the mean ROC AUC using 5-fold cross-validation on training data stratified on the target (Figure 8). For `RandomForestClassifier`, `ExtraTreesClassifier`, and SVC kernels we used a broad range of values across many hyperparameters (Table 2). For `XGBClassifier` and `LGBMClassifier` we used the ranges suggested in the documentation of Amazon SageMaker [35, 36], with the addition of the `drop_rate` and `skip_drop` hyperparameters over their full ranges for `LGBMClassifier` when using the DART algorithm. In addition to the software and libraries already mentioned, we also leveraged the following: SciPy [37], NumPy [38], polars [39], and pandas [40]. Plots were produced using matplotlib [41] and seaborn [42]. The statistical differences between ROC AUC values were assessed using the sign test provied by `statsmodels` [43]. In general, this involved a pairwise difference over 10 ROC AUC values—10 random seeds for each of the 8 estimators per condition. We reported the minimum and maximum p-values across those 8 comparisons for a given set of cases. We did not perform the sign test directly across all 80 estimator ROC AUC values because it was not clear that different estimators operating on the same random seed were sufficiently independent.

Hyperparameter	Random Forest	Extra Trees	LGBM Boost	LGBM DART	XGBoost	SVC linear	SVC poly	SVC rbf
max_samples	$[1/n, 1.0)$	$[1/n, 1.0)$	N/A	N/A	N/A	N/A	N/A	N/A
min_samples_leaf	$[1/n, \min(1, 10/n))$	$[1/n, \min(1, 10/n))$	N/A	N/A	N/A	N/A	N/A	N/A
min_samples_split	$[1/n, \min(1, 300/n))$	$[1/n, \min(1, 300/n))$	N/A	N/A	N/A	N/A	N/A	N/A
max_features	$[1/f, \min(1, 2\sqrt{f}/f))$	$[1/f, \min(1, 2\sqrt{f}/f))$	N/A	N/A	N/A	N/A	N/A	N/A
criterion	gini | log_loss	gini | log_loss	N/A	N/A	N/A	N/A	N/A	N/A
class weight	None | balanced | balanced_subsample	None | balanced | balanced_subsample	N/A	N/A	N/A	None | balanced	None | balanced	None | balanced
max_depth	$\infty \| [1, 30]$	$\infty \| [1, 30]$	$[15, 100)$	$[15, 100)$	N/A	N/A	N/A	N/A
num_leaves	N/A	N/A	$[10, 100)$	$[10, 100)$	N/A	N/A	N/A	N/A
learning_rate	N/A	N/A	$[0.001, 0.01)$	$[0.001, 0.01)$	$[0.1, 0.5)$	N/A	N/A	N/A
subsample	N/A	N/A	$[0.1, 1.0)$	$[0.1, 1.0)$	$[0.5, 1.0)$	N/A	N/A	N/A
subsample_freq	N/A	N/A	$[0, 10)$	$[0, 10)$	N/A	N/A	N/A	N/A
min_child_samples	N/A	N/A	$[10, 200)$	$[10, 200)$	N/A	N/A	N/A	N/A
colsample_bytree	Data	Data	$[0.1, 1.0)$	$[0.1, 1.0)$	Data	Data	Data	Data
reg_alpha	N/A	N/A	N/A	N/A	$[0.001, 1000)$	N/A	N/A	N/A
min_child_weight	N/A	N/A	N/A	N/A	$[0.001, 120)$	N/A	N/A	N/A
n_estimators	constant	constant	constant	constant	$[1, 4000)$	constant	constant	constant
drop_rate	N/A	N/A	N/A	$[0.0, 1.0)$	N/A	N/A	N/A	N/A
skip_drop	N/A	N/A	N/A	$[0.0, 1.0)$	N/A	N/A	N/A	N/A
C	N/A	N/A	N/A	N/A	N/A	$[10^{-4}, 10^{4})$	$[10^{-4}, 10^{4})$	$[10^{-4}, 10^{4})$
degree	N/A	N/A	N/A	N/A	N/A	N/A	$[2, 6)$	N/A
gamma	N/A	N/A	N/A	N/A	N/A	N/A	$[10^{-4}, 100)$	$[10^{-4}, 100)$
coef0	N/A	N/A	N/A	N/A	N/A	N/A	$[0.0, 10.0)$	N/A

Table 2 The range of hyperparameter values over which optimizations were performed. n represents the number of records in the design matrix, while f represents the number of features in the design matrix. In many cases the samples were drawn over the intervals using a uniform distribution, but other approaches were also used—see the open source machine learning workflow for details.

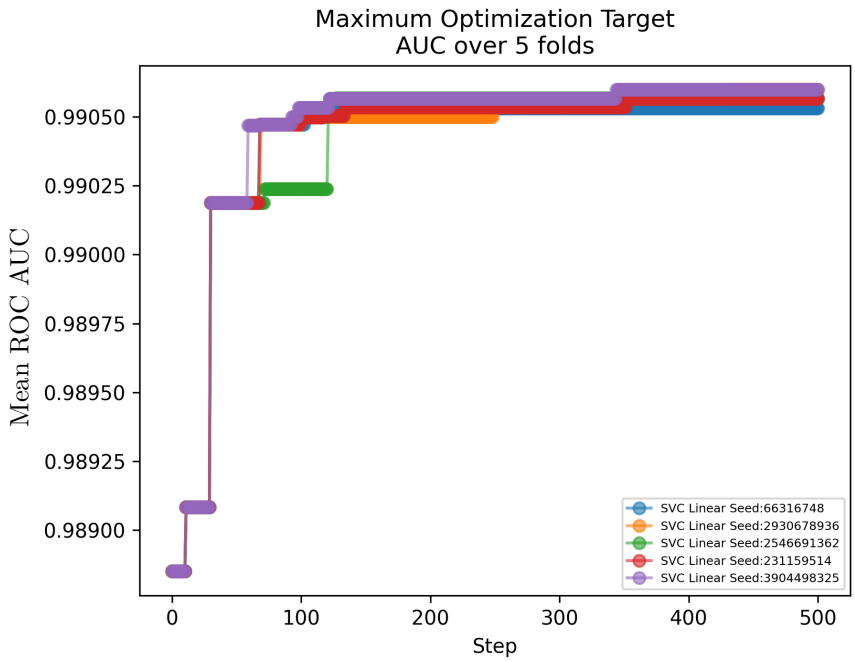


Fig. 8 Representative example of the process of determining the number of trials required for hyperparameter optimization. The estimation is based on a plateau of performance improvement across five independent random seeds with 5-fold cross-validation mean ROC AUC as the metric. In this case, approximately 500 steps appears to be sufficient to optimize the hyperparameters for `SVC` with a linear kernel.

5 Data Availability

All datasets used in this study are available in our GitHub repository at https://github.com/lanl/ldrd_virus_work. To ensure the long-term stability of our analysis workflow, the underlying viral genomes are stored in the above repository using `git lfs`, and the hash of the compressed data at the time of publication is `3a3d3f4e81`. This helps avoid changes that occur in upstream databases—over time, however, it may be helpful to update the genomes for corrections in future works. At the time of writing, the `git lfs` bandwidth of the above repository is limited, so we have also provided the dataset on `figshare` with DOI https://doi.org/10.6084/m9.figshare.30025597. The identities of the viruses in the various training and test sets are available in CSV files at project path `viral_seq/data`, at release version `v0.1.0`, and at DOI https://doi.org/10.5281/zenodo.17074080.

6 Code Availability

The ML code for this project, including the end-to-end workflow, is available on GitHub at https://github.com/lanl/ldrd_virus_work, and is CI-tested for CPython versions 3.10 through 3.13. The results of this work were specifically generated using the commit hash `c21dfd8`, release version `v0.1.0` and DOI https://doi.org/10.5281/zenodo.17074080. The code was released under a GPL-3.0 license because it depends on the GPL-licensed `taxonomy-ranks` [44] project, which in turn depends on the GPL-licensed `ete` project [45]. We are open to relicensing our project to a more liberal scheme if these dependencies, which are used for phylogenetic heat map production, could be replaced.

7 Author Contributions

A.R.H., N.H., and T.R. designed the study. A.R.H., A.W., A.S., and T.R. contributed to code development and dataset curation. A.R.H., A.W., A.S., N.H. and T.R. carried out the analysis and wrote the manuscript. T.R. and N.H. acquired funding for the project and provided feedback at all stages.

8 Competing Interests

The authors declare no competing interests.

9 Acknowledgements

We thank Emma Goldberg and Anastasiia Kim for helpful feedback on the overall project design. This research used resources provided by the Darwin testbed at Los Alamos National Laboratory (LANL) which is funded by the Computational Systems and Software Environments subprogram of LANL's Advanced Simulation and Computing program (NNSA/DOE). This research used resources provided by the Los Alamos National Laboratory Institutional Computing Program, which is supported by the U.S. Department of Energy National Nuclear Security Administration under

Contract 89233218CNA000001. The project was funded by the Laboratory Directed Research and Development grant 20230044DR at LANL.

References

[1] Piret, J., Boivin, G.: Pandemics throughout history. Frontiers in Microbiology **Volume 11 - 2020** (2021) https://doi.org/10.3389/fmicb.2020.631736

[2] Mollentze, N., Babayan, S.A., Streicker, D.G.: Identifying and prioritizing potential human-infecting viruses from their genome sequences. PLOS Biology **19**(9), 1–25 (2021) https://doi.org/10.1371/journal.pbio.3001390

[3] Grange, Z.L., Goldstein, T., Johnson, C.K., Anthony, S., Gilardi, K., Daszak, P., Olival, K.J., O'Rourke, T., Murray, S., Olson, S.H., Togami, E., Vidal, G., Panel, E., Consortium, P., Mazet, J.A.K., Anderson, K., Auewarakul, P., Coffey, L., Corley, R., Dauphin, G., Epstein, J., Fukuda, K., Goodman, S., Han, B., Hughes, J., Jeggo, M., Karesh, W., Kazwala, R., Kelly, T., Keusch, G., Kurilla, M., Mackenzie, J., Markotter, W., Monagin, C., Morens, D., Munster, V., Muhlberger, E., Pandit, P., Peel, A., Pfeiffer, D., Restif, O., Tomori, O., Towner, J., Werf, S.V.D., VonDobschetz, S., Wacharapluesadee, S., Ward, M., Weirsma, L., Wilson, M., Wolking, D., Wongsathapornchai, K., Brierley, L., Edinburgh Epigroup members, U., Tambrana-Torellio, C., anonymous, Islam, A., Islam, S., Raman, Z., Hul, V., Duong, V., Mouiche, M., Nwobegahay, J., Coulibaly, K., Kumakamba, C., Syaluha, E.K., Lukusa, J.-P., Belay, D., Kebede, N., Ampofo, W., Bel-Nono, S., Suu-Ire, R., Douokoro, K., Dursman, H., Pamungkas, I., Rachmitasari, N., Saputro, S., Damanik, W., Kusumaningrum, T., Rambitan, M., Rey, B., Safari, D., Soebandrio, A., Triastuti, J., Abu-Basha, E., Allan, K., Joseph, K., Samson, M., Khamphaphonphane, B., Theppanga, W., Desmond, J., Samules, S., Lee, M.H., Lee, J., Damdinjav, B., Shiilegdamba, E., Aung, O., Bista, M., Karmacharya, D., Shrestha, R., Nziza, J., Tumushime, J.-C., Lo, M.M., Ndiaye, A., Seck, M.C., Bangura, J., Lavalie, E., Mwangoka, G., Sijali, Z., Okello, R.O., Ssebide, B., Wacharpluesadee, S., Nguyen, N., Epstein, J., Hagan, E., Karesh, W., Latinne, A., Laudisoit, A., Li, H., Machalaba, C., Martinez, S., Ross, N., Sullivan, A., Torrelio, C.Z., Mackenzie, J., Waldman, R., Morzaria, S., Kalpravidh, W., Makonnen, Y., Dubscheutz, S.V., Claes, F., Pelican, K., Behravesh, C.B., Mumford, E., Clark, J.P., Vu, T.D., Saylors, K., Edison, B., Euren, J., Gillis, A., Lange, C., LeBreton, M., McIver, D., O'Rourke, D., Valitutto, M., Zimmerman, D., Belkhiria, J., Bird, B., Chale, H., Preston, E., Gardner, N., Genovese, B., Gonzalez, K., Keatts, L., Kelly, T., Leasure, E., Monagin, C., Pandit, P., Randhawa, N., Smith, B., Smith, W., Tremeau-Bravard, A., Wolking, D., Churchill, C., Olson, S., Walzer, C., Fine, A.: Ranking the risk of animal-to-human spillover for newly discovered viruses. Proceedings of the National Academy of Sciences **118**(15), 2002324118 (2021) https://doi.org/10.1073/pnas.2002324118 https://www.pnas.org/doi/pdf/10.1073/pnas.2002324118

[4] Olival, K.J., Hosseini, P.R., Zambrana-Torrelio, C., Ross, N., Bogich, T.L.,

Daszak, P.: Host and viral traits predict zoonotic spillover from mammals. Nature **546**(7660), 646–650 (2017) https://doi.org/10.1038/nature22975

[5] Tseng, K.K., Koehler, H., Becker, D.J., Gibb, R., Carlson, C.J., Pilar Fernandez, M.d., Seifert, S.N.: Viral genomic features predict orthopoxvirus reservoir hosts. Communications Biology **8**(1), 309 (2025) https://doi.org/10.1038/s42003-025-07746-0

[6] Benson, N.C., Daggett, V.: A comparison of multiscale methods for the analysis of molecular dynamics simulations. The Journal of Physical Chemistry B **116**(29), 8722–8731 (2012) https://doi.org/10.1021/jp302103t

[7] Merk, A., Bartesaghi, A., Banerjee, S., Falconieri, V., Rao, P., Davis, M.I., Pragani, R., Boxer, M.B., Earl, L.A., Milne, J.L.S., Subramaniam, S.: Breaking cryo-em resolution barriers to facilitate drug discovery. Cell **165**(7), 1698–1707 (2016) https://doi.org/10.1016/j.cell.2016.05.040

[8] Wang, A., Singh, A., Michael, J., Hill, F., Levy, O., Bowman, S.R.: GLUE: A multi-task benchmark and analysis platform for natural language understanding. CoRR **abs/1804.07461** (2018) https://doi.org/10.48550/arXiv.1804.07461 1804.07461

[9] Russakovsky, O., Deng, J., Su, H., Krause, J., Satheesh, S., Ma, S., Huang, Z., Karpathy, A., Khosla, A., Bernstein, M., Berg, A.C., Fei-Fei, L.: Imagenet large scale visual recognition challenge. International Journal of Computer Vision **115**(3), 211–252 (2015) https://doi.org/10.1007/s11263-015-0816-y

[10] Zhang, Z., Cai, Z., Tan, Z., Lu, C., Jiang, T., Zhang, G., Peng, Y.: Rapid identification of human-infecting viruses. Transboundary and Emerging Diseases **66**(6), 2517–2522 (2019) https://doi.org/10.1111/tbed.13314 https://onlinelibrary.wiley.com/doi/pdf/10.1111/tbed.13314

[11] Bartoszewicz, J.M., Seidel, A., Renard, B.Y.: Interpretable detection of novel human viruses from genome sequencing data. NAR Genomics and Bioinformatics **3**(1), 004 (2021) https://doi.org/10.1093/nargab/lqab004 https://academic.oup.com/nargab/article-pdf/3/1/lqab004/36165658/lqab004.pdf

[12] Woolhouse, M., Scott, F., Hudson, Z., Howey, R., Chase-Topping, M.: Human viruses: discovery and emergence. Philosophical Transactions of the Royal Society B: Biological Sciences **367**(1604), 2864–2871 (2012) https://doi.org/10.1098/rstb.2011.0354 https://royalsocietypublishing.org/doi/pdf/10.1098/rstb.2011.0354

[13] Forni, D., Cagliani, R., Clerici, M., Sironi, M.: Disease-causing human viruses: novelty and legacy. Trends in Microbiology **30**(12), 1232–1242 (2022) https://doi.org/10.1016/j.tim.2022.07.002

[14] Young, F., Rogers, S., Robertson, D.L.: Predicting host taxonomic information from viral genomes: A comparison of feature representations. PLOS Computational Biology **16**(5), 1–24 (2020) https://doi.org/10.1371/journal.pcbi.1007894

[15] Bernett, J., Blumenthal, D.B., Grimm, D.G., Haselbeck, F., Joeres, R., Kalinina, O.V., List, M.: Guiding questions to avoid data leakage in biological machine learning applications. Nature Methods **21**(8), 1444–1453 (2024) https://doi.org/10.1038/s41592-024-02362-y

[16] Lones, M.A.: Avoiding common machine learning pitfalls. Patterns **5**(10) (2024) https://doi.org/10.1016/j.patter.2024.101046

[17] Japkowicz, N.: 8. Assessment Metrics for Imbalanced Learning, pp. 187–206. John Wiley & Sons, Ltd, ??? (2013). https://doi.org/10.1002/9781118646106.ch8 . https://onlinelibrary.wiley.com/doi/abs/10.1002/9781118646106.ch8

[18] Sokolova, M., Japkowicz, N., Szpakowicz, S.: Beyond accuracy, f-score and roc: A family of discriminant measures for performance evaluation. In: Sattar, A., Kang, B.-h. (eds.) AI 2006: Advances in Artificial Intelligence, pp. 1015–1021. Springer, Berlin, Heidelberg (2006)

[19] Haixiang, G., Yijing, L., Shang, J., Mingyun, G., Yuanyue, H., Bing, G.: Learning from class-imbalanced data: Review of methods and applications. Expert Systems with Applications **73**, 220–239 (2017) https://doi.org/10.1016/j.eswa.2016.12.035

[20] Maier, S., Zimmermann, P., Daub, R.: Machine learning driven design of experiments for predictive models in production systems. ESSN: 2701-6277, 110–118 (2023) https://doi.org/10.15488/15289

[21] Bartoszewicz, J.M., Seidel, A., Rentzsch, R., Renard, B.Y.: Deepac: predicting pathogenic potential of novel dna with reverse-complement neural networks. Bioinformatics **36**(1), 81–89 (2019) https://doi.org/10.1093/bioinformatics/btz541 https://academic.oup.com/bioinformatics/article-pdf/36/1/81/48981489/bioinformatics_36_1_81.pdf

[22] Pedregosa, F., Varoquaux, G., Gramfort, A., Michel, V., Thirion, B., Grisel, O., Blondel, M., Prettenhofer, P., Weiss, R., Dubourg, V., VanderPlas, J., Passos, A., Cournapeau, D., Brucher, M., Perrot, M., Duchesnay, E.: Scikit-learn: Machine learning in python. CoRR **abs/1201.0490** (2012) https://doi.org/10.48550/arXiv.1201.0490 1201.0490

[23] Ke, G., Meng, Q., Finley, T., Wang, T., Chen, W., Ma, W., Ye, Q., Liu, T.-Y.: Lightgbm: A highly efficient gradient boosting decision tree. In: Guyon, I., Luxburg, U.V., Bengio, S., Wallach, H., Fergus, R., Vishwanathan, S., Garnett, R. (eds.) Advances in Neural Information Processing Systems, vol. 30. Curran Associates, Inc., ??? (2017)

[24] Chen, T., Guestrin, C.: Xgboost: A scalable tree boosting system. In: Proceedings of the 22nd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining. KDD '16, pp. 785–794. Association for Computing Machinery, New York, NY, USA (2016). https://doi.org/10.1145/2939672.2939785 . https://doi.org/10.1145/2939672.2939785

[25] Johnson, C.K., Hitchens, P.L., Pandit, P.S., Rushmore, J., Evans, T.S., Young, C.C.W., Doyle, M.M.: Global shifts in mammalian population trends reveal key predictors of virus spillover risk. Proceedings of the Royal Society B: Biological Sciences **287**(1924), 20192736 (2020) https://doi.org/10.1098/rspb.2019.2736 https://royalsocietypublishing.org/doi/pdf/10.1098/rspb.2019.2736

[26] Rosenberg, R.: Detecting the emergence of novel, zoonotic viruses pathogenic to humans. Cellular and Molecular Life Sciences **72**(6), 1115–1125 (2015) https://doi.org/10.1007/s00018-014-1785-y

[27] Korlakai Vinayak, R., Gilad-Bachrach, R.: DART: Dropouts meet Multiple Additive Regression Trees. In: Lebanon, G., Vishwanathan, S.V.N. (eds.) Proceedings of the Eighteenth International Conference on Artificial Intelligence and Statistics. Proceedings of Machine Learning Research, vol. 38, pp. 489–497. PMLR, San Diego, California, USA (2015). https://doi.org/10.48550/arXiv.1505.01866 . https://proceedings.mlr.press/v38/korlakaivinayak15.html

[28] Brierley, I.: Ribosomal frameshifting on viral rnas. Journal of General Virology **76**(8), 1885–1892 (1995) https://doi.org/10.1099/0022-1317-76-8-1885

[29] Bellido-Martín, B., Rijnink, W.F., Iervolino, M., Kuiken, T., Richard, M., Fouchier, R.A.: Evolution, spread and impact of highly pathogenic h5 avian influenza a viruses. Nature Reviews Microbiology, 1–16 (2025)

[30] Reddi, V.J., Cheng, C., Kanter, D., Mattson, P., Schmuelling, G., Wu, C.-J., Anderson, B., Breughe, M., Charlebois, M., Chou, W., Chukka, R., Coleman, C., Davis, S., Deng, P., Diamos, G., Duke, J., Fick, D., Gardner, J.S., Hubara, I., Idgunji, S., Jablin, T.B., Jiao, J., John, T.S., Kanwar, P., Lee, D., Liao, J., Lokhmotov, A., Massa, F., Meng, P., Micikevicius, P., Osborne, C., Pekhimenko, G., Rajan, A.T.R., Sequeira, D., Sirasao, A., Sun, F., Tang, H., Thomson, M., Wei, F., Wu, E., Xu, L., Yamada, K., Yu, B., Yuan, G., Zhong, A., Zhang, P., Zhou, Y.: MLPerf Inference Benchmark (2019)

[31] Lin, T.-Y., Maire, M., Belongie, S., Hays, J., Perona, P., Ramanan, D., Dollár, P., Zitnick, C.L.: Microsoft coco: Common objects in context. In: Fleet, D., Pajdla, T., Schiele, B., Tuytelaars, T. (eds.) Computer Vision – ECCV 2014, pp. 740–755. Springer, Cham (2014)

[32] Bergstra, J., Bardenet, R., Bengio, Y., Kégl, B.: Algorithms for hyper-parameter optimization. In: Proceedings of the 25th International Conference on Neural Information Processing Systems. NIPS'11, pp. 2546–2554. Curran Associates Inc.,

Red Hook, NY, USA (2011)

[33] Akiba, T., Sano, S., Yanase, T., Ohta, T., Koyama, M.: Optuna: A next-generation hyperparameter optimization framework. In: Proceedings of the 25th ACM SIGKDD International Conference on Knowledge Discovery & Data Mining. KDD '19, pp. 2623–2631. Association for Computing Machinery, New York, NY, USA (2019). https://doi.org/10.1145/3292500.3330701 . https://doi.org/10.1145/3292500.3330701

[34] Liaw, R., Liang, E., Nishihara, R., Moritz, P., Gonzalez, J.E., Stoica, I.: Tune: A research platform for distributed model selection and training. arXiv preprint arXiv:1807.05118 (2018) https://doi.org/10.48550/arXiv.1807.05118 arXiv:1807.05118 [cs.LG]

[35] Amazon Web Services, I.: Tune an XGBoost Model - Amazon SageMaker AI. https://docs.aws.amazon.com/sagemaker/latest/dg/xgboost-tuning.html Accessed 2025-06-21

[36] Amazon Web Services, I.: Tune a LightGBM model - Amazon SageMaker AI (2025). https://docs.aws.amazon.com/sagemaker/latest/dg/lightgbm-tuning.html Accessed 2025-06-21

[37] Virtanen, P., Gommers, R., Oliphant, T.E., Haberland, M., Reddy, T., Cournapeau, D., Burovski, E., Peterson, P., Weckesser, W., Bright, J., Walt, S.J., Brett, M., Wilson, J., Millman, K.J., Mayorov, N., Nelson, A.R.J., Jones, E., Kern, R., Larson, E., Carey, C.J., Polat, İ., Feng, Y., Moore, E.W., VanderPlas, J., Laxalde, D., Perktold, J., Cimrman, R., Henriksen, I., Quintero, E.A., Harris, C.R., Archibald, A.M., Ribeiro, A.H., Pedregosa, F., Mulbregt, P., Vijaykumar, A., Bardelli, A.P., Rothberg, A., Hilboll, A., Kloeckner, A., Scopatz, A., Lee, A., Rokem, A., Woods, C.N., Fulton, C., Masson, C., Häggström, C., Fitzgerald, C., Nicholson, D.A., Hagen, D.R., Pasechnik, D.V., Olivetti, E., Martin, E., Wieser, E., Silva, F., Lenders, F., Wilhelm, F., Young, G., Price, G.A., Ingold, G.-L., Allen, G.E., Lee, G.R., Audren, H., Probst, I., Dietrich, J.P., Silterra, J., Webber, J.T., Slavič, J., Nothman, J., Buchner, J., Kulick, J., Schönberger, J.L., Miranda Cardoso, J., Reimer, J., Harrington, J., Rodríguez, J.L.C., Nunez-Iglesias, J., Kuczynski, J., Tritz, K., Thoma, M., Newville, M., Kümmerer, M., Bolingbroke, M., Tartre, M., Pak, M., Smith, N.J., Nowaczyk, N., Shebanov, N., Pavlyk, O., Brodtkorb, P.A., Lee, P., McGibbon, R.T., Feldbauer, R., Lewis, S., Tygier, S., Sievert, S., Vigna, S., Peterson, S., More, S., Pudlik, T., Oshima, T., Pingel, T.J., Robitaille, T.P., Spura, T., Jones, T.R., Cera, T., Leslie, T., Zito, T., Krauss, T., Upadhyay, U., Halchenko, Y.O., Vázquez-Baeza, Y., Contributors, S...: Scipy 1.0: fundamental algorithms for scientific computing in python. Nature Methods **17**(3), 261–272 (2020) https://doi.org/10.1038/s41592-019-0686-2

[38] Harris, C.R., Millman, K.J., Walt, S.J., Gommers, R., Virtanen, P., Cournapeau, D., Wieser, E., Taylor, J., Berg, S., Smith, N.J., Kern, R., Picus, M., Hoyer,

S., Kerkwijk, M.H., Brett, M., Haldane, A., Río, J.F., Wiebe, M., Peterson, P., Gérard-Marchant, P., Sheppard, K., Reddy, T., Weckesser, W., Abbasi, H., Gohlke, C., Oliphant, T.E.: Array programming with numpy. Nature **585**(7825), 357–362 (2020) https://doi.org/10.1038/s41586-020-2649-2

[39] Vink, R.: Polars: Blazingly fast DataFrames in Rust, Python, Node.js, R, and SQL. https://github.com/pola-rs/polars

[40] McKinney, W., Team, P.: Pandas-powerful python data analysis toolkit. Pandas—Powerful Python Data Analysis Toolkit **1625** (2015)

[41] Hunter, J.D.: Matplotlib: A 2d graphics environment. Computing in Science & Engineering **9**(3), 90–95 (2007) https://doi.org/10.1109/MCSE.2007.55

[42] Waskom, M.L.: seaborn: statistical data visualization. Journal of Open Source Software **6**(60), 3021 (2021) https://doi.org/10.21105/joss.03021

[43] Skipper, S., Josef, P.: statsmodels: Econometric and statistical modeling with python. 9th Python in Science Conference (2010)

[44] Meng, G., Li, Y., Yang, C., Liu, S.: Mitoz: a toolkit for animal mitochondrial genome assembly, annotation and visualization. Nucleic acids research **47**(11), 63–63 (2019)

[45] Huerta-Cepas, J., Serra, F., Bork, P.: Ete 3: reconstruction, analysis, and visualization of phylogenomic data. Molecular biology and evolution **33**(6), 1635–1638 (2016)