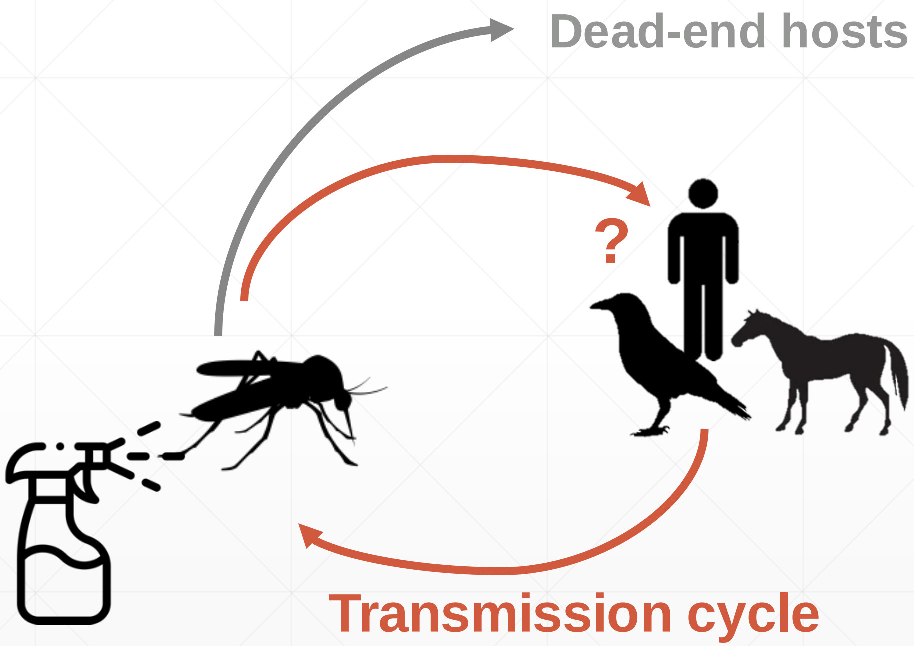
A novel machine learning approach to assess the risk of future mosquito-borne disease outbreaks

Clara Delecroix, PhD candidate
28/08/2024

Why is it important?

Increasing threat of mosquito-borne diseases

Control efforts: mosquito population control



Machine learning approaches in epidemiology

Many potential applications for machine learning approaches in epidemiology.

Risk mapping

(Mapping the transmission risk of Zika virus using machine learning models, Jiang et al, 2018)

Forecasting

(Machine learning and dengue forecasting: Comparing random forests and artificial neural networks for predicting dengue burden at national and sub-national scales in Colombia, Zhao et al, 2020)

Machine learning approaches in epidemiology

Many potential applications for machine learning approaches in epidemiology.

Risk mapping

Only tells us about suitability, not if an outbreak will start

Forecasting

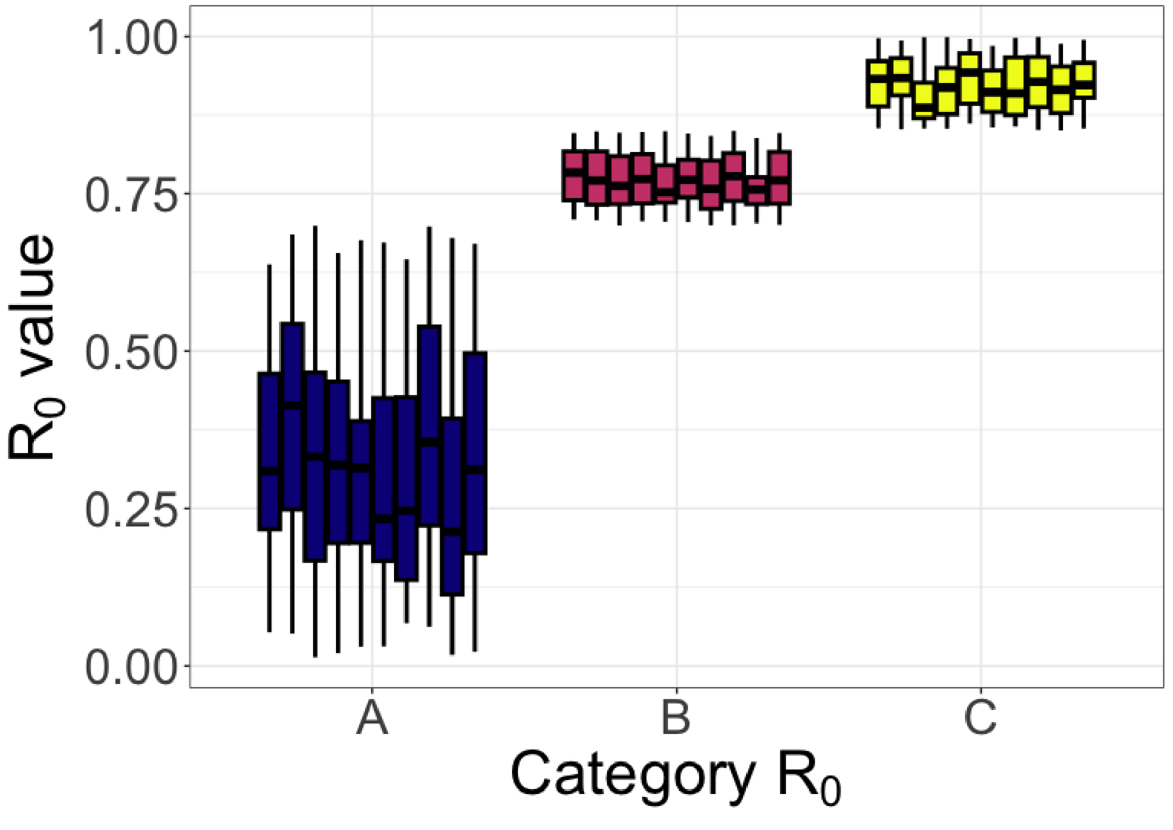
Requires large amounts of data for training the algorithm

Objective: predict the risk of future outbreaks

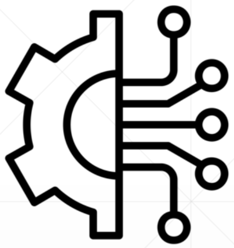
Use time series to classify into risk categories based on R_0

Generic, suitable for several mosquito-borne diseases

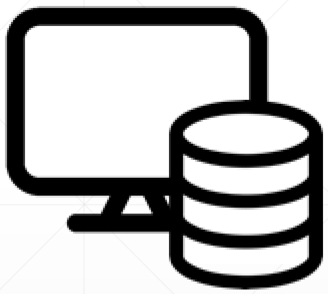
Reduce the data requirements compared to existing approaches:
1. Use pre-trained, computer vision models
2. Train using synthetic data



Can we predict the risk of future outbreaks of mosquito-borne diseases using a machine learning approaches?

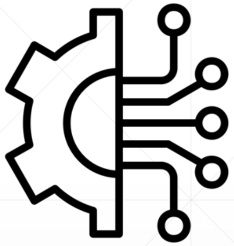


Machine learning pipeline



Creation of the synthetic training dataset

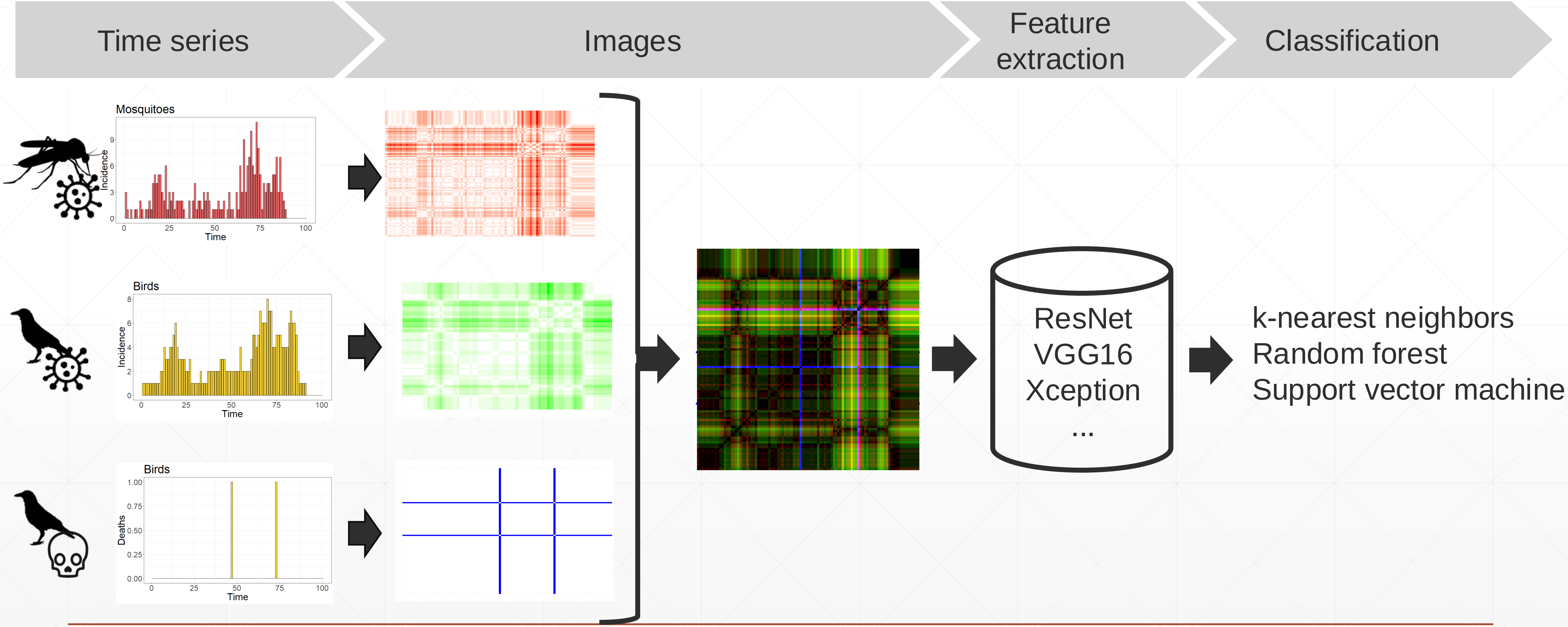
Can we outperform existing machine learning approaches to predict the risk of future outbreaks of mosquito-borne diseases by transforming the data?



Machine learning pipeline

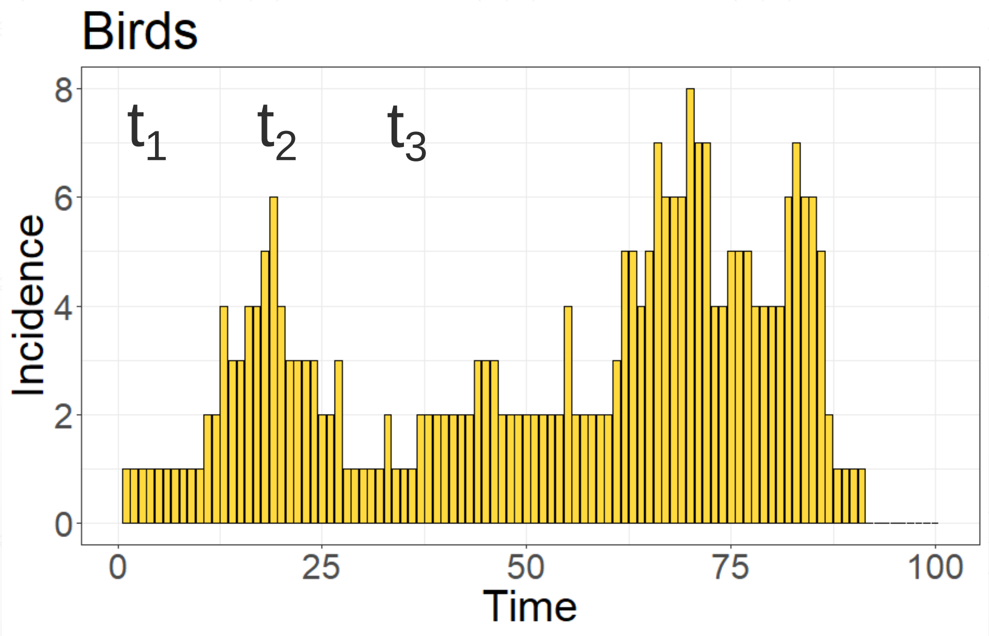
Creation of the synthetic training dataset

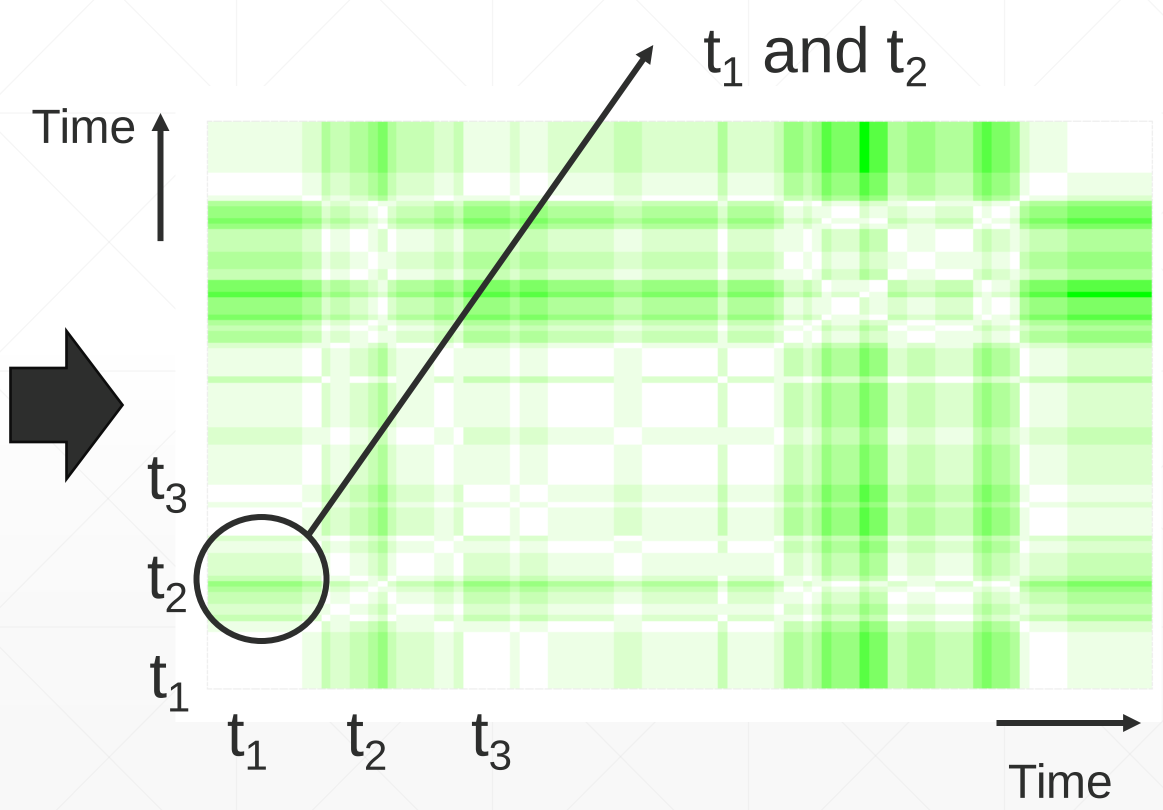
Machine learning pipeline



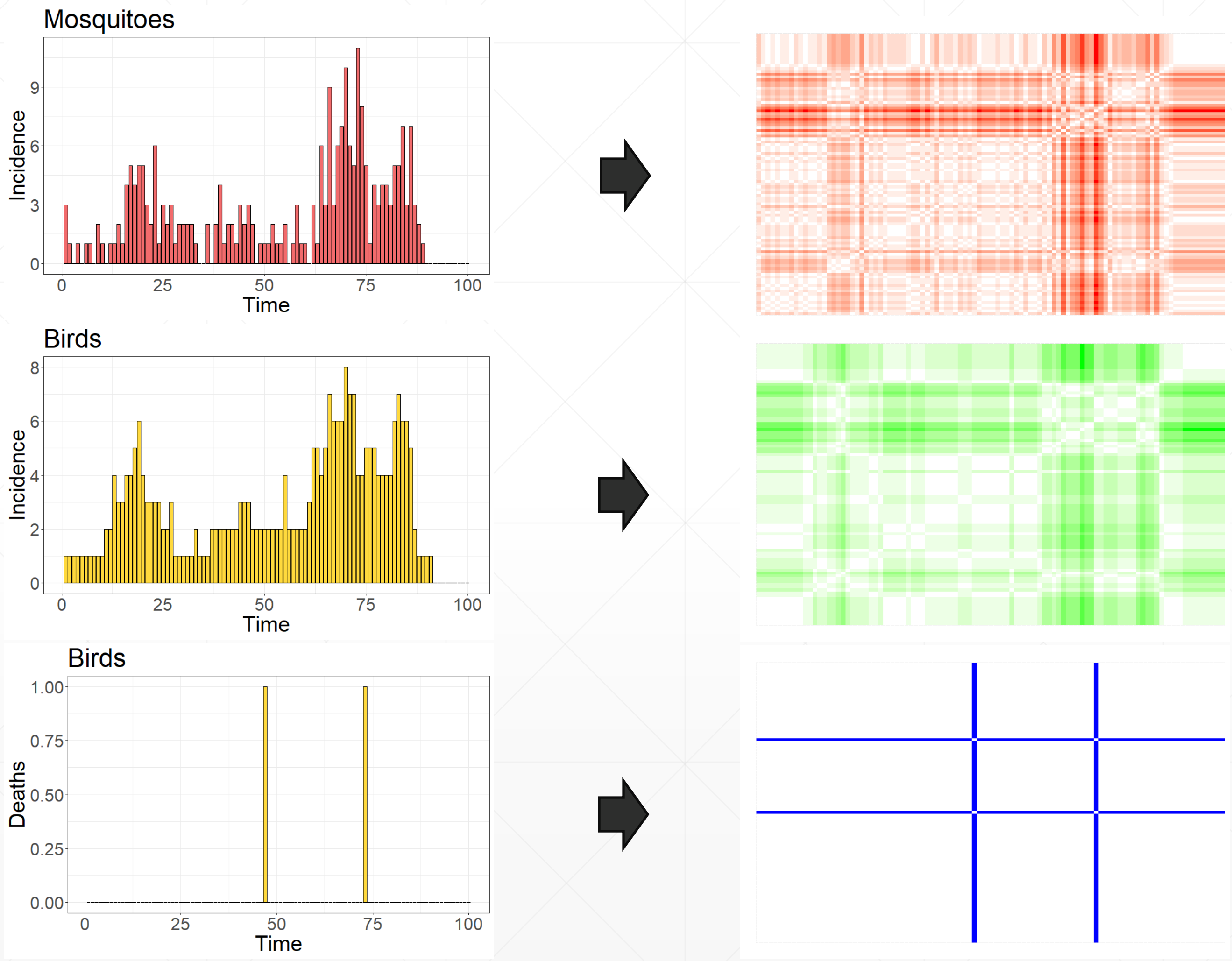
From time series to images

Recurrence plots:
Euclidean distance between each pair of points



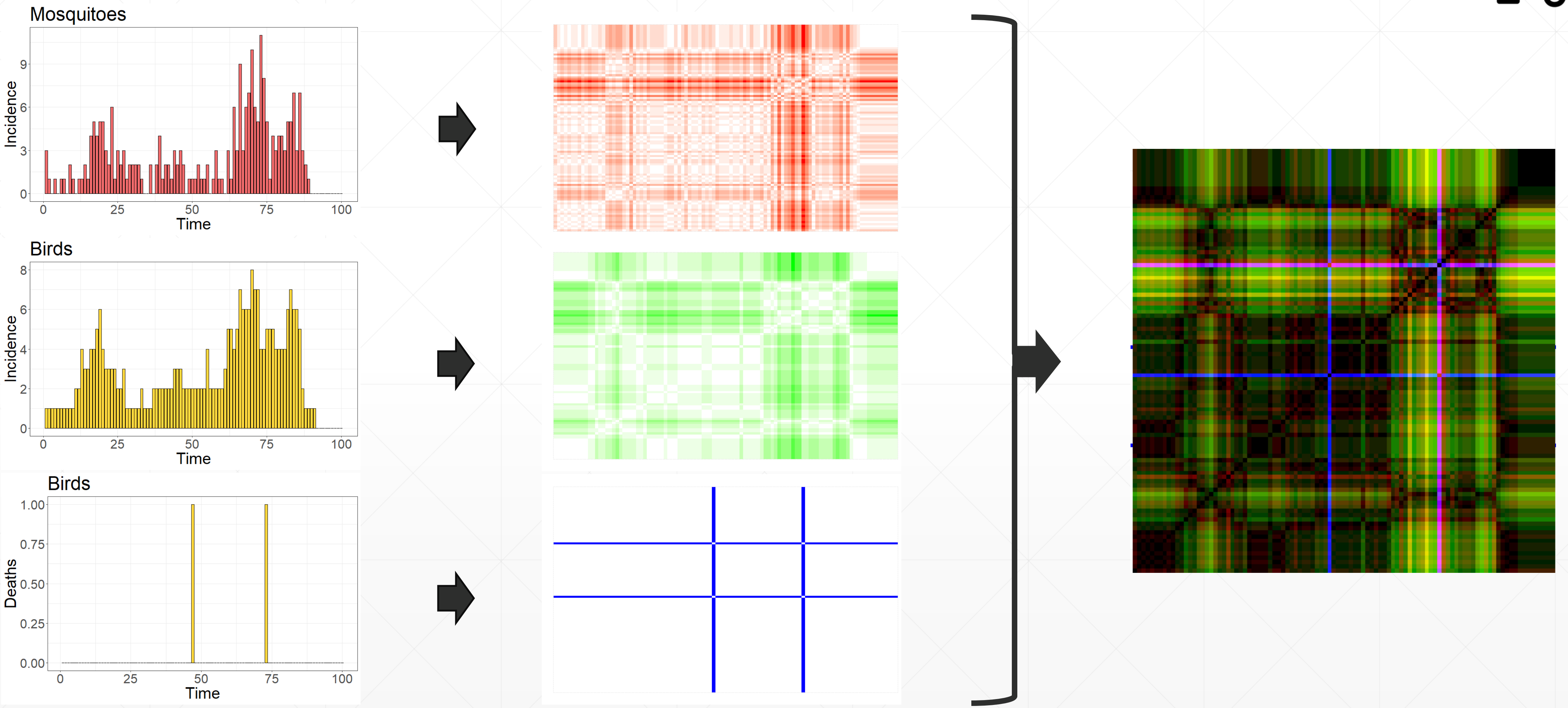


From time series to images



Recurrence plots: Euclidean distance between each pair of points

From time series to images



Transfer learning using feature extraction

- Used in computer vision: extract important features in an image
- Algorithms trained on large dataset of images
- **What is in this picture?**

Transfer learning using feature extraction

- Used in computer vision: extract important features in an image
- Algorithms trained on large dataset of images
- **What is in this picture?**
- Important features of the picture

Classification

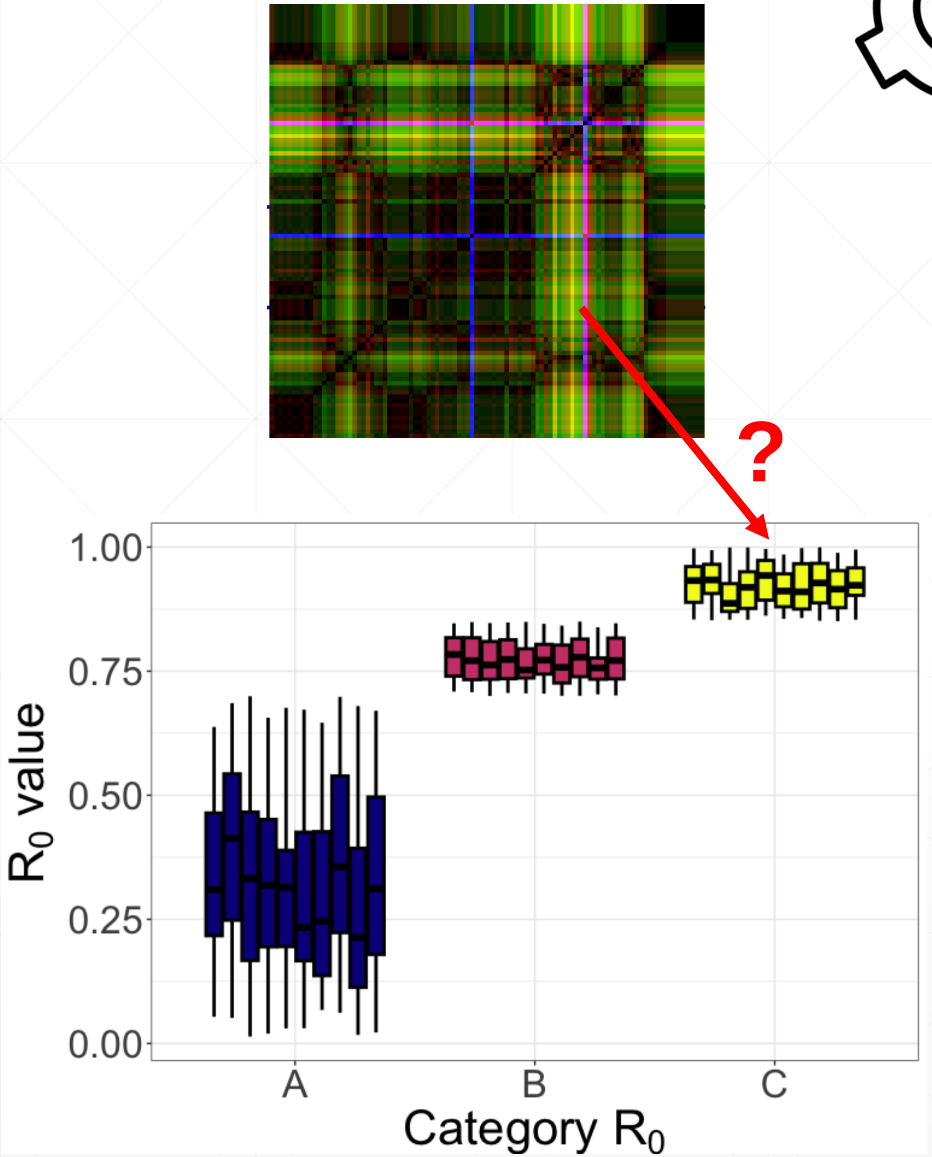
- Predict the correct label given input data
- **What is in this picture?**
  - A truck
  - A cat
  - A bike

Classification

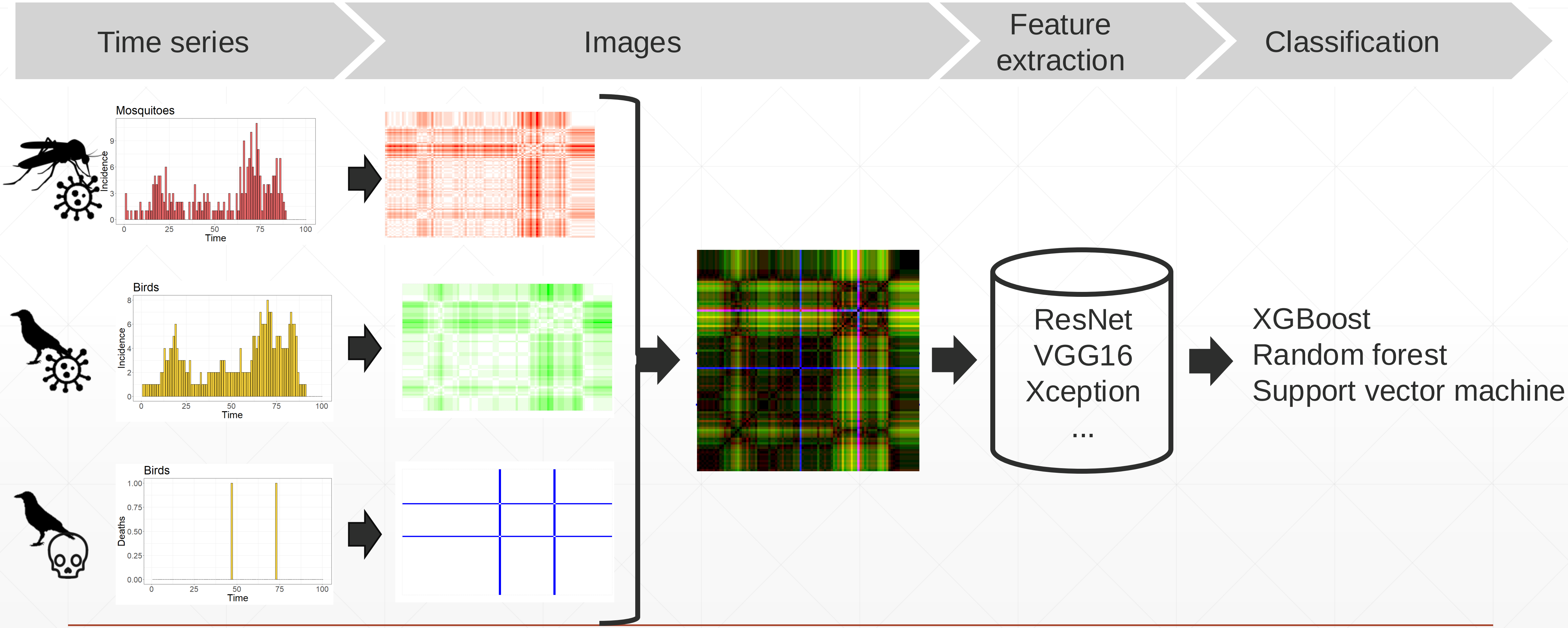
- Predict the correct label given input data
- **What is in this picture?**
  - A truck
  - A cat
  - **A bike**

Classification

- Predict the correct label given input data
- **For our problem: identify the correct R_0 category**



Machine learning pipeline



Can we outperform existing machine learning approaches to predict the risk of future outbreaks of mosquito-borne diseases by transforming the data?

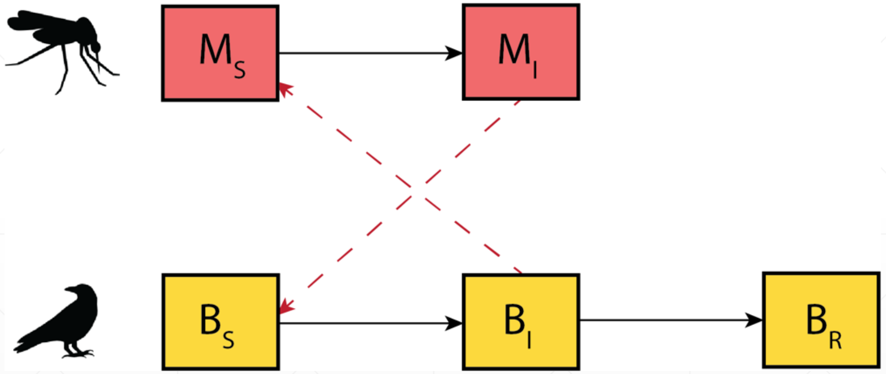
Machine learning pipeline

Creation of the synthetic training dataset

Creation of the synthetic dataset

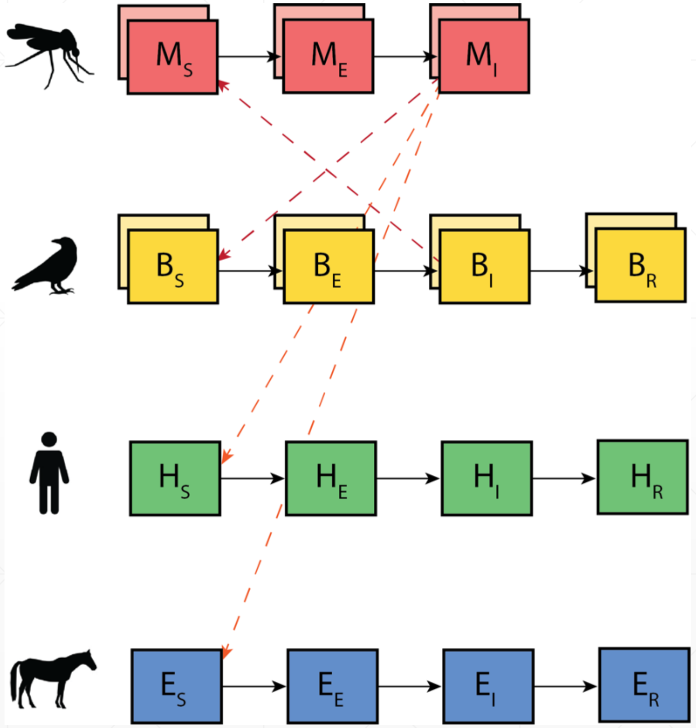
To ensure enough diversity in the dataset: random model generator

Minimal model assumptions



And everything in-between

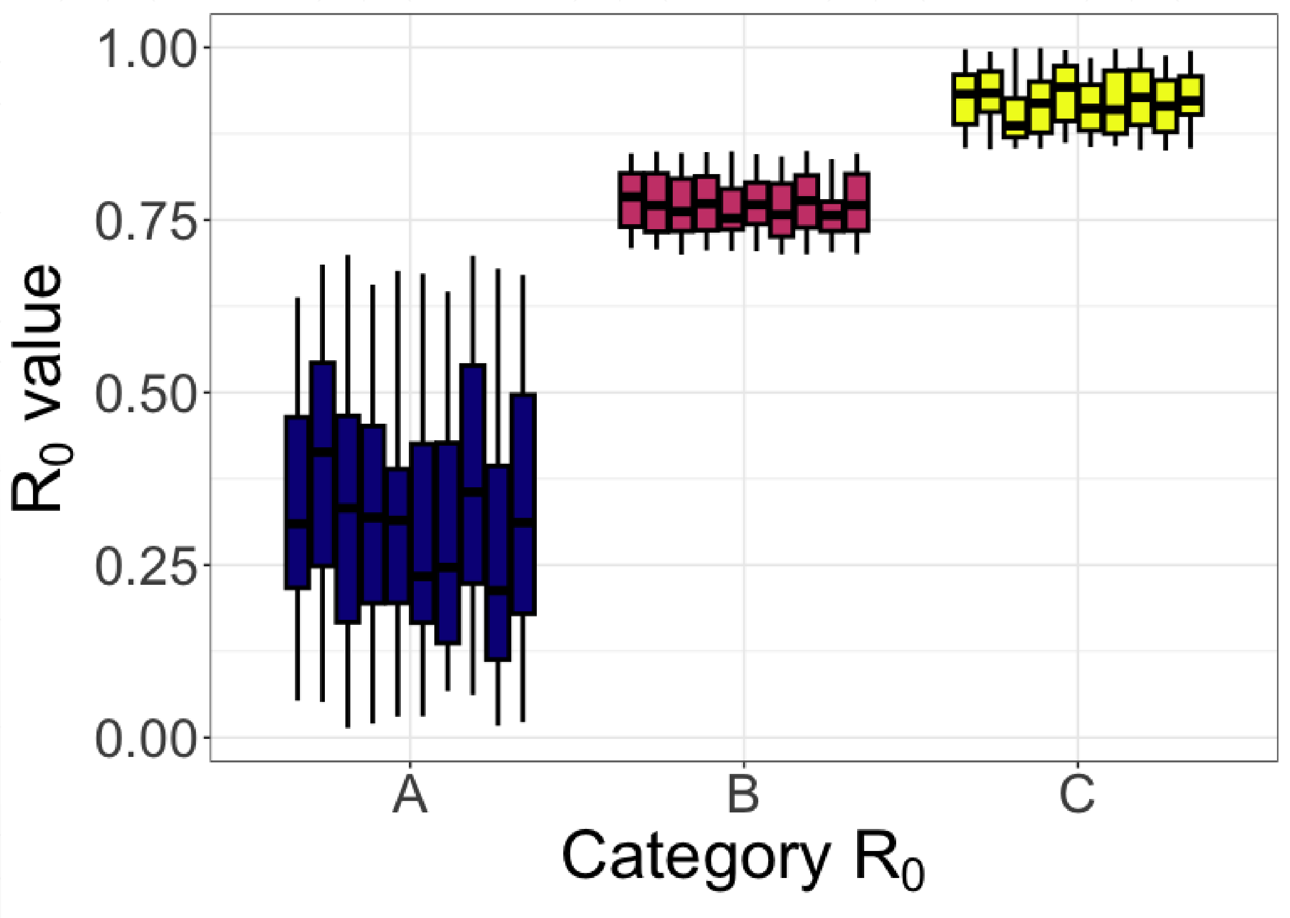
All model assumptions



Creation of the synthetic dataset

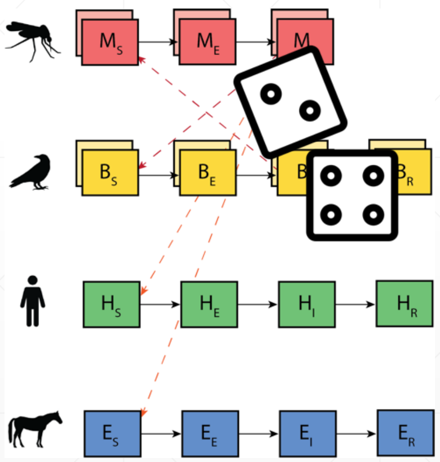
R_0 is calculated using the Next Generation Matrix

We defined several categories for the classification

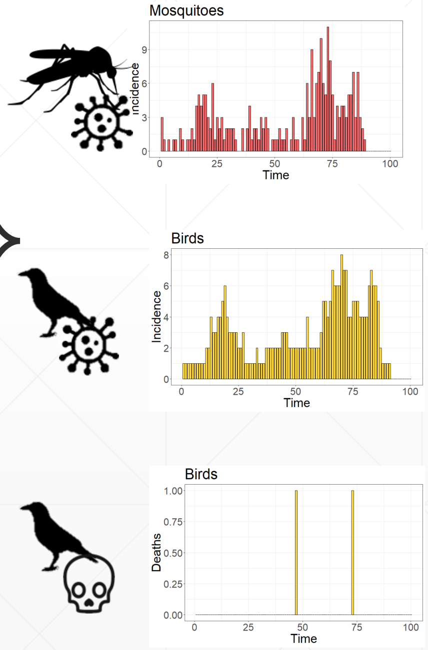


Summary of the approach

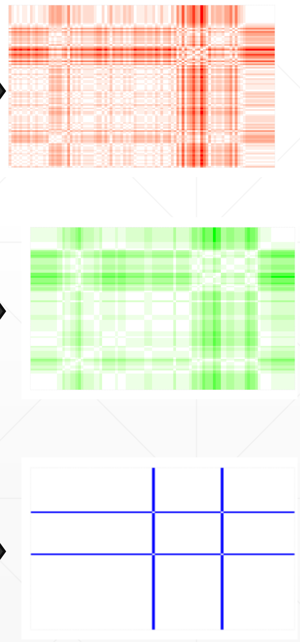
Random model generator

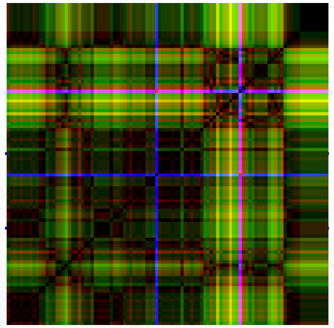


Used to train



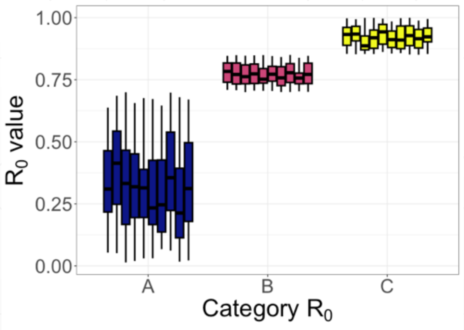
Machine learning pipeline





Feature extraction + classification

Assess risk of future outbreak



Results

Performance metric: **accuracy**, i.e. proportion of correctly classified instances

Feature extractor \ Classification algorithm	XGBoost	Support vector machine (SVM)	Random forest (RF)	CNN (on raw time series)
ResNet	0.794	0.827	0.817	0.649

Conclusion

- Machine learning approaches have the potential to improve control efforts for mosquito-borne diseases by assessing the risk of future outbreaks
- Transforming time series into images allows to leverage pre-trained computer vision algorithms
- Using a random model generator makes the framework flexible to many mosquito-borne diseases
- Future work: validate the pipeline with real-data

Thank you for your attention!

Special thanks to everybody involved in the project


Hien thi dieu Truong


Ingrid van de Leemput


Ricardo da Silva Torres


Quirine ten Bosch


Egbert van Nes


Marten Scheffer

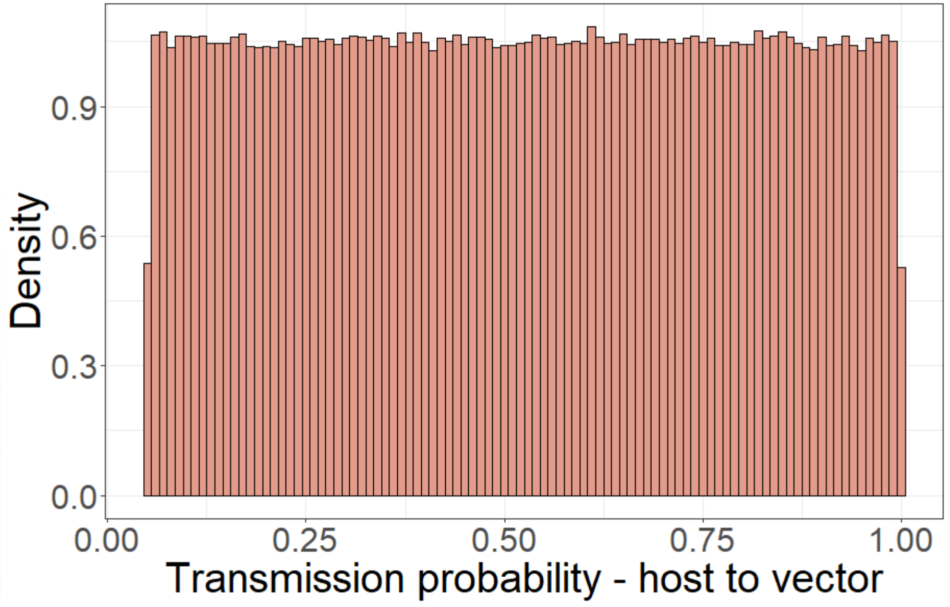
WAGENINGEN
UNIVERSITY & RESEARCH

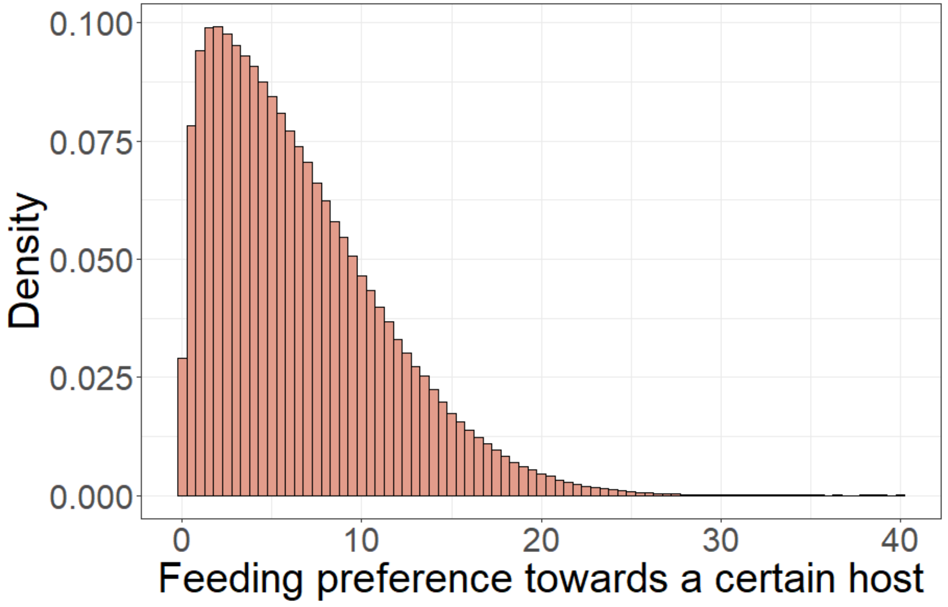
one health pact

Creation of the synthetic dataset

Parameters are randomly sampled from large distributions

(de Wit et al., 2024, Proceedings of the Royal Society B)





Confusion matrix

		Predicted label		
		Class A	Class B	Class C
True label	Class A	True A (TA)	False B (FBA)	False C (FCA)
	Class B	False A (FAB)	True B (TB)	False C (FCB)
	Class C	False A (FAC)	False B (FBC)	True C (TC)

Metrics

- $Accuracy = \frac{TA+TB+TC}{TA + FAB + FAC + FBA + TB + FBC + FCA + FCB + TC}$

- $PrecisionA = \frac{TA}{TA+FAB+FAC}$, total precision = average of precision A, B and C

- $RecallA = \frac{TA}{TA+FBA+FCA}$, total recall = average of recall A, B and C

- $F1\ score = 2\ \frac{Precision \times Recall}{Precision+Recall}$

Metrics

- $Accuracy$ = Ability of the model to classify correctly
- $PrecisionA$ = Ability of the model to identify all instances of A
- $RecallA$ = Ability of the model to classify instances of A correctly
- $F1\ score$ = Average of precision and recall

Linearsvm_ResNet	Precision	Recall	F1	Accuracy
Class A	0.848	0.837	0.842	0.842
Class B	0.799	0.806	0.802	0.802
Class C	0.834	0.837	0.836	0.836
Average	0.827	0.827	0.827	0.827
XGBoost_ResNet				
Class A	0.756	0.836	0.794	0.794
Class B	0.799	0.763	0.781	0.781
Class C	0.835	0.783	0.808	0.808
Average	0.796	0.794	0.794	0.794
svm_pca_ResNet				
Class A	0.797	0.877	0.835	0.835
Class B	0.841	0.783	0.811	0.811
Class C	0.852	0.826	0.839	0.839
Average	0.830	0.829	0.828	0.829

1D-CNN (datapoints)				
Class A	0.72	0.84	0.777	0.777
Class B	0.52	0.597	0.556	0.556
Class C	0.743	0.506	0.602	0.602
Average	0.661	0.649	0.645	0.649