I. Method

Since the downloaded data varies in resolution as well as coordinate reference systems (CRS), calibration is needed to be performed first, then assigning force of infection (FOI) values for each pixel. Finally, we use these pixels to predict FOI in other pixels in endemic regions by using Random Forest algorithm.

1. Pre-processing

All input maps are standardized as these following steps. Firstly, data will be re-projected to the same CRS which is a standardised equirectangular Plate Carrée projection under the World Geodetic System 1984. After that, maps having their resolution which is not 5km x 5km will be aggregated or disaggregated to this resolution by using bilinear regression for continuous data (bioclimatics, population density, ...) or nearest neighbor for categorical data (urban-rural covariate). Finally, all maps are cropped within the endemic regions. These operations are done by using *raster* and *rgdal* libraries in R.

2. Assigning FOI values

We create a FOI map in which a FOI value for each studied region are is modelled from its serology data by using the catalytics model. The map is then standardized as above steps. We use EM algorithm to assign FOI values for pixels [1]. They are subjected to that mean of FOI of pixels in a studied region is equal to the FOI in that region.

Perform EM

Based on [1], there are 2 ancillary variables needed to perform EM. To decide which variables are chosen, we first simply assign FOI values for all pixels so that a pixel will have a value of FOI of a studied region to which the pixel belongs. A pixel is called belongs to a studied region if its center is inside the region. This simple assigning can resolve a overlay problem. The overlay problem is the issue that there are some studied region are inside another region (such as China in Fig 3). Then we find the correlation matrix between FOI and other variables. We choose the feature having the highest absolute value of Pearson correlation coefficient to be 1 of 2 ancillary variables for EM. We also apply a random forest model to the above simple map to find which variable is the most important (has the most contribution) for the model. After finding 2 necessary features, we scale them to smaller range, then re-assign FOI values by using EM algorithm.

3. Random Forest

Random Forest is a machine learning technique which is usually used for both classification and regression. By combining many decision trees, it can avoid the overfitting issue and have better performance. Another advantage of this technique is that it can cope with the problem of outliers and assumption of input data. Moreover, random forest is also used to impute missing values.. We use randomforestSRC library in R for the imputation and scikit-learn library in Python for the regression.

Train – Validate – Test

Before running a model, entire data is divided into 3 subsets including training, validating and testing set. Training set is used to train a random forest model. We refine parameters which are used in the model by using validating set. Parameters are tuned until we achieve the highest performance on the validating set. Testing set is left intact and only used after we finish tuning parameter. Currently we have 2 types of dividing data into 3 subsets: based on pixels and based on regions. The ratio between these subset is 7 : 1.5 : 1.5.

Sampling by Grids

In the training set, we perform sampling by grids (choose pixels in some specific grids) to create multiple training subsets for multiple random forest models. This sampling way will reduce the spatial correlation in our model. We apply grids of 200km x 200km for training set, then randomly choose 70% of grids to train a single random forest model. This step is repeated 200 times to build 200 models. Final estimation is the average over 200 models. Fig 9 is the example of sampling by grids (each squared tile is one 200km x 200km grid)

Parameter

There are many adjustable parameter in a random forest. However, number of trees and number of features which will be used for a single tree are the most important parameters. We use the default number for both of them. Number of trees is recommended to be 500 and number of features is equals to a third of total features.

II. Result

1. Variables important

A random forest model also produce the variable important metric to measure which feature contributes the most for the model. After training, the model will randomly permute values in one feature and make the estimation again. Then the model measure the difference in performance between the original prediction and permuted prediction. The most important feature is the feature having the highest difference in performance. Fig 1 shows that Bio\_04 (The meaning of bioclimatic variable can be found at http://worldclim.org/bioclim) is the most critical feature.

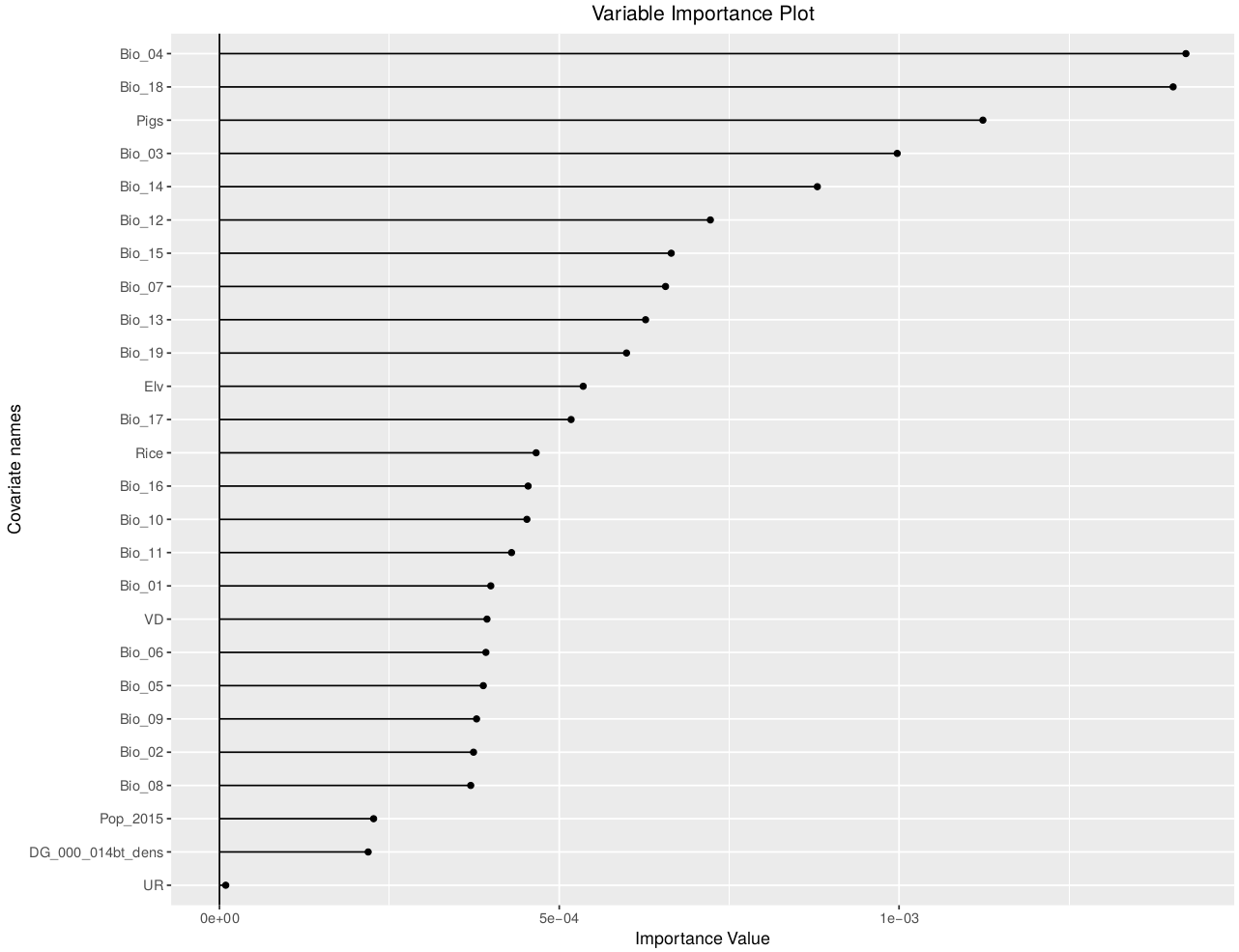


Figure 1. Plot of variable importance

2. EM assigning

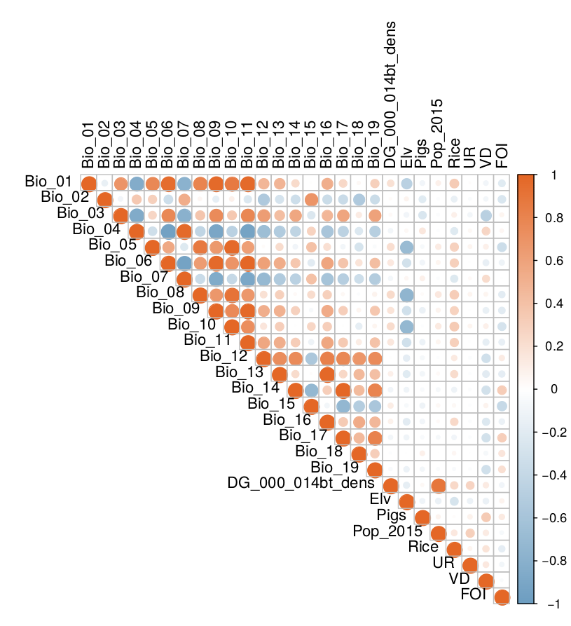
Fig 2 illustrates the correlation matrix between FOI and other covariates. The size of a circle shows how strong a relation between 2 variables. The color shows how variables affect to each other (blue is for negative relation, red is for positive relation). Bio\_15 is the most related to FOI values. Fig 3 and Fig 4 depict the simple map (pixels in the same region have the same value of FOI of that region) and EM disaggregated map. As can be seen on the graphs, China includes several overlay studied regions.

Figure 2. Correlation matrix

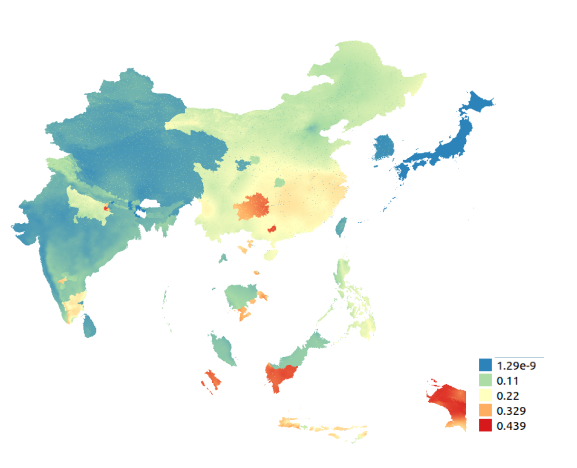


Figure 4. EM Assigning

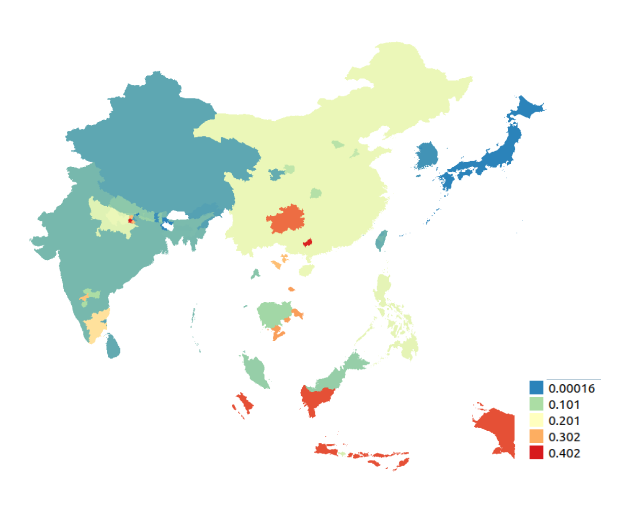


Figure 3. Simple Assigning

3. Mapping Result

We run our models in the scenarios of using only 19 bioclimatic features. Fig 5 and Fig 6 illustrate the results by sampling pixels and sampling regions. The results are quite similar in some areas.

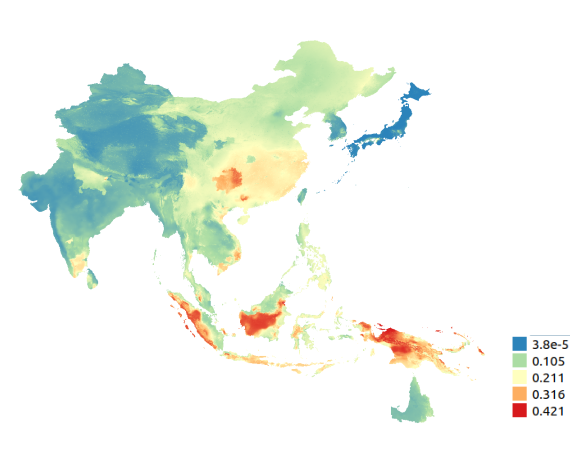


Figure 5. Result by sampling pixels

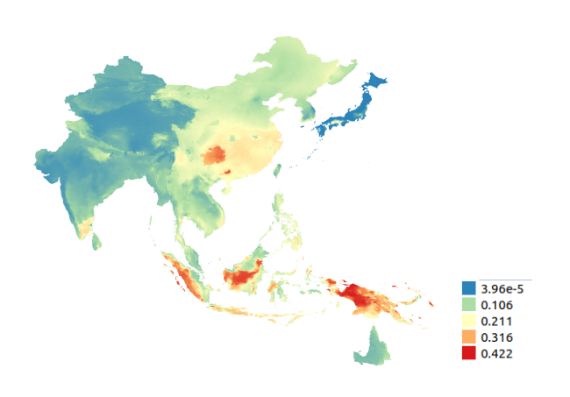


Figure 6. Result by sampling regions

4. Evaluation metrics

We decide to use R-squared to measure the accuracy of our models. Higher values mean better models and the maximum value of R-squared is 1. Fig shows the change in R-squared based on number of models. There is plateau in R-squared when the number of models is higher than 100.

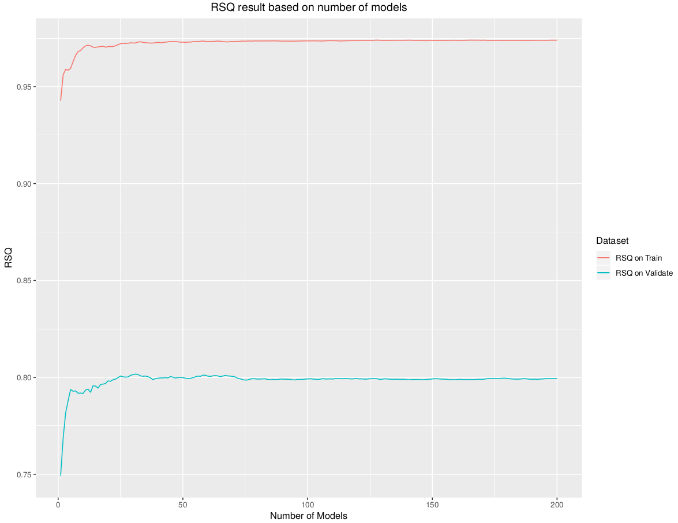


Figure 7. R-squared of the models based on number of trees

5. Imputation result

We create a NA-pseudo data by randomly assign NA for non-NA values of all features in some pixels. We use random forest to impute a NA-pseudo data again and calculate R-squared to measure the performance. We perform twice with 2 scenarios of the proportion of NA-pseudo data over the entire dataset: 20% and 50% (Fig 8).

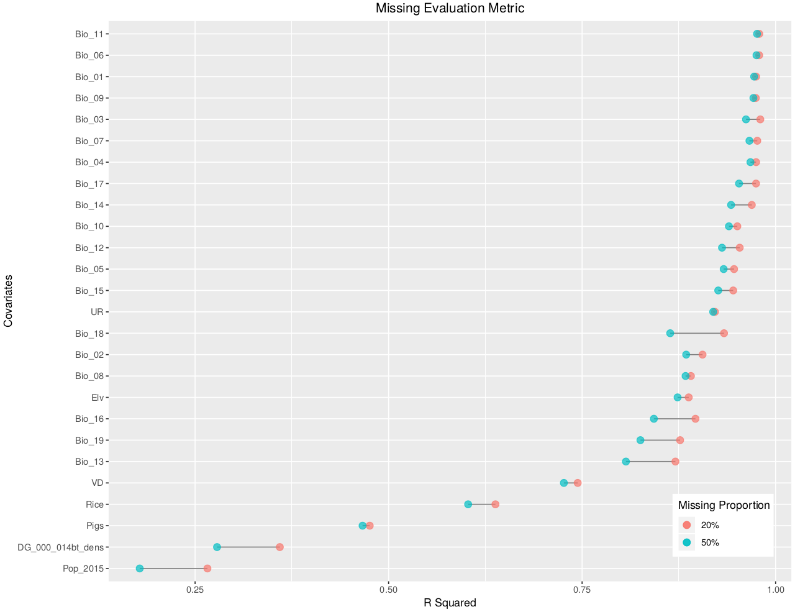


Figure 8. R-squared of estimation of paseudo-NA data in 2 scenarios.

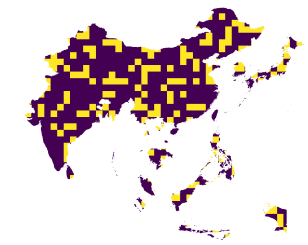


Figure 9. Example of Grids

Reference

[1] https://www.sciencedirect.com/science/article/pii/0098300495001123