Supplementary S2 Text: Instructions for S2 Code

S2 Code: Multinomial Regression

Section 1: Basic Data Preparation and Data Cleaning

In the original data, an older version of PRNT50 assays performed on DENV1 was recorded as “D1-VN” and was replaced by “D1”, so it should be excluded from the “PRNT50” column. Moreover, unexpected values in “Serotype” and “PRNT50” columns were also excluded from our analysis. The formats of the test results from MAC-ELISA and GAC-ELISA are unified. These processes are commented on in the code since they are already excluded from the generating process of fake data.

The data is imported with check.names = FALSE since the synthesis data follows the column names in the original data which have illegal symbols. Missing values in “Gender”, “Age_clean” and “OnsetYear” are removed. The *clean_prnt50*, *check_PRNT50_match* and *update_wrong_PRNT50* functions are created for the original data set to make sure the serotypes in the “PRNT50” column are correctly recorded.

Section 2: Update “PRNT50” Column based on the thresholds

The *replace_with_na* and *update_prnt50_colum*n functions help to update the data based on the pre-defined lower and upper thresholds. The sensitivity analysis of the thresholds is based on this process, remember to change the threshold_low and threshold_high accordingly and re-run the code from the beginning every time after changing thresholds.

Section 3: Differentiate Current and Past Infections

After excluding co-infections, the basic features of the data are summarised in *summary_table_long*. The “PRNT50” column is classified by *classify_PRNT50* function. “Current” and “Past” columns are created to store current and past infection serotypes. Groups 1-8 in Table 2 used to impute these values are defined in this section and dengue_summary counts the number of individuals in each group.

Section 4: Data Preparation for multinomial regression

To run the multinomial regression, the categorical variables are releveled using *update_factors* to make the reference level more meaningful. The PRNT50 DENV1-4 values are classified into three levels and the levels associated with the past infections are extracted based on the "Past" column by *process_PRNT_data* function.

Note that before running the analysis, make sure the threshold_low and threshold_high are the same as what you expected and the pre-processing of the data (Section 1-4) should performed every time before changing thresholds. Remember to change the name of the model accordingly within the bootstrapping algorithm as well. The AIC is much higher than it resulted from the original data due to the lack of inherent structure.

Analysis 1: Multinomial regression using PCR+ cases v.s PCR- & IgM non-cases

This is the analysis using the baseline data set where all the cases are confirmed by the WHO definition (G1-G2). We compared different combinations of model covariates using this data set and selected the one with the smallest AIC, Model 6. We also visualise and analyse the data excluding those cases that tested positive for PCR but with a notable antibody titer.

Analysis 2: Multinomial regression using data excluding G9

This is the analysis using data categorized by S1-S8 which has more imputed observations. The process of building multinomial regression followed analysis 1.

Analysis 3: Multinomial regression using all the data

To impute the current infection based on G9.1, we used the distribution of the homologous PRNT50 values when PCR is positive to identify the common thresholds for each serotype. The cloest_serotype_func returns the serotype that is most likely to be categorized as a current infection under G9.1. Then, the data was fitted to the model following the previous process.

The bootstrapping method for Analyses 1-3 is identical which is coded under Analysis 3. Just change the model names (multi_modeli_low_high, where i refers to analysis i to run the bootstrapping for each analysis. The resulting bootstrapping estimates and confidence intervals (CI) are structured in a standard format: estimates (lower CI, upper CI) and the *result_in_CI* function help check whether the estimates are within the CI.

Analysis 4: Use all the data from G1-G8 with 20% individuals from G9.2 using EM algorithm

We select 20% of those individuals for analysis to avoid too over-imputed data. The bootstrapping method is a bit different from the first three analyses, so a new bootstrapping section is coded in this section. The format of the resulting estimates and the way to check whether the CI includes the estimate is still the same as before. The convergence of the EM algorithm depends on the sampled data, so it is not unusual for some iterations that fail to converge, especially in the case of synthesis data where the underlying patterns may not exist. *log_likelihood* function is therefore defined to choose the set of parameters with the highest log-likelihood under this situation.

Section 5: Data Visualization

The way to visualise the data is identical for all the analyses, just change the data set names accordingly: DENV_analysisi_low_high, where i refers to analysis i using data set i.

- Plot 1a (bar): visualizes age distribution by serotype
- Plot 1b (bar): visualizes gender distribution by serotype
- Plot 2 (line + bar): visualizes yearly dengue infection counts and percentages by serotype
- Plot 3 (violin): visualizes the distribution of PRNT50 values by serotype
- Plot 4 (forest): visualizes the odds ratios with confidence intervals
- Plot 5 (bar): Distribution between examination date and onset date
- Plot 6 (line): Cumulative distribution plots for PRNT50 values