STA363 Project 1

Zishan Shao

2022-09-05

Abstract

Dengue is a viral disease spreading worldwide. It could cause acute symptoms and irreversible damage to personal health without medical care. Even worse, dengue is difficult to diagnose without accession to advanced medicare. Therefore, it is necessary to develop an efficient model that could relatively accurately diagnose the disease with suspicious symptoms. The main goal of this report is to create a model that accurately predicts (by classifying) dengue patients by evaluating clinical data collected from Vietnam hospitals. Considering the nature of the response variable, we decided to choose between the K-Nearest Neighbor (KNN) based model and the multivariable logistic regression model to fulfill the task. To determine the prediction capacity of the two models, we used accuracy, sensitivity, and specificity as evaluation metrics to make the choice. Finally, we compare three metrics of KNN based model and the multivariable logistic regression model and decided to choose the multivariable logistic regression model (Model 2) as our final model from the statistical and practical considerations. The model has 74.1% of accuracy, 90.9% of sensitivity, and 34.7% of specificity.

Contents

Section 1: Introduction and Data

In this report, we are working with patient data from the local hospitals in Vietnam. The data was composed of 5726 children who have symptoms of dengue fever when they were brought to clinics. Dengue was *a viral infection transmitted to humans through the bite of infected mosquitoes with dengue virus (DENV)*. While most of them only develop a mild illness, some develop into a potentially lethal complication, called severe dengue [1]. The study of dengue has long been conducted worldwide, but detection method without the accession of advanced medical care is yet to be invented. This caused difficulty in the diagnosis of dengue in underdeveloped regions. Therefore, the main goal of this analysis is to construct a relatively applicable statistical model that efficiently, yet accurately, predicts dengue with common symptoms of the disease. The model will be built with features that have the highest potential influence on the response variable (label) and trained with the data from real-life data. The model is expected to be applicable in most places in the world even without the accession of advanced medication, so the simplicity of the model will also be considered. Finally, we expected to select the best model from multiple trained models with different tuning parameters, thresholds, and sampling techniques. We will evaluate our model based on precision metrics of statistical models (e.g. accuracy, sensitivity, specificity).

Label Analysis & Data Cleaning

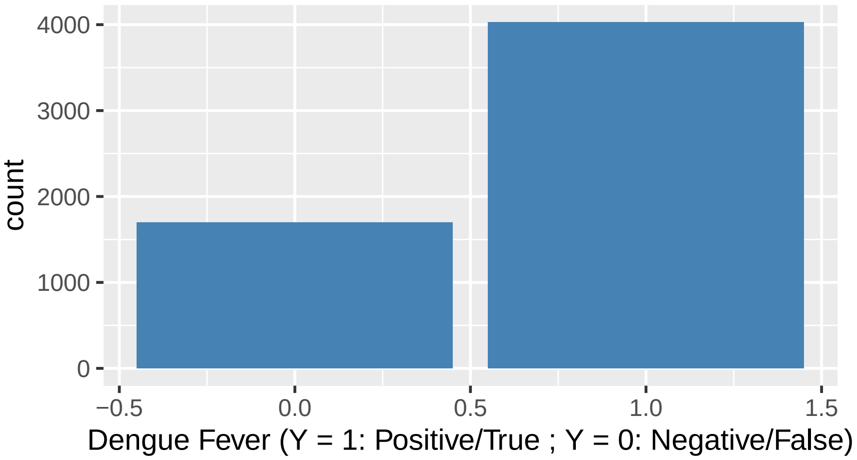

Figure 1.1: Barplot of Label Y

In this data, there are a total of 15 variables comprised of 14 features (categorical or numerical) and 1 label. Our goal is to predict the label Y according to features with different weights. Meanwhile, the Y is a binary variable: Y = 1 means the patient has dengue fever and Y = 0 means the patient does not have dengue fever. In other words, this report is dealing with a prediction task with a classification problem. From Figure 1.1, we could see that the distribution of the outcomes (0 & 1) is about 1: 2.5 in ratio, which is valid for model building because there is a fair chance to meet each scenario.

Table 1: Data Cleaning

	Num Observations
Before Cleaning	5726
After Cleaning	5726

[1] "Dengue and Severe Dengue." World Health Organization, World Health Organization, https://www.who.int/news-room/fact-sheets/detail/dengue-and-severe-dengue.

Along with the label analysis, it is also necessary to clean the data by removing or refilling the rows with missing information. However, the cleaning results indicate that the total observation number remains unchanged after the cleaning (both equal to 5726), indicating that there is no missing data existed in the explanatory variables. This also indicates there is no missed information in any features (indicated by columns), so the dimension of the entire data set remains same. Therefore, we proceed to construct the model.

Section 2: K-Nearest Neighbor (KNN)

Because the label Y is a binary variable, there are two kinds of ubiquitous methods to apply: multivariable logistic regression model (MLR) and K-Nearest Neighbor (KNN). The KNN is a clustering technique/algorithm that is effective in classifying labels based on the neighboring outcomes. The KNN assumes that similarity exists among outcomes of nearby observations, and is effective if there is a clustering of outcomes in the data. Therefore, KNN is expected to be suitable because the diagnosis of dengue is difficult without the accession of the standard test.

2.1: Brief Example of K-Nearest Neighbor (KNN)

KNN algorithm is surprisingly simple for interpretation. In the following example, a mock sample was provided with age = 8 and height = 100 cm, and a KNN-based model was constructed with the feature Age and Height. We set the tuning parameter k, the number of neighbors for consideration, equal to 5 and make predictions with the model and a scatterplot.

Prediction with Scatterplot & Model

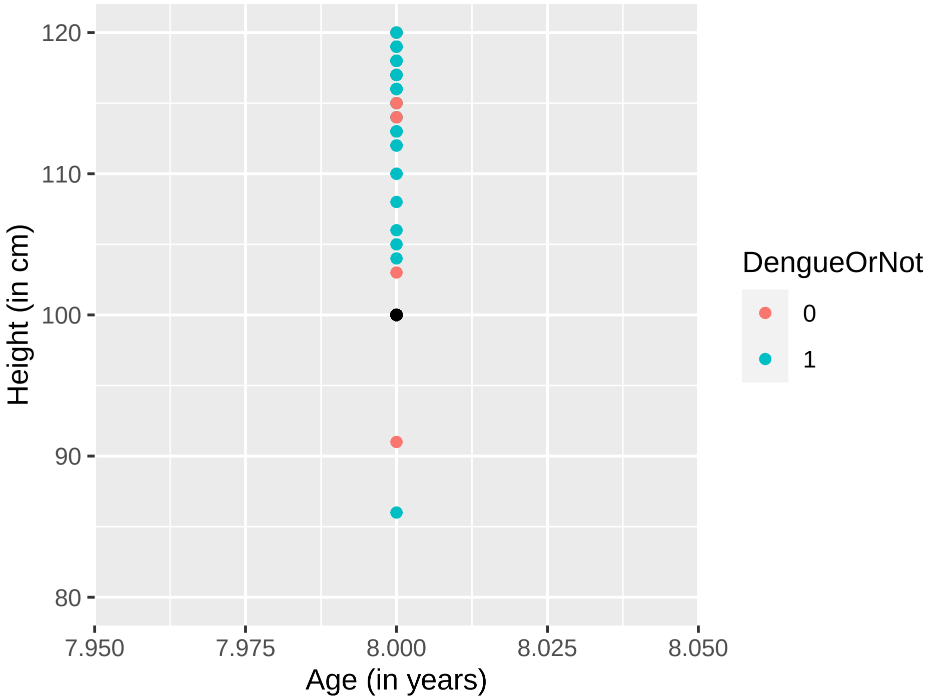

Figure 2.1.1: Scatterplot of Age & Height (Age = 8)

Figure 2.1.1 provides all observations with age = 8. The orange dots indicate that the patient is negative and the cyan dots indicates the patient is positive. The black point, which is the mock observation, lies at 100 cm in height. The closest 5 observations to the mock observation have the outcome of [0,1,1,1,1], in which the positive result was the mode of the neighbors. Therefore, it is likely that, by assumption, the mock observation also has a positive outcome.

The model prediction also supports our prediction with the scatterplot. The predicted label of the mock observation indicates that the result should be Y = 1, which means the kid gets dengue fever.

However, the prediction result does not necessarily reflect the accuracy of the model. To determine if this model is effective, we need to compare the performance metrics such as geometric means of KNN models adapting different sampling methods or tuning parameters. This part will not be specified in the example but will be included and explained later.

2.2: KNN models (with all features) & train-test split method

In section 2.2, we will construct our first model with all 14 features via the train-test split method. The train-test split method, by its name, splits the data into training and testing datasets. It is necessary when the testing dataset was not provided and was effective in minimizing the overfitting problem. In this case, I will compare the traditional train-test split by 20% to 80% split and k-fold cross-validation to construct the train-test dataset.

The tuning parameter is also considered. The k is the tuning parameter of every KNN-based model, which defines the number of neighbors to be considered. This parameter enables the KNN algorithm to adapt to different scenarios. For instance, if the sample size was very small, the neighbor number should be comparably smaller than the neighbor numbers in relatively larger datasets. In this case, I will construct models with k = 3, 5, 7, 9 and compare their geometric means (computed by the square root of sensitivity x specificity). The model with the highest geometric mean will be selected.

Train-Test Spliting

Table 2: Observations in Train/Test Sets & True/False Counts

	Count
Num observations in test set	1145
Num observations in training set	4581
True Dengue Number in test set	803
True Not Dengue Number in test set	342

The training set has 4581 observations and the test set has 1145 observations, which in total has 5726 observations. The test set includes 803 patients who were diagnosed with dengue and 342 patients who do not diagnose with dengue. The later two numbers are essential for the computation of geometric means.

Model Creation (via loop)

To create and compare models with different tuning parameters, a loop is adapted to iteratively train and test the model. The predicted result was recorded in a 4x2 data frame with a value of k and a corresponding geometric mean (GMean) in table 3.

Table 3: Models with different k values with train-test split method

K	Geometric Mean
3	0.526
5	0.554
7	0.540
9	0.548

Table 3 indicates that k = 5 has the highest geometric mean, indicating that, at k = 5, the model classifies the observations at the highest precision because this means we have the highest proportion of correctly classified positive and negative cases. Therefore, the **k = 5 should be the best choice** from the KNN model built with the train-test split method.

Estimation Variation Test

It is worth mentioning that the train-test split method has two drawbacks. Firstly, it could cause a reduction in training data size by splitting 20% of it into testing data. Such a large proportion of data loss could potentially cause inaccuracy in prediction. Secondly, the randomly chosen rows could cause high estimation variation because the training results are highly dependent on the chosen observations, especially for data sets with a limited number of observations. Therefore, it is necessary to test if an alternative sample could lead to a different conclusion.

Table 4: Models with Resampled Data

K	Geometric Mean
3	0.539
5	0.555
7	0.543
9	0.556

Under seed = 1919810 (previously 114514), the k = 9 obtains a geometric mean equals 0.556, while the k = 5 is 0.555 with 0.001 lower than k = 9. Nonetheless, the closeness of the two geometric means failed to indicate a statistically significant difference in prediction capacity. Meanwhile, the value of the geometric mean of k = 5 only varies by 0.001 when seed = 114514. Therefore, it is proven that the model with k = 5 is not significantly sensitive to the observations chosen.

Confusion Matrix of KNN (train-test split method) & Metrics Computation

Table 5: Confusion Matrix of KNN

	True Not Dengue	True Dengue
0	120	109
1	220	696

Metric Computation

$$Sensitivity = \frac{TrueDengue \text{ \& } PredictedTrueDengue}{Total\ TrueDengue} = \frac{696}{696 + 109} \approx 0.865$$

$$Specificity = \frac{TrueNotDengue \ \& \ Predicted \ TrueNotDengue}{Total \ TrueNotDengue} = \frac{120}{120+220} \approx 0.353$$

$$Accuracy = \frac{Total \ Correct \ Predictions}{Total \ Observations} = \frac{696+120}{120+220+696+109} \approx 0.713$$

2.3: KNN Models (with All Features) & 10-Fold Cross Validation

In section 2.3, we applied the 10-fold cross-validation method as the sampling method to obtain the training and testing data. The k-fold cross-validation, like the train-test split, was applied when testing data was not provided. K-fold cross-validation splits the data into k folds and uses each fold's data iteratively as the testing data (the rest data as training data). The results of the prediction will be computed each time with the size of a fold, and finally, the size equals 5726.

The two most common fold numbers are 5 and 10. Considering the size of the data, the 10-fold cross-validation should be a more appropriate choice. The dataset was not exactly divisible by 10, so we will create a folder with larger storage space and balance the number of observations in each fold with a difference no larger than 2. This could be achieved by taking the ceiling of the data assigned to folds so that all observations will be sampled to folds.

Table 6: Models with different k values with 10-Fold Cross Validation

K	Geometric Mean
3	0.533
5	0.529
7	0.523
9	0.538

The testing result in table 6 indicates that the k = 9 with 10-fold cross-validation has the highest geometric mean (0.538), which means that the KNN-based model with k = 9 achieves the highest precision under the 10-fold cross-validation method.

Confusion Matrix of KNN (10-fold cross validation method)

According to the previous result, we construct our model with k = 9 and fold number = 10. We then construct the confusion matrix of the model and compute the metrics to evaluate the prediction capacity of the model.

Table 7: Confusion matrix (fold=10) of DengueClean

True Not Dengue	True Dengue
525	508
1173	3520

Metrics Computation

$$Sensitivity = \frac{TrueDengue \ \& \ PredictedTrueDengue}{Total \ TrueDengue} = \frac{3520}{3520+508} \approx 0.874$$

$$Specificity = \frac{TrueNotDengue\ \&\ Predicted\ TrueNotDengue}{Total\ TrueNotDengue} = \frac{525}{525 + 1173} \approx 0.309$$

$$Accuracy = \frac{Total\ Correct\ Predictions}{Total\ Observations} = \frac{525 + 3520}{525 + 3520 + 508 + 1173} \approx 0.706$$

General Table of Metrics

	Geometric Mean	Sensitivity	Specificity	Accuracy
train-test	0.554	86.5%	35.3%	71.3%
10-fold	0.538	87.4%	30.9%	70.6%

The general table of metrics indicates that the train-test split model has higher geometric mean, specificity, and accuracy, while the 10-fold method has higher sensitivity. Therefore, the KNN-based model with the train-test split method and k = 5 should be our final choice as it has better performance on most metrics.

The reason behind this result makes sense because the data set is not limited in size. With 5726 observations, it turns out that losing 20% of the data does not harm the prediction capacity of the model or cause a very large estimation variation. What's more, the estimation variation test also indicates that the prediction capacity of the model with k = 5 and the train-test split method is not overly sensitive to the observation selections. Considering that the 10-fold cross-validation method is computationally expensive, the choice of model with k = 5 and the train-test split method makes sense.

Section 3: Multivariable Logistic Regression

Noticing that the label Y (Dengue Fever) of the dataset is binary, we could also apply the multivariable logistic regression model in this case. The multivariable logistic regression model is a statistical model that conducts binary classification. It assumes that the binary outcome of a sample follows a Bernoulli distribution and has a population model as follows:

$$Y_i \sim Bernoulli(\pi_i)$$

where in this case

$$\pi_i = P(Y_i = 1|X_i)$$

$$log(\frac{\pi_i}{1-\pi_i}) = \beta_0 + \beta_i X_i, \quad i \in \{1, 2, ..., n\}$$

n is the number of features chosen to be used in the model

In a logistic regression model, we first conduct exploratory data analysis to select features that have visually statistically significant influences on the outcome. Then we construct the model with the features and get rid of variables with p-values larger than 0.05, which means we failed to prove there is a statistically significant relationship between the feature and the label.

3.1 Exploratory Data Analysis (EDA)

The exploratory data analysis provides researchers a chance to explore the relationship between individual features with the label. The logistic regression model requires the feature to be linearly related to the label, which can be explored by the empirical logit plot. If there is a visually non-linear relationship between the feature and the label, we then need to conduct the linear transformation (usually convert the feature to the log of the feature) to fix the non-linearity. However, if the non-linearity exists after the transformation, we could decide to choose another transformation function or choose not to use the feature.

EDA (Part 1): Variable Type Analysis & Standardization

It is worth mentioning that the linear relationship is only required for numerical features, while the categorical features need mosaic plot to determine their relationship with the model. Therefore, we need to determine the type of the features and convert the defined-numerical features to categorical if they perform more similar to a categorical feature and vice verso.

Table 9: Feature Range

feature	range begin	range end
Sex	1.000000	2.00000
Age	1.000000	15.00000
DayDisease	1.000000	3.00000
Vomiting	1.000000	2.00000
Abdo	1.000000	2.00000
Muco	1.000000	2.00000
Skin	1.000000	2.00000
Temp	35.800000	41.00000
BMI	8.503401	34.70986
Height	58.000000	176.00000
Weight	7.200000	91.00000
Flush	0.000000	1.00000
Hepatomegaly	0.000000	1.00000
Rash	0.000000	1.00000

From the feature range table, we noticed that, in the description of features, DayDisease is defined as a numerical rather than a categorical variable. Therefore, we need to convert it to a categorical variable.

EDA (Part 2): Empirical Logit Plots for Categorical Features

From the feature range of the data, we could see that sex, DayDisease, vomiting, Abdo, Muco, Skin, Flush, Hepatomegaly, and Rash all have only 2 or 3 categories, indicating that these are categorical variables. Therefore, the 5 remaining features are numerical: Age, Temp, BMI, Height, and Weight. We need to construct empirical logit plots for all of these features and explore if there is non-linearity existed.

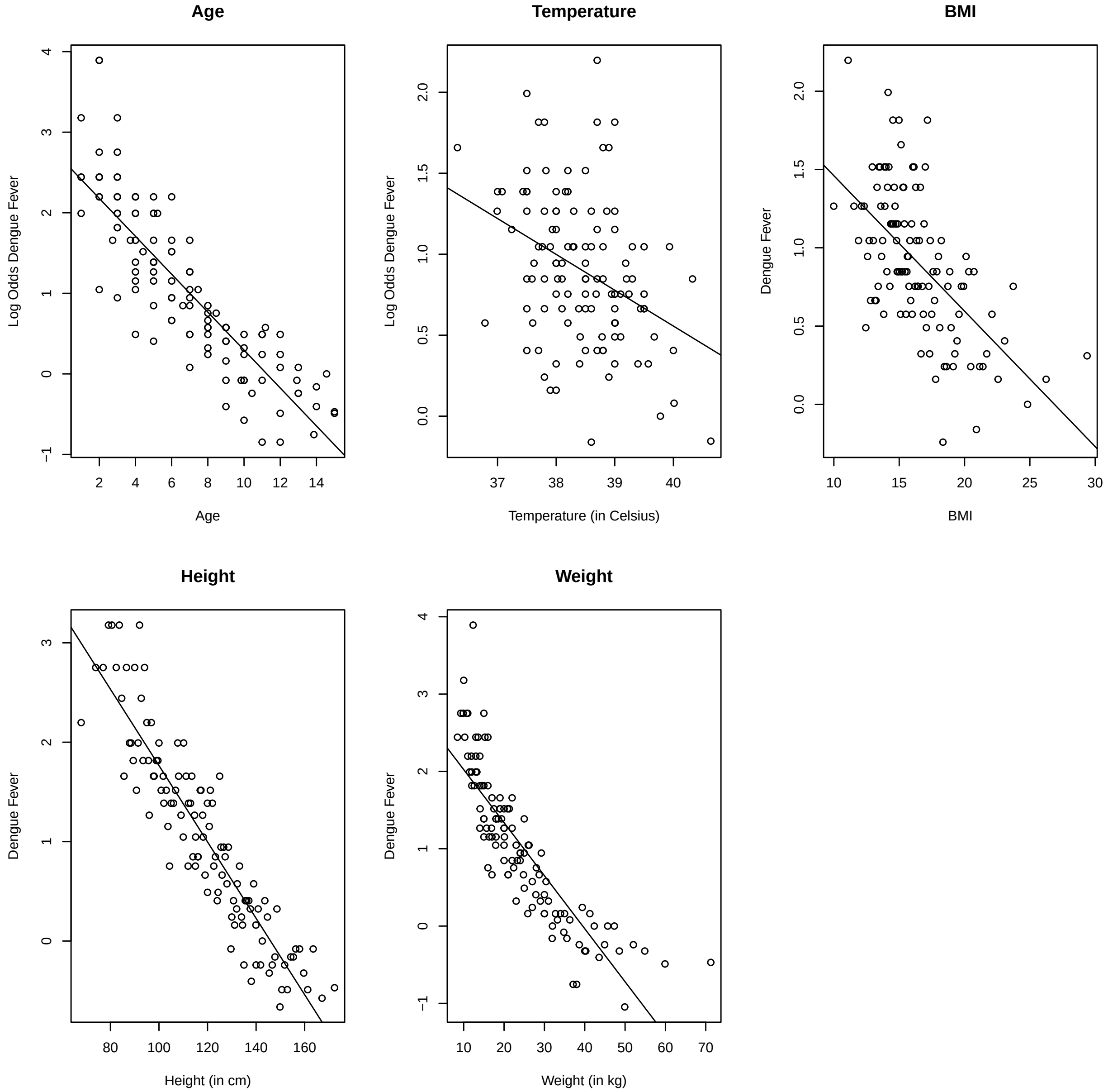

Figure 3.1.1: Relationship Analysis of Numerical Variables vs Log Odds Dengue Fever

Figure 3.1.1 shows that age, temperature, BMI, and height have a strong linear relationship with the log odds of dengue fever, indicating that these four features should be added to the model with no need for transformation.

However, the empirical logit plot between the log odds of dengue fever and the weight reflects a curved relationship, which is not linear, so it is necessary to perform the linear transformation to modify the relationship.

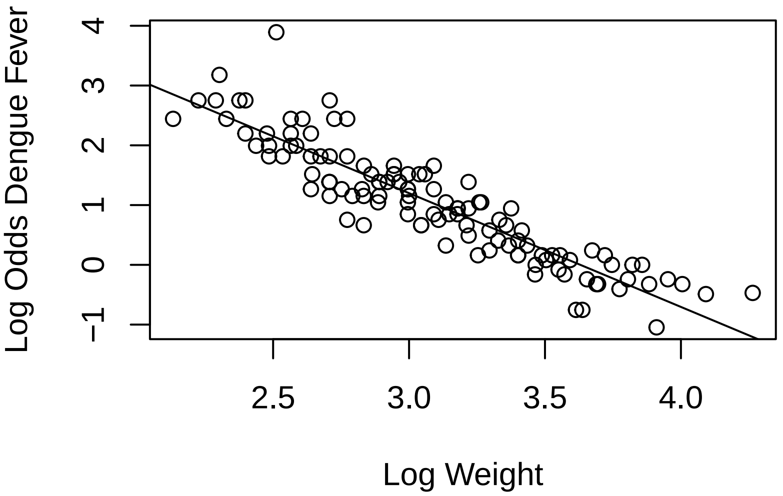

Figure 3.1.2: Log Weight vs Log Odds Dengue Fever

Figure 3.1.2 indicates that, the log weight is strong linearly related with the log odds of dengue fever. Therefore, we add the log weight into our model.

EDA (Part 3): Multicollinearity

Besides checking the relationship between numerical features and the label, it is important to check if there is a linear relationship exists between features. Therefore, we use a correlation matrix to explore linear relationships between numerical features.

Table 10: Correlation Matrix of Numerical Variables

	Age	Temp	BMI	Height	Weight
Age	1.0000000	0.0117088	0.1708381	0.9445484	0.8469863
Temp	0.0117088	1.0000000	0.0408907	0.0229824	0.0335897
BMI	0.1708381	0.0408907	1.0000000	0.1493719	0.5608645
Height	0.9445484	0.0229824	0.1493719	1.0000000	0.8828541
Weight	0.8469863	0.0335897	0.5608645	0.8828541	1.0000000

From the table, Height and Weight are strongly related to Age with a correlation larger than 0.8. The reason behind this is that the patient's age was no greater than 15 years old, so children's age is a direct indicator of their physique as they grow fast. Therefore, we could keep age as the only indicator in the model and remove the height and weight, which are redundant.

EDA (Part 4): Categorical variables analysis

Along with the numerical variables, we use mosaic plot to select categorical features with statistically significant influences of the outcome.

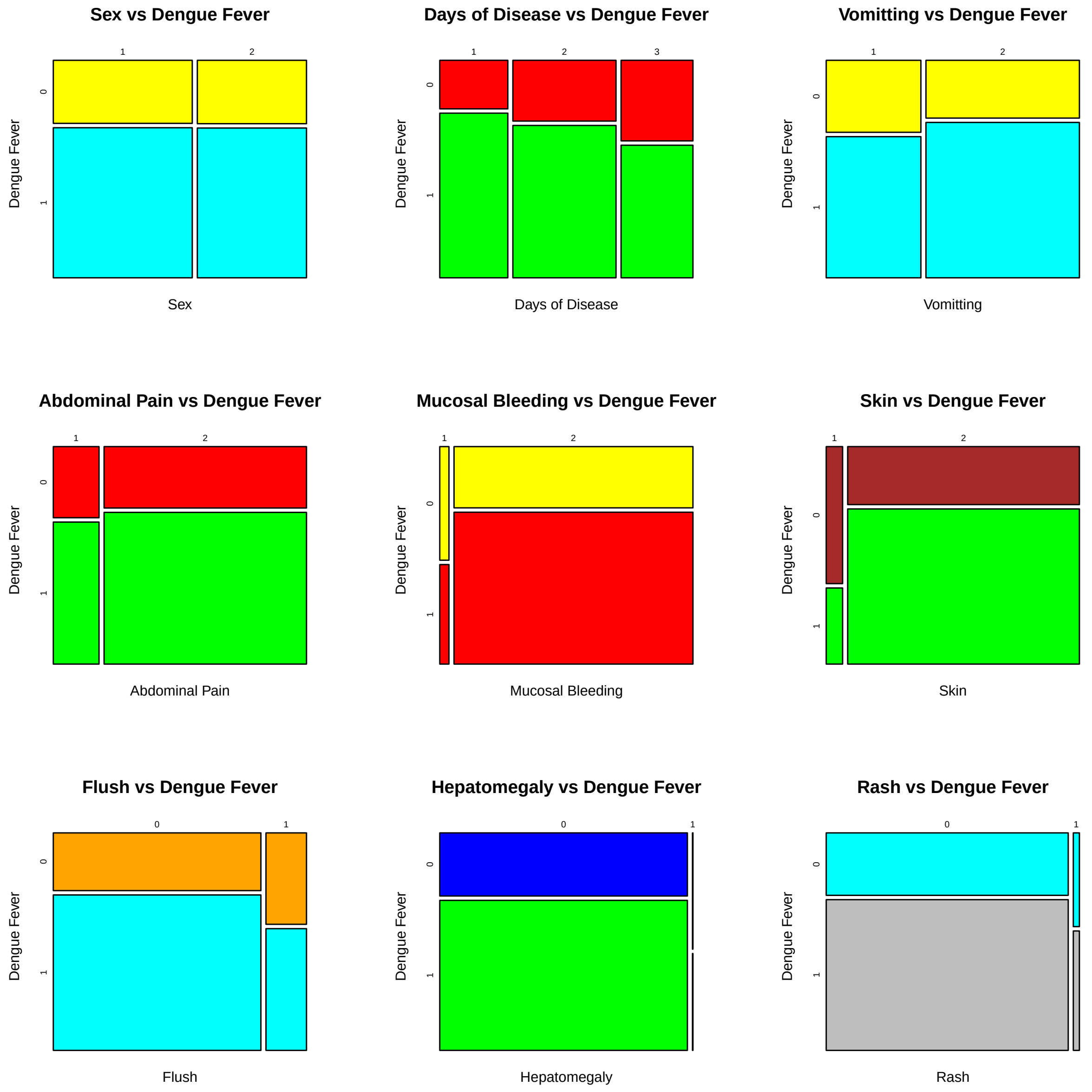

Figure 3.1.3: Relationship Analysis of Categorical Variables and Dengue Fever

From Figure 3.1.3, we see that there is no relationship between sex and dengue fever. Although there is an evident change in the proportion of outcomes with different categories for hepatomegaly, one category dominates the entire feature and is therefore unlikely to have statistically significant influence on the outcome. Therefore, both sex and hepatomegaly are excluded from the model.

3.2: Model building and training

In section 3.2, we will construct model with the variables selected from the EDA: DayDisease, Vomiting, Abdo, Muco, Skin, Flush, Rash, Age, Temp, and BMI. We applied the train-test split method in this case to explore the prediction capacity of the model.

Model 1 Construction & Coefficients Evaluation

Table 11: Coefficients Table of Logistic Model 1

	Estimate	Std. Error	z value	Pr(>|z|)
(Intercept)	8.433	1.656	5.093	0.000
DayDisease2	-0.304	0.081	-3.752	0.000
DayDisease3	-0.582	0.086	-6.767	0.000
Vomiting	0.308	0.067	4.602	0.000
Abdo	0.086	0.083	1.036	0.300
Muco	0.166	0.177	0.935	0.350
Skin	1.283	0.133	9.648	0.000
FlushTRUE	-0.617	0.085	-7.261	0.000
RashTRUE	-0.339	0.203	-1.665	0.096
Age	-0.213	0.009	-22.879	0.000
Temp	-0.221	0.042	-5.201	0.000
BMI	-0.033	0.010	-3.198	0.001

From the critical test, we could see that the P-values for Abdo, Muco, and RashTRUE are all over 0.05. This means that given all features are added to the model, based on 95% of confidence, we failed to testify that there is a statistically significant relationship existed between these features and the response variable. Therefore, I recommend removing these variables from the model. If the remaining variable still performs relatively same level of prediction capacity, then we should adopt the simpler model.

Model 2 Construction & Coefficients Evaluation

Table 12: Coefficients Table of Logistic Model 2

	Estimate	Std. Error	z value	Pr(>|z|)
(Intercept)	8.476	1.643	5.158	0.000
DayDisease2	-0.309	0.081	-3.817	0.000
DayDisease3	-0.592	0.086	-6.901	0.000
Vomiting	0.321	0.065	4.924	0.000
Skin	1.356	0.123	11.059	0.000
FlushTRUE	-0.643	0.084	-7.672	0.000
Age	-0.212	0.009	-22.877	0.000
Temp	-0.213	0.042	-5.055	0.000
BMI	-0.033	0.010	-3.220	0.001

From the table, all features in model 2 has p-value less than 0.05, indicating that there is a statistical significant relationship between each feature and the label dengue fever.

AIC Table of Model 1 & Model 2

Table 13: AIC Table of M1 & M2

	AIC values
Model 1	5942.354
Model 2	5941.160

From the table, the Model 2 has smaller AIC than the model 1. This indicates that, after removing all these variables, the performance of the model 2 better fits the data. Therefore, we should select model 2 as the candidate.

3.3: Prediction & Result Evaluation

In KNN section, our result proves that the data size is large enough that the influence from the selection of observations does not influence the prediction capacity of the model. Therefore, we also apply the train-test split method in logistic model and make predictions with threshold of 0.5.

Model 2 predicts 209 patients who are not dengue and 936 person who is dengue. The total prediction numbers equal the observations in the test dataset. To determine the accuracy of the prediction, we construct the confusion matrix for analysis.

Confusion Matrix of Logistic Regression Model (Model 2)

Table 14: Confusion Matrix of Logistic Regression

	True Not Dengue	True Dengue
0	123	86
1	219	717

Metrics Computation

$$Sensitivity = \frac{TrueDengue\ \&\ PredictedTrueDengue}{Total\ TrueDengue} = \frac{717}{717+86} \approx 0.892$$

$$Specificity = \frac{TrueNotDengue\ \&\ Predicted\ TrueNotDengue}{Total\ TrueNotDengue} = \frac{123}{123+219} \approx 0.360$$

$$Accuracy = \frac{Total\ Correct\ Predictions}{Total\ Observations} = \frac{717+123}{717+123+86+219} \approx 0.734$$

Section 4: Compare and Evaluate: Logistic/KNN

Summary Table

Summary Table of Metrics (Logistic vs KNN)

	Sensitivity	Specificity	Accuracy
KNN	86.5%	35.3%	71.3%
Logistic	89.2%	36.0%	73.4%

Conclusion

From the summary table, the logistic regression model outperformed the KNN-based model in all three metrics in the summary table. Therefore, we could conclude that the multivariable logistic regression model is more effective in predicting dengue fever through the symptoms, and Model 2 should be adapted as our final choice.

The logistic regression model is also more effective for the prediction of higher-dimension data compared with KNN. Without the help of the computer, client could compute the log odds of the dengue by hand and make predictions based on the threshold, while the KNN-based model examines the clusters in high dimension, which is difficult for visualization or computation. In underdeveloped areas, logistic regression is a better choice because advanced computational tools are unlikely to be available. Therefore, for practical consideration, logistic regression is also a better choice for clients.