Predicting rice loss due to typhoons in the Philippines

Selma Boeke
STUDENT NUMBER: 2031198

THESIS SUBMITTED IN PARTIAL FULFILLMENT
OF THE REQUIREMENTS FOR THE DEGREE OF
MASTER OF SCIENCE IN DATA SCIENCE: BUSINESS AND GOVERNANCE
DEPARTMENT OF COGNITIVE SCIENCE & ARTIFICIAL INTELLIGENCE
SCHOOL OF HUMANITIES AND DIGITAL SCIENCES
TILBURG UNIVERSITY

Thesis committee:

Maryam Alimardani
Menno van Zaanen

External advisor 510:

Marc van den Homberg

Tilburg University
School of Humanities and Digital Sciences
Department of Cognitive Science & Artificial Intelligence
Tilburg, The Netherlands
June 2019

Preface

I would like to express my gratitude to my supervisor Maryam Alimardani for the useful comments and remarks through the learning process of this master thesis. I also want to thank Menno van Zaanen for his recommendations on how to place my research in the bigger picture. Furthermore, I would like to thank Marc van den Homberg of 510 for introducing me to the topic as well for the support on the way. Finally, I like to thank Jannis Visser and Aklilu Teklesadik of 510 for their guidance. They were of great help with obtaining data and resolving modeling obstacles.

Selma Boeke, 2019

Predicting rice loss due to typhoons in the Philippines

Selma Boeke

In this work, both regression and binary classification algorithms were trained to predict rice loss in the Philippines due to typhoons. In addition, feature selection methods were used to find the most important explanatory features for this task. This study has been carried out because it is expected that a lot of damages that farmers encounter as a result of these typhoons can be saved when preventative actions are taken. The Philippines is extremely vulnerable to typhoons and farmers are often affected by these hazards. Unlike previous studies, the goal of this study was to predict the impact of the typhoon a couple of days before it makes landfall. The model was on province-level and could help humanitarian organizations and the Department of Agriculture to decide where to to take preventative action. The models were trained on data from 11 typhoons that occurred in the past 8 years in the Philippines. Both geographical data from every province, and typhoon specific features were used as input. The rice loss that these typhoons caused was considered as the output. The average percentage of lost rice area amounted 7.1%. As for the regression task, the support vector regressor performed best with a Mean Absolute Error of 6.83 percentage points. For the classification model, thresholds of 20%, 30% and 40% were tested in order to find the best performing model. These thresholds represent the critical level of lost rice fields that could be considered for triggering anticipatory action towards farmers. The binary classifiers were trained to increase its ability to rightly predict the positive samples. In all three cases, the support vector classifier performed best with a recall score of 88%, 75% and 81.82%, respectively. However, the precision score for each of these models was low: 17.05%, 14.46% and 10.84%, respectively. For both the support vector regressor and classifier, of all 14 available input features, only wind speed was selected as explanatory feature. Yet, for the other algorithms that were trained in this study, other sets of features were selected. Moreover, for some algorithms the feature selection was sensitive to the hyperparameter settings of that algorithm. This variation in selected feature sets as well as the imprecise predictions are consequences of the small dataset that was used for this study. It is therefore important that more data are gathered in order to make more robust and accurate predictions. Also, if there can be damage data collected on municipality-level, rather than province-level, it is expected that the models would be more accurate and valuable.

1. Introduction

Every year, natural hazards like typhoons, earthquakes and wildfires affect close to 160 million people worldwide (WHO 2019). Due to climate change it is expected that scale, frequency and impact of these hazards will increase over the coming years which will result in an increasing demand of humanitarian aid (Ashdown 2011). However, not all of these people can be helped due to limited funding and resources for disaster response. Therefore, the areas and individuals with the highest priority have to be determined. During this process, humanitarian organizations quickly need insight on the damage in the areas that were hit by the disaster. Nevertheless, this can often be very time consuming and sometimes also subjective because of a lack of data and tools to interpret them (510.Global a).

About 20 years ago, machine learning techniques were introduced in the field of disaster management and have become one of the most effective methods for removing unrelated data and speeding up the analysis in disaster situations, which helps in fast prediction analysis and finding optimal response approaches (Yu, Yang, and Li 2018). Many researchers across the world have applied machine learning techniques in a wide variety of risk assessment tasks. Choi et al. (2018) experimented with algorithms like decision trees and random forests to predict heavy rain damage. Rajasekaran, Gayathri, and Lee (2008) used a support vector regressor to predict storm surges in order to avoid property loss and reduce risk by taking selective preventative action. In the study of Caragea et al. (2011), tweets and text messages were classified to address the most urgent needs and to understand the emergency situation better after the earthquake in Haiti.

These developments show that data science is not only applicable in straightforward industries like retail and finance but that it is also of great importance in disaster management. All the developments made in the applications of machine learning can and are being used to deal with bigger issues confronting humans, like preparing for and recovering from disasters (GFDRR 2018).

One of the sectors that is often affected by natural hazards is the agriculture. In developing countries this sector absorbs 23 percent of the total damage and losses. These damages have large negative impacts like disrupting production cycles, trade flows and livelihoods means (FAO 2017). It is therefore important to try to diminish the consequences by improving the preparation for these hazards. This study has focused on the prediction of rice loss due to typhoons. These storms are also known as cyclones or hurricanes, depending on where they occur. To make these rice loss predictions, data of the Philippines were used. Due to its geographical location, the Philippines is very vulnerable to typhoons. Annually, an average of 22 typhoons enter the archipelagic country, and with approximately 7 typhoons that cause significant damage it ranks second after China (Division 2015; OCHA 2019). Farmers that harvest rice, which is the staple food in the Philippines, often suffer from these hazards. According to a study of Israel (2012), the average damages to rice farming between 2007 and 2010 due to typhoons in the Philippines amounted 7,996.59 million Philippine Pesos per year (±135 million Euro). In order to diminish these damages, the Department of Agriculture (DA) regional office can lend two combine harvesters/threshers to farmers in order to save their crops from impending typhoons. According to the DA, farmers found this to be very advantageous in the past.

This study has aimed to answer the following research question:

Which machine learning algorithm performs best at predicting rice loss due to typhoons in the Philippines, and which features are selected for this algorithm?

The data for this study were provided by *510*: an initiative of the Netherlands Red Cross that uses data to improve speed, quality and cost-effectiveness of humanitarian aid. Together with Philippines Red Cross (PRC), German Red Cross (DRK) and Climate Centre it plays a big role in piloting Forecast-based Financing (FbF). FbF enables access to humanitarian funding for early action based on in-depth forecast information and risk analysis. The goal of FbF is to anticipate disasters, prevent their impact, if possible, and reduce human suffering and losses (DRK 2019).

In order to answer the research question, both regression and binary classification algorithms were implemented and their performance evaluated. Obviously, a model with continuous outcome is ideal, as aid workers can then carefully choose which province to help first. However, it was expected that predicting a continuous outcome would be hard due to typhoon specific variation that cannot be explained by the data. Therefore, also a binary classifier was trained. In detail, a linear regressor, random forest regressor and support vector regressor were implemented to predict the percentage of lost rice area. The evaluation metric for these regressors was the Mean Absolute Error (MAE). Furthermore, a binary classification was trained. For these tasks, random forest, support vector machine and k-nearest neighbor were applied. Every classification model was trained three times, each of them with a different threshold between the two classes: 20%, 30% and 40%. These numbers are the percentage lost rice area and represent the critical level for triggering anticipatory action towards farmers. For these models, the ability to rightly classify the positive samples was evaluated with the use of the recall score.

The models were trained on data of 11 big typhoons that hit the Philippines in the past 8 years. Data on the rice damage that these typhoons caused, were provided by the Philippines Department of Agriculture. The input data, which can be subdivided in geographical and typhoon data, were provided by 510 and were obtained with the use of Quantum Geographic Information System (QGIS). In total, 224 observations were available to train the models with. If, eventually, one of these models will be used in practice, the typhoon data (e.g. rainfall and wind speed) will be provided by University College London, a couple of days before the typhoon is expected to make landfall. As these variables cannot be known in advance, they will be obtained through a prediction model.

This study is organized as follows: Section 2 discusses the related work. Section 3 elaborates on the data, method and models used for this study. Section 4 shows the results and Section 5 discusses them. Finally, in Section 6 a conclusion will be drawn.

2. Related work and relevance

In this section, both the societal and academic relevance of this study are discussed.

2.1 Societal relevance

Originally, humanitarian organizations were created with the idea to respond to disasters after they occurred. Over the last years however, humanitarian funding is also being spent on the pre-disaster phase. Still, preventative actions that can be executed in the period between a warning and a potential disaster often stay out in the case of climate and weather forecasts (Coughlan de Perez et al. 2015). According to a study of disaster-related financing by the Global Facility for Disaster Reduction and Recovery (GFDRR) and the Overseas Development Institute (ODI) only 13% of disaster funding in the last 20 years was invested in reducing the risk of disaster before it happens, the rest was spent on emergency response, reconstruction, and rehabilitation (Kellett and Caravani 2013). That such a small amount is spent on the pre-disaster phase is a shame, according to the review study of Mechler (2005). In the majority of evaluations of preventative action, the avoided disaster losses could at least double the investment in risk reduction.

Nobre et al. (2019) also found positive results regarding preventative actions. In their study the potential cost-effectiveness of cash transfer responses were evaluated, comparing the relative costs of ex-ante cash transfers during the maize growing season to ex-post cash transfers after harvesting in Kenya. Overall, their findings suggest that early response can yield significant cost savings, and can potentially increase the effectiveness of existing cash transfer systems. Although this study is focused on crop damages due to extreme drought, it could be inferred that ex-ante cash transfers can also be helpful in cases of other weather and climate hazards.

510 has recently made a priority index that predicts the extent of damage to houses per municipality within 12 hours of the typhoon. The results can give organizations like the Philippines Red Cross and UN OCHA an overview of the geographic distribution of damage when there are no other sources of reports available yet (510.Global b).

2.2 Academic relevance

The models that are being used in the examples above rely on many different data types that involve complex science and big amounts of data. It is not always possible for experts to develop models that map out the potential impacts of a hazard on society. Therefore, machine learning techniques are being used instead. Machine learning can provide new methods of looking into these connections and provide more accurate and useful answers (GFDRR 2018). As mentioned before, data science and machine learning in particular have proven to be of great importance in the domain of disaster management. More and more studies use machine learning techniques in order to predict different types of damage due to natural disasters. Rajasekaran, Gayathri, and Lee (2008) used support vector regression to predict storm surges, while Hu and Ho (2014) used the same method to predict the impact of typhoons on transportation networks. Wang et al. (2015) applied a random forest classifier to flood hazard risk assessment and Pradhan and Jebur (2017) used both K-nearest neighbour and logistic regression algorithms to predict regions susceptible to natural hazards.

As for rice damage specifically, until now not more than a handful of studies have focused on the prediction of losses due to typhoons. Both studies of Blanc and

Strobl (2016) and Masutomi et al. (2012) have used fragility curves for estimating the extent of crop areas damaged by typhoons. Such a function relates external forces and the probability of damage. Chiang, Cheng, and Chang (2012) evaluated the impact of typhoons on agriculture in Taiwan and predicted the agricultural losses by a neural network. One limitation mentioned in the latter study is that the constructed model can only produce the losses of the whole Taiwan instead of on a more local level. The same limitations was also addressed in the study of Masutomi et al. (2012).

This study has experimented with several machine learning techniques that were not used before for crop loss prediction. Data of the Philippines were used and the output of these models were on province-level. Moreover, in contrast with the previous studies regarding crop damage prediction, in this study the models were trained to eventually make predictions a couple of days before the typhoon will make landfall, rather than shortly after the typhoon occurred.

3. Experimental Setup

In this section, the data as well as the methods used for the different prediction algorithms will be discussed.

3.1 Data

Now, first the output variable will be discussed, followed by a section regarding the input variables. Finally, in the last subsection, some visualizations and insights on the data will be given.

3.1.1 Output variable. The output for both regression and classification were on province-level. The output of the regression models is the percentage of the total rice area that has been totally damaged. To obtain the damaged proportion, the total rice area and the damaged rice area had to be known. First, the total rice area for every province in the Philippines was obtained via Philippines Rice Information System (PRISM). PRISM collects data and generates rice production information using mobile technology, remote sensing, and Geographical Information System (GIS) (PRISM). PRISM has rice area data on province-level of 2018 and 2017 for both the first and second semester. Semester 2 refers to the first cropping, usually starting around June. Semester 1 refers to the second cropping, which starts around November. The majority of the provinces has more rice area in season 2. This could be because farmers usually plant more rice at the end of the dry season (June-July) to benefit from the rain, especially in areas without irrigation systems. For this study, the rice area data of 2018 were used. For every observation, depending on when the typhoon occurred, the rice area datum for that province corresponding to either the first or second semester was used.

As for the damage data, these were provided by the Philippines Department of Agriculture. After filing a Freedom of Information (FOI)[1] request, the data were made public. It contained the damage data for 12 different typhoons on province-level. These 12 typhoons were chosen by 510 because these were the biggest typhoons in the past 8 years, and most of the input data for these typhoons were already available. The names of these typhoons are: Goni, Kalmaegi, Koppu, Haima, Sarika, Melor, Rammasun, Utor, Nock-Ten, Hagupit, Haiyan and Bopha. Typhoons Sarika and Haima occured in Oc-

1 https://www.foi.gov.ph/requests, ID: #DA-254466385434

tober 2016 a couple of days after another. In the data provided by the Department of Agriculture, the output of these typhoons were taken together. In general, Haima was much bigger than Sarika, so this output data have been treated as Haima only. This means that a total of 11 typhoons were used.

In the damage data, a distinction was made between totally damaged and with chance of recovery. For this study, only the totally damaged (lost) area was taken into account. After calculating the loss percentage for each observation, 3 of them exceeded 100%. It could be the case that in that time, there was more rice planted than 2018. Therefore, these values were replaced by 100%. In total, 224 observations were used to train the models.

Next to the regression models, three binary classification models have been trained, each of them having a different threshold. These thresholds are the percentage lost rice fields and represent the critical value for triggering early action. As there is not yet a gold standard for preventative action, a threshold of 20%, 30% and 40% were used.

3.1.2 Input variables. The 14 input variables can be subdivided in two categories: geographical and typhoon features. In the table below, the features for every category are listed. The majority of these features were used for earlier prediction models of 510. Besides, the the rice area, total length of streams and rivers and the drainage density were added.

Input Variables	
Geographic	Typhoon
Total area	Average wind speed
Total rice area	Total rainfall
Average elevation	Distance to typhoon
Average slope	
Average ruggedness	
Coast length	
Coast-perimeter ratio	
X and Y coordinate	
Total length of streams and rivers	
Drainage density	

Most of the geographical and typhoon features were obtained by 510 with the use of Quantum Geographical Information system (QGIS). QGIS is an application in which geospatial data can be analyzed. The land map of the Philippines was covered with so called raster layers: matrices of cells that represent features on the earth's surface. Each layer regards a different feature, and every cell in this layer contains the value for this feature (QGIS 2019). Some of these features were initially on municipality-level, and therefore needed a transformation to province-level. The average elevation, slope and ruggedness, were obtained by taking the area weighted average for all the municipalities in a certain province. The ruggedness is defined as the mean difference between a central cell's elevation value and the elevation values of its surrounding cells. The slope is defined as the angle of inclination to the horizontal. The X and Y coordinate are the transformed latitude and longitude of the centroid of the province. The drainage density is the ratio between the total length of streams and rivers and the total area of the province. The total rainfall is defined by the commutative rainfall during the typhoon event. Furthermore, the distance to typhoon is the perpendicular distance from the center of a province to the track of that typhoon.

For 3 of the 11 typhoons, the rainfall data were incomplete, so a second source was used to fill the missing rainfall data. This source was used for a previous project of 510 and contains data on municipality-level and thus had to be aggregated. This was done by taking the area weighted average. Still, after using this second source, the rainfall data for 17 observations were still not filled. For the observations that did contain rainfall data, the correlation of this feature with rice loss was insignificant (r=0.15, α=0.13). It was therefore expected that this feature would not be of importance when predicting rice loss, and that it would not be selected by the feature selection modules, which will be discussed in the next section. Therefore, the missing entries were replaced by the mean of the available rainfall data.

Worth noting is that the average wind speed was only captured when it was equal or larger than 40 miles per hour (mph). If this was not the case, an average wind speed of 0 mph was noted. As typhoons often have a high wind speed, this probably did not affect provinces close to the typhoon track. Therefore, it was expected that this would not negatively impact the prediction models in a significant way.

3.1.3 Data exploration. As mentioned before, the data consisted of 224 observations. These observations concern 57 different provinces. In Figure 1, the average loss per typhoon is visualized. The total average is 7.1% and was computed by taking the average over all observations. Although Haiyan was one of the strongest typhoons the Philippines ever encountered, one can see that typhoon Nock-Ten caused the biggest rice loss. Of these 11 typhoons, Nock-Ten was the most recent one (2016). It could be the case that before that time, the measuring of the lost rice areas was not properly done. Hence, the rice loss due to earlier typhoons, like typhoon Haiyan (2013), seem smaller. In Figure 2, one can see that the number of observations vary a lot between typhoons. This was taken into account when evaluating the models, which will be explained in detail in the next subsection. Furthermore, in Figure 3, the Pearson correlations of the features with rice loss are presented. As expected, the wind speed has the largest correlation (r=0.41, α=1.78e-5). The distance from the typhoon has the largest negative correlation (r=-0.29, α=0.002).

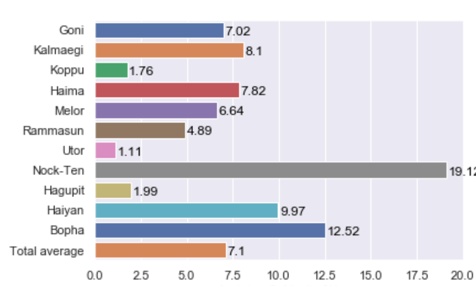

Figure 1
Average rice loss

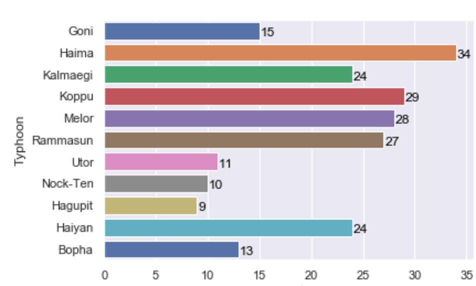

Figure 2
Observations per typhoon

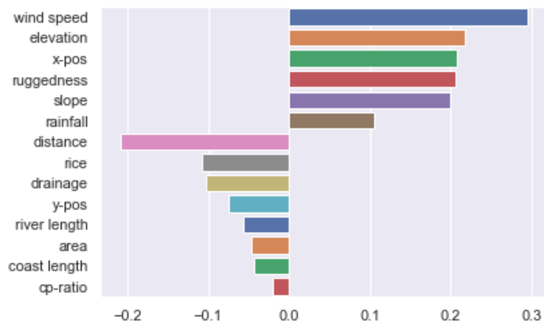

Figure 3
Correlation with rice loss

3.2 Method and models

In this section, the regression and classification models that were used for prediction are discussed in detail. But first, the evaluation and feature selection methods of the models are discussed.

Evaluation. To evaluate every model's performance, an 11-fold cross validation was carried out. For every fold, all the observations that concern one of the 11 typhoons were left out as a test set. This method is the most realistic, as in a real situation the damage of a typhoon is also being predicted on the basis of previous typhoons. Because of the inexplicable typhoon specific variation in the output, the performance of the model would be biased if both the training and test set contained observations of a certain typhoon. Finally, the average of the 11 scores was taken. However, as there was a big variation in the number of observations per typhoon (see Figure 2), the weighted average was taken. The evaluation metrics that were used for the regression and classification models are discussed later on.

Feature Selection. For every separate algorithm, feature selection was carried out. This means that only the most important features are taken into account, as sometimes too many features can lead to poor generalization of the model (Guyon and Elisseeff 2003). It is important that this is done together with the cross validation procedure, in order to prevent bias of the evaluation metric. Namely, when first the best features are selected based on all the data, and only thereafter cross-validation is used to estimate the prediction error, the test set is not completely independent. More precisely, the test data in every fold of the cross-validation procedure were also used to choose the best features. Feature selection is actually a kind of training and training can never be done on the test set as well. Therefore, test samples must be left out before selection steps are carried out (Friedman, Hastie, and Tibshirani 2001). As there are different approaches to feature selection, this study has experimented with different modules in order to find the best performing feature set for each algorithm. Every model has tried each of the following two approaches.

First of all, Scikit-Learn's module Recursive Feature Elimination (RFE) was used. In this case, the estimator starts with all the initial features. Then recursively, the least important feature is being discarded until the desired number of features is reached.

The importance of a feature is measured by the estimators rise in error or drop in accuracy, when this feature is being discarded. In contrast to the next method, RFE does not support the use of a support vector machine/regressor and k-nearest neighbor algorithm. This is because there is no notion of feature importance given in these algorithms. Therefore, this method was only used when training a random forest model and a linear regressor.

Secondly, the selection method SelectKBest from Scikit-Learn's univariate feature selection approach was used. Here, statistical tests are used to select k features that have the strongest relationship with the output variable. For the regression models, the scoring functions that were used are f_regression and mutual_info_regression. As for the classification models, chi2, f_classif and mutual_info_classif were used as scoring functions. Both the f_regression and f_classif tests rank features according to their correlation with the output variable. The mutual_info_regression, as well as the mutual_info_classif, are non-parametric tests, meaning that there are no distribution assumptions, which use k-nearest neighbors to measure the degree of relatedness between the features and the output variable (Ross 2014). Finally, the chi2 test computes the chi-squared statistic between each non-negative feature and class. The chi-square test measures dependence between stochastic variables, so using this function leaves out the features that are the most likely to be independent of class and therefore irrelevant for classification (Pedregosa et al. 2011).

For every model, the number of features, the selection method, and the hyperparameters used for that model are dependent of each other. Therefore, the number of features and the feature selection method were included in the grid search for every separate algorithm. In practice, this was carried out through nested for-loops, such that every possible combination of hyperparameters, number of features and selection methods was evaluated. For some algorithms, the features that were selected were not the same for every fold. In this case, the features that had the maximum number of votes were chosen and the cross validation procedure, including a new hyperparameter search, was repeated with these features. See Appendix A for the code of the cross-validation procedure for one of the algorithms.

Computer specifications. The algorithms in this study were trained and tested on a MacBook Air (2017) with an Intel Core i5 1.8-GHz processor, 8 GB 1600 MHz DDR3 RAM. The programming language used is Python 3.6.8, using the software libraries Scikit-learn version 0.20.3 and Pandas version 0.24.1.

3.2.1 Regression. For the regression algorithms, the Mean Absolute Error (MAE) was used to evaluate the performance. This easily interpretable metric was preferred over Root Mean Squared Error (RMSE) and Mean Squared Error (MSE) as larger errors did not necessarily need to be more penalized in this study. As mentioned before, after applying 11-fold cross-validation, the MAE was determined by taking the test size-weighted average of all the MAE's. After tuning the algorithm's hyperparameters, a baseline was defined by running that same model but only with wind speed as explanatory variable, as this feature has the biggest correlation with rice loss. Setting this baseline was only to put the model's performance into perspective. If the baseline would perform very well, this was also a positive outcome.

Now, the different regression models and their hyperparameter settings will be discussed.

Multiple linear regression. First of all, a multiple linear regression model was implemented. One of the reasons is that this model has the ability to make a prediction that is higher than the values in the training data. This is called extrapolating. In other words, it is possible that there will occur an extreme typhoon that was not seen earlier in the training data. In that case, the model has to extrapolate. For this straight forward algorithm, no hyperparameters were tuned. As for the feature selection, the f_regression test selecting only wind speed as explanatory variable led to the lowest MAE.

Random forest. Second, a random forest regressor was used to predict the lost rice area. In contrast with linear regression, random forest models are able to detect non-linear relations. However, if applied to extrapolating domains, it could lead to poor predictions (Hengl et al. 2018). In a random forest, multiple decision trees are generated, where after the random forest predictor is formed by taking the average over the trees. Each of these trees uses a random subset of features and a new training set. This training set is drawn, with replacement, from the original training set, also known as bagging (Breiman 2001). There are several hyperparameters that can be tuned for this algorithm. The optimal values for the number of estimators (n_est), the maximum depth (max_depth), the minimum sample split (min_split) and the minimum sample leaf (min_leaf) were found through grid search. The other hyperparameters were kept at their default values. The number of estimators represents the number of trees used in the forest. The maximum depth represents the maximum depth of every tree in the forest. The deeper the tree, the more information it can extract from the data. The minimum sample split stands for the minimum number of samples required to split an internal node. The minimum sample leaf parameter specifies the minimum number of samples in a leaf node (last node of the tree). Setting it lower leads to trees with a larger depth which means that more splits are performed until the leaf nodes (Probst, Wright, and Boulesteix 2018).

The search ranges were initialized to be $[4, 8]$, $\{2, 4, 6, 8, 10\}$, $[1, 5] \cup \{0.1\}$ and $[1, 5] \cup \{0.1\}$ for n_est, max_depth, min_split and min_leaf, respectively. For min_split and min_leaf, 0.1 represents the fraction of samples.

A total of 5 features were selected with the use of RFE. These were not identical for every fold. The most common features were wind speed, rice area, drainage density, average slope and average elevation. After repeating the cross validation procedure with these features, the following hyperparameter values obtained the lowest MAE: n_est=4, max_depth=4, min_split=2 and min_leaf=4.

Support vector regression. Finally, a support vector regressor (SVR) was used. This choice was based on several studies that have predicted the impact of typhoons using this algorithm (Hu and Ho 2014; Chang et al. 2018). SVR supports both linear and non-linear regression tasks. The goal is to find a function f that has at most ϵ deviation of the actual targets y_i (Smola and Schölkopf 2004). However, such a function does not always exist. Therefore, the penalty parameter C is introduced, which determines the trade-off between the algorithm's complexity and the amount up to which deviations larger than ϵ are tolerated. The kernel coefficient gamma defines how far the influence of a single training example reaches. When gamma is too small, the model is too constrained and cannot capture the complexity of the data.

For gamma, C and ϵ the hyperparameters tried are {'auto', 2e-4, 1e-4, 0.001, 0.1, 1}, {0.001, 0.1, 1, 2} and {0, 0.001, 0.1, 0.4, 0.8, 0.9}, respectively. When gamma='auto', the inverse of the number of features is taken.

For the SVR, the f_regression test selected just one feature: wind speed. The values of the hyperparameters that led to the best generalization were: ϵ=0.4, C=1 and gamma=0.001.

3.2.2 Classification. In this subsection, the algorithms that were used for binary classification are discussed. Every algorithm was trained three times, each of them with a different threshold: 20%, 30% or 40%. The numbers are the rice loss percentages and represent the critical level for triggering early action. For these classification models, the accuracy, recall, precision and f1-score were evaluated. The models were trained with the task to improve its recall score. Namely, it is more important to rightly classify the provinces that are in need of aid, than to rightly classify the provinces that are not in need of aid. Also, although it will be a waste of money to send aid to those that are not in need, skipping the farmers that are in need is considered as a more serious error.

As mentioned before, after the cross-validation procedure, the weighted average of the evaluation metrics was taken. The accuracy was test size-weighted. The recall was weighted on the basis of the number of positive samples in y_{true}. As for the precision, this metric was weighted on the basis of the number of positive samples in y_{pred}. For some folds, either the precision or recall was undefined (weight 0) such that the f1-score could not be computed. Forman and Scholz (2010) have experimented with different methods when dealing with this problem. They recommended numbering the number of true positives (TP), false positive (FP) and false negatives (FN) over the folds and then use equation (1), as this method is almost perfectly unbiased.

$$\text{F1-score} = 2 \cdot \frac{\text{Pr} \cdot \text{Re}}{\text{Pr} + \text{Re}} = 2 \cdot \frac{(\frac{\text{TP}}{\text{TP+FP}}) \cdot (\frac{\text{TP}}{\text{TP+FN}})}{(\frac{\text{TP}}{\text{TP+FP}}) + (\frac{\text{TP}}{\text{TP+FN}})} = \frac{2 \cdot \text{TP}}{\text{TP} + \text{FN} + \text{FP}} \tag{1}$$

Furthermore, as in all the three cases the classes were imbalanced (see Figure 4), a re-sampling method was applied. In detail, a random oversampler aimed to balance class distribution through the random replication of minority class examples (Batista, Prati, and Monard 2004). It is important to apply such a re-sampling method only after the test set is split from the training set. If first re-sampling is applied and then the data are split, both the training and test set could contain the exact same observations, resulting in a bias of the evaluation metric.

Now, the different classification models and their hyperparameter settings will be discussed.

Random forest. Next to regressors, random forest models can also serve as classifiers. Instead of taking the average of the outcomes of every decision tree, the classifier assigns the observation to the majority vote class. The hyperparameters and their search ranges used for the random forest regressor, were also used for for the classification task.

For the binary classification where the threshold was set at 20%, the RFE method selected 3 features: wind speed, drainage density and rice area. The following hyperparameters performed best: n_est=6, max_depth=2, min_split=0.1 and min_leaf=0.1.

For the the threshold of 30%, the model performed best when a total of 9 features were included. The features that were not included are: total area, coast-perimeter ratio and the X and Y coordinate. The following hyperparameters values were obtained: n_est=4, max_depth=2, min_split=0.1 and min_leaf=0.1.

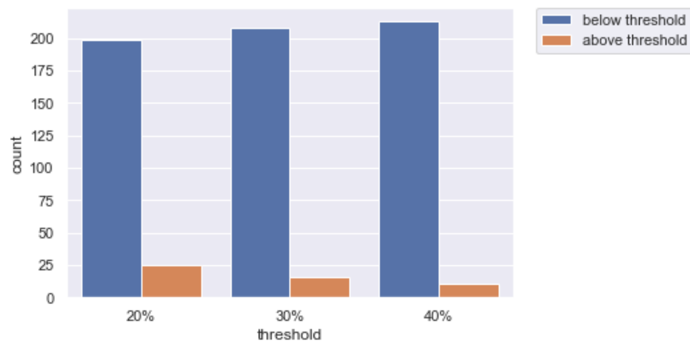

Figure 4
Class distribution for different thresholds

For the the model with a threshold of 40%, the following three features were selected with the RFE method: wind speed, slope and drainage density. Using these 3 features, the model performed best with the following hyperparameter values: n_est=4, max_depth=4, min_split=0.1 and min_leaf=4.

Support vector machine. Furthermore, a support vector machine (SVM) classifier was trained. The same hyperparameters as for the support vector regressor were tuned, except for ϵ. This hyperparameter is nonexistent for classification, as the aim is not to create a function that predicts the target value but to find a hyperplane that maximizes the margin between the two classes. In addition, the tolerance (tol) of the stopping criterion was also taken into account during the grid search, as this hyperparameter substantially increased the model's performance. For the tolerance, the set {0.1, 0.2, 0.3, 0.4, 0.5} was used for grid search. For the other hyperparameters, the hyperparameter sets were equal to the ones of the support vector regressor.

In all three cases, only wind speed was selected as feature with the f_classif test.

For the model with a threshold of 20%, the hyperparameter values that performed best were C=0.001, gamma='auto' and tol=0.1.

When the threshold was set at 30%, the following hyperparameter values were selected: gamma=1e-4, C=0.001, tol=0.1.

Finally, for the model with a threshold of 40% the following hyperparameter values were found through grid search: C=0.001, gamma=2e-4 and tol=0.1.

K-nearest neighbour. Finally, a k-nearest neighbour (k-NN) classifier was trained. This algorithm assigns every observation from the test set to the class of majority vote of the nearest k neighbors (Cover and Hart 1967). The value of k is a hyperparameter for which the search range was set at $[1, 15]$.

For the classification with a threshold of 20%, two features were selected with the use of the f_classif test: wind speed and distance to typhoon. The optimal value for k was 15.

For the model with a threshold of 30%, a total of 4 features were selected with the use of the mutual_info_classif test: distance to typhoon, total rainfall, slope and rice area. Setting the value of k at 11 made the model perform best.

Finally, the f_classif test selected one feature for the model with a threshold at 40%. Not all selected features were the same, but wind speed was the most common feature. After repeating the grid search, an optimal value of 11 was found for k.

4. Results

In this section, the results of the regression and classification models are presented. The outcomes of the regression algorithms are given in percentage point (p.p.). For every task, the algorithm in bold is the best performing algorithm. One can see that the support vector regressor performed best at the regression task with an MAE of 6.83 percentage points, representing the average difference between the predicted and the actual value. However, with an MAE of 7.36 percentage points the random forest regressor did not perform significantly worse. The linear regression model performed relatively worst, with an MAE of 8.90 percentage points.

As for the binary classifiers, the support vector machine obtained the highest recall score for all three thresholds. Recall scores of 88%, 75% and 81.82% were obtained when the threshold was set at 20%, 30% and 40%, respectively. One can clearly see the trade-off between the recall and precision of every classification model. When more positive labels are predicted, chances are high that most of the positive samples are rightly predicted, resulting in a high recall score. However, there is also a big chance that the predictor falsely labels the negative samples as positive, which causes a low precision score.

Regression		
Model	MAE (p.p.)	Baseline (p.p.)
Linear Regression	8.90	8.90
Random Forest	7.36	8.65
SVR	**6.83**	**6.83**

Binary Classification 20%				
Model	Accuracy (%)	Recall (%)	Precision (%)	F1-score (%)
Random forest	81.25	68.00	33.33	57.63
SVM	**50.89**	**88.00**	**17.05**	**33.33**
k-NN	76.79	84.00	30.43	57.53

Binary Classification 30%				
Model	Accuracy (%)	Recall (%)	Precision (%)	F1-score (%)
Random forest	72.32	68.75	16.18	26.20
SVM	**66.52**	**75.00**	**14.46**	**27.59**
k-NN	70.09	56.25	13.04	23.68

Binary Classification 40%				
Model	Accuracy (%)	Recall (%)	Precision (%)	F1-score (%)
Random forest	83.93	72.73	19.51	36.36
SVM	**66.07**	**81.82**	**10.84**	**21.18**
k-NN	79.91	72.73	16.00	30.19

5. Discussion

In this section, the results of the study are put into perspective. Then, limitations of this study and suggestions for future studies will be discussed.

As mentioned before, the support vector regressor performed best with a Mean Absolute Error (MAE) of 6.83 percentage points. By way of comparison, the average rice loss amounted 7.1%.

As for the binary classification, the models were evaluated on their ability to rightly classify the positive samples. In all three cases, the support vector classifier obtained the highest recall score. In all three cases, almost all provinces that would be in need of aid, would also receive this. However, all of these models also had a low precision score, meaning that lots of provinces would be helped although they would not be in need.

To conclude, for both regression and classification tasks, the support vector machine performed best. For each of these models, only wind speed was used as explanatory feature. It is remarkable that for most of the other algorithms, different sets of features came out best. Also, for the random forest algorithms, the hyperparameters used for the algorithms had great impact on features that were selected. This suggests that with the data that was available for this study, there was no convincing outcome regarding the most important features when predicting rice loss.

As for the possibility to use one of these models, the Philippines Red Cross can now evaluate the models and their performances in order to decide whether these can be used in the future.

5.1 Limitations and suggestions for future research

There are a number of limitations of this study that could be improved in future research. The first and most important limitation is that the amount of data was limited. It is important that more data are gathered such that more robust and accurate predictions can be made. With only 224 observations, this study has given an insight on the possibilities of rice loss prediction but the estimations are imprecise. Not only were there not much training data to train the models on, as an 11-fold cross-validation was used, each predictor was only tested on a test set consisting of only 20 instances on average, which does not give a thorough reflection of the prediction performance. As already stated above, the most important features were not similar across the different algorithms and for some algorithms the hyperparameter settings had great impact on the features that were selected. It is expected that when there are more data available to train the models on, it will become clearer which features are most important when predicting rice loss. However, it is important to keep in mind the set of most important features may differ per country. For example, it could be the case that for a certain country with a varied vegetation, the tree cover density is considered as an important feature whereas for another country this does not hold. However, a feature like wind speed is likely to be of great importance in all countries.

Next to the amount of data, also the quality of the data can be improved. For example, the rainfall data were not complete and the total rice area data from 2018 were used although the typhoons occurred between 2012 and 2016. In reality the rice area, and thus the percentage rice loss, could have been larger or smaller. Also, the fact that the models in this study were on province-level rather than on municipality-level could have played a role in the mediocre performance of some of the models. It is expected that the prediction models on municipality-level would perform better, as the input features would be more specific, instead of the global, aggregated province features. From the disaster management perspective, it would also be more valuable when predictions would be on municipality-level rather than on province-level, as in the latter case the outcome does not give a thorough insight of where exactly the aid is needed. On average, a province of the Philippines has an area of almost 4000 km^2, implying that the damages can vary a lot per municipality within that province. It is therefore important that a similar study, but then on municipality-level, will be carried out.

Furthermore, the models in this study were trained with the objective to eventually predict rice loss a couple of days before a typhoon would make landfall. This means that the typhoon related features themselves are also obtained through a prediction model. This can negatively influence the performance of the model by limited accuracy of these predictions.

Finally, the use of additional features could result in a better performance of the models. For example, in their study, Blanc and Strobl (2016) made a distinction between irrigated and rain-fed rice fields. Their findings suggested that irrigation technology is better able to deal with the potential damage due to typhoons. For example, irrigation systems may be able to counteract the excessive flooding during a storm. Chiang, Cheng, and Chang (2012) used a neural network to predict typhoon induced losses on agriculture. They used among others, the minimum atmospheric pressure and the coverage of the typhoon as explanatory features.

6. Conclusion

To conclude, this study has tried different machine learning techniques to predict rice loss due to typhoons in the Philippines. This study has been carried out because it is expected that a lot of damages that farmers encounter as a result of these typhoons can be saved when preventative actions are taken.

For this task, the data of 11 typhoons were used to train both regression and binary classification models on province-level. For the binary classification, three different thresholds that represent the critical value for triggering early action were used: 20%, 30% and 40% lost rice area. The classifiers were trained to increase its ability to rightly predict the positive samples. For the regression task, as well as for the three different binary classification tasks, the support vector machine performed best but were still imprecise. For each of these latter models only wind speed was selected as explanatory feature. However, for the other algorithms that were trained in this study, other sets of features were used. Moreover, for some algorithms the hyperparameter settings had great impact on the the features that were selected. This is most probably a consequence of the small dataset that was used for this study. It is therefore important that more data are gathered in order to make more robust and accurate predictions. Also, if there can be damage data collected on municipality-level, rather than province-level, it is expected that the models would be more accurate and valuable.

References

510.Global. a. Community risk assessment. *https://dashboard.510.global/#!/.*

510.Global. b. Faster response: With capacity building philippines part 1. *https://www.510.global/surge-informatio-the-philippines/.*

Ashdown, P. 2011. *Humanitarian emergency response review*.

Batista, G. E. A. P. A, R. C. Prati, and M. C. Monard. 2004. A study of the behavior of several methods for balancing machine learning training data. *ACM SIGKDD explorations newsletter*, 6.1:20–29.

Blanc, Elodie and Eric Strobl. 2016. Assessing the impact of typhoons on rice production in the philippines. *Journal of Applied Meteorology and Climatology*, 55.4:993–1007.

Breiman, Leo. 2001. Random forests. *Machine learning*, 35-1:5–32.

Caragea, C., N. McNeese, A. Jaiswal, G. Traylor, H.W. Kim, P. Mitra, D. Wu, A. H. Tapia, L. Giles, B. J. Jansen, and J Yen. 2011. Classifying text messages for the haiti earthquake. *Proceedings of the 8th international conference on information systems for crisis response and management*, 8.5.

Chang, M. J., H. K. Chang, Y. C. Chen, G. F. Lin, P. A. Chen, J. S. Lai, and Y. C. Tan. 2018. A support vector machine forecasting model for typhoon flood inundation mapping and early flood warning systems. *Water*, 10.

Chiang, Yen-Ming, Wei-Guo Cheng, and Fi-John Chang. 2012. A hybrid artificial neural network-based agri-economic model for predicting typhoon-induced losses. *Natural hazards*, 63.2:769–787.

Choi, C., J. Kim, J. Kim, D. Kim, Y. Bae, and H.S. Kim. 2018. Development of heavy rain damage prediction model using machine learning based on big data. *Advances in Meteorology*.

Cover, T. M. and P. E. Hart. 1967. Nearest neighbor pattern classification. *IEEE transactions on information theory*, 13.1:21–27.

Division, Hurricane Research. 2015. Subject: E25) which countries have had the most tropical cyclones hits? frequently asked questions. *http://www.aoml.noaa.gov/hrd/tcfaq/E25.html*.

DRK, (German Red Cross). 2019. What is forecast-based financing? *https://www.forecast-based-financing.org/.*

FAO, (Food & Agriculture Organization). 2017. The impact of disasters on agriculture: Addressing the information gap.

Forman, G. and M. Scholz. 2010. Apples-to-apples in cross-validation studies: pitfalls in classifier performance measurement. *ACM SIGKDD Explorations Newsletter*, 12.1:49–57.

Friedman, J., R. Hastie, and T. Tibshirani. 2001. *The elements of statistical learning: Volume 1*. Springer, New York.

GFDRR. 2018. Machine learning for disaster risk management.

Guyon, Isabelle and A. Elisseeff. 2003. An introduction to variable and feature selection. *Journal of machine learning research*, pages 1157–1182.

Hengl, T., M. Nussbaum, M. N. Wright, G. B. Heuvelink, and B. Graler. 2018. Random forest as a generic framework for predictive modeling of spatial and spatio-temporal variables. *PeerJ*, 6:1578–1587.

Hu, T. Y. and W. M. Ho. 2014. Prediction of the impact of typhoons on transportation networks with support vector regression. *Journal of Transportation Engineering*, 10.

Israel, D. 2012. Typhoons, floods, and droughts: Regional occurrence and value of damages to rice farming in the philippines. *PIDS Policy Notes*, 1.

Kellett, J. and A. Caravani. 2013. Financing disaster risk reduction: A 20 year story of international aid. *ODI, http://www.odi.org.uk/publications/7452-climate-finance-disaster-risk-reduction*.

Masutomi, Yuji, T. Iizumi, K. Takahashi, and M. Yokozawa. 2012. Estimation of the damage area due to tropical cyclones using fragility curves for paddy rice in japan. *Environmental Research Letters*, 7.1.

Mechler, R. 2005. Cost-benefit analysis of natural disaster risk management in developing countries. *Deutsche Gesellschaft fuer Technische Zusammenarbeit*.

Nobre, G. G., F. Davenport, K. Bischiniotis, T. Veldkamp, B. Jongman, C.C. Funk, P.J. Ward, and J.C.J.H Aerts. 2019. Financing agricultural drought risk through ex-ante cash transfers. *Science of the Total Environment*, 653.

OCHA. 2019. About ocha in the philippines. *https://www.unocha.org/philippines/about-ocha-philippines*.

Pedregosa, F., G. Varoquaux, A. Gramfort, V. Michel, B. Thirion, O. Grisel, M. Blondel, P. Prettenhofer, R. Weiss, V. Dubourg, J. Vanderplas, A. Passos, D. Cournapeau, M. Brucher,

M. Perrot, and E. Duchesnay. 2011. Scikit-learn: Machine learning in python. *Journal of Machine Learning Research*, 12:2825–2830.

Coughlan de Perez, E., B. van den Hurk, M. K. van Aalst, B. Jongman, T. Klose, and P. Suarez. 2015. Forecast-based financing: an approach for catalyzing humanitarian action based on extreme weather and climate forecasts. *Natural Hazards Earth System Sciences*, 15(2012):895+.

Pradhan, B. and M. N. Jebur. 2017. *Spatial prediction of landslide-prone areas through k-nearest neighbor algorithm and logistic regression model using high resolution airborne laser scanning data*. Springer, Cham, Switzerland.

PRISM. Dynamic reports. *https://prism.philrice.gov.ph/dynamic-reports/*.

Probst, P., M. N. Wright, and A. L. Boulesteix. 2018. Hyperparameters and tuning strategies for random forest. *Wiley Interdisciplinary Reviews: Data Mining and Knowledge Discovery*, e1301.

QGIS, Development Team. 2019. Qgis geographic information system open source geospatial foundation project. *http://qgis.osgeo.org*.

Rajasekaran, S., S. Gayathri, and T-L. Lee. 2008. Support vector regression methodology for storm surge predictions. *Ocean Engineering*, 35.16:1578–1587.

Ross, Brian C. 2014. Mutual information between discrete and continuous data sets. *PloS one*, 9-2.

Smola, Alex J. and Bernhard Schölkopf. 2004. A tutorial on support vector regression. *Statistics and computing*, 14-3:199–222.

Wang, Zhaoli, C. Lai, X. Chen, B. Yang, S. Zhao, and X Bai. 2015. Flood hazard risk assessment model based on random forest. *Journal of Hydrology*, 527:1130 – 1141.

WHO, (World Health Organization). 2019. Natural hazards. *https://www.who.int/environmental_health_emergencies/natural_events/en/*.

Yu, M., C. Yang, and Y. Li. 2018. Big data in natural disaster management: a review. *Geosciences*, 8.5.

Appendix A: Cross-validation procedure

The cross-validation procedure for the k-NN classifier when the threshold was set at 40%. The grid search includes the model's hyperparameter k as well as the number of features and feature selection test.

```
best_acc=[]
for y in ['all',1,2,3,4,5,6,7,8,9,10,11,12,13]:
    for j in range(1,16):
        for u in [mutual_info_classif, chi2, f_classif]:
            np.random.seed(15)
            accuracy=[]
            recall=[]
            for i in not_null.Typhoon.unique():

                train = bin_classi4.loc[bin_classi4["Typhoon"]!= i].drop(['Typhoon', "Pro_Name",
                                                    "Pro_Code", "no_recovery", "percentage"], axis=1)
                test = bin_classi4.loc[bin_classi4["Typhoon"]== i].drop(['Typhoon', "Pro_Name",
                                                    "Pro_Code", "no_recovery", "percentage"], axis=1)

                y_train = train["output"]
                y_test = test["output"]
                x_train = train.drop(["output"],axis=1)
                x_test = test.drop(["output"],axis=1)

                sm = RandomOverSampler(random_state=6, sampling_strategy='auto')
                x_train, y_train = sm.fit_resample(x_train, y_train)

                best = SelectKBest(u, k=y)
                x_train = best.fit_transform(x_train,y_train)

                knn = KNeighborsClassifier(n_neighbors=j).fit(x_train,y_train)

                score = accuracy_score(y_test,knn.predict(best.transform(x_test)))
                accuracy.append(round(score,4))
                recall.append(recall_score(y_test,knn.predict(best.transform(x_test))))

            if not best_acc:
                best_acc.append(round(np.average(recall, weights=length4),4))
                best_acc.append(round(np.average(accuracy,weights=length),4))
                best_acc.append([y,j,str(u)])
            if round(np.average(recall, weights=length4),4) > best_acc[0]:
                best_acc[0] = round(np.average(recall, weights=length4),4)
                best_acc[1] = round(np.average(accuracy,weights=length),4)
                best_acc[2] = [y,j,str(u)]
                print(best_acc)
```

Selecting just one feature with the use of the f_classif test, resulted in the highest recall score. The following features were selected.

```
['avg_speed_mph']
['avg_speed_mph']
['avg_speed_mph']
['avg_speed_mph']
['avg_speed_mph']
['avg_speed_mph']
['avg_speed_mph']
['Ruggedness_avg']
['avg_speed_mph']
['avg_speed_mph']
['avg_speed_mph']
```

The grid search was carried out again with the most common feature: wind speed.

```
best_acc=[]
for j in range(1,16):
    np.random.seed(6)
    accuracy, recall, precision, precision_weights=[], [], [], []
    TP, FP, FN =0,0,0
    for i in not_null.Typhoon.unique():
        train = bin_classi4.loc[bin_classi4["Typhoon"]!= i].drop(['Typhoon', "Pro_Name",
                                                    "Pro_Code", "no_recovery", "percentage"], axis=1)
        test = bin_classi4.loc[bin_classi4["Typhoon"]== i].drop(['Typhoon', "Pro_Name",
                                                    "Pro_Code", "no_recovery", "percentage"], axis=1)
        y_train = train["output"]
        y_test = test["output"]
        x_train = train[['avg_speed_mph']]
        x_test = test[['avg_speed_mph',]]
        sm = RandomOverSampler(random_state=6, sampling_strategy='auto')
        x_train, y_train = sm.fit_resample(x_train, y_train)
        knn = KNeighborsClassifier(n_neighbors=j).fit(x_train,y_train)
        accuracy.append(round(accuracy_score(y_test,knn.predict(x_test)),4))
        recall.append(recall_score(y_test,knn.predict(x_test)))
        precision.append(precision_score(y_test,knn.predict(x_test)))
        precision_weights.append(list(knn.predict(x_test)).count(1))
        for t in range(len(y_test)):
            if list(knn.predict(x_test))[t] == list(y_test)[t] and list(y_test)[t]== 1:
                TP +=1
            if list(knn.predict(x_test))[t] == 1 and list(y_test)[t] == 0:
                FP+=1
            if list(knn.predict(x_test))[t] == 0 and list(y_test)[t] == 1:
                FN+=1
        fscore = (2*TP) /(TP+FP+FN)
    if not best_acc:
        best_acc.append(round(np.average(recall, weights=length4),4))
        best_acc.append(round(np.average(accuracy,weights=length),4))
        best_acc.append(round(np.average(precision,weights=precision_weights),4))
        best_acc.append(round(fscore,4))
        best_acc.append([j])
    if round(np.average(recall, weights=length4),4) > best_acc[0]:
        best_acc[0] = round(np.average(recall, weights=length4),4)
        best_acc[1] = round(np.average(accuracy,weights=length),4)
        best_acc[2] = round(np.average(precision,weights=precision_weights),4)
        best_acc[3] = round(fscore,4)
        best_acc[4] = [j]
```

k=11 obtained highest recall.

```
Recall: 0.7273
Precision: 0.16
Accuracy: 0.7991
fscore: 0.3019
k= [11]
```