Predicting Eosinophil Oesofagitis with machine learning - report

Jim Jim Valkema
11/16/2020

introduction

Eosinophilic esophagitis is a food allergy in the esophagus. The allergy causes a type of white blood cells called eosinophils to build up in the lining of the esophagus which cause swelling, pain and discomfort. The allergy gets triggered by different types of food for different people. These triggers are usually found by making the patient follow an elimination diet.
Eosinophilic esophagitis is diagnosed by a gastroenterologist who will examine the esophagus with an endoscope while the patient is under anesthesia and take a biopsy from the esophagus if needed. This is a moderately invasive procedure for diagnosis and that is why there is another method of diagnoses researched in this study. This study proposes a method of diagnosis by measuring certain blood markers and identifying patients with eosinophilic esophagitis with a machine learning algorithm.

materials and methods

data

Nutritional intake was assessed in 40 Dutch adult EoE patients participating in the Supplemental Elemental Trial (SET) using 3-day food diaries. In this randomized controlled trial, diagnosed patients received either a four-food elimination diet alone (FFED) or FFED with addition of an amino acid-based formula (Neocate) for 6 weeks. Disease severity was assessed by peak eosinophil count/high power field (PEC) in esophageal biopsy specimens. Multiple linear regression analyses were performed to assess associations between the intake of nutrients and foods per 1000 kCal and PEC, both at baseline and after the 6 weeks diet, while controlling for baseline variables.

software

The exploratory data analysis has been done using rmarkdown with the following packages: * ggplot: for creating plots
* gridExtra: further styling plots
* reshape2: transform data from wide and long formats
* rlist: appending items in a list
* ggcorrplot: creating a correlation plot

Creating the machine learning model has been done in weka.

Development of the wekawrapper command line tool has been done in intellij with the java programming language.

exploratory data analysis

There are quite a bit of missing attributes and missing values in this data set. Luckily there are plenty of attributes that do have data from where the machine learning algorithm can make its prediction. Figure 1 shows that a lot of attributes miss more than 70% of their values which in combination with the few instances of this data set means that those attributes are not of value to a machine learning algorithm and are therefore removed in this study. However the data does only have 42 instances so training an algorithm could result in over-fitting.

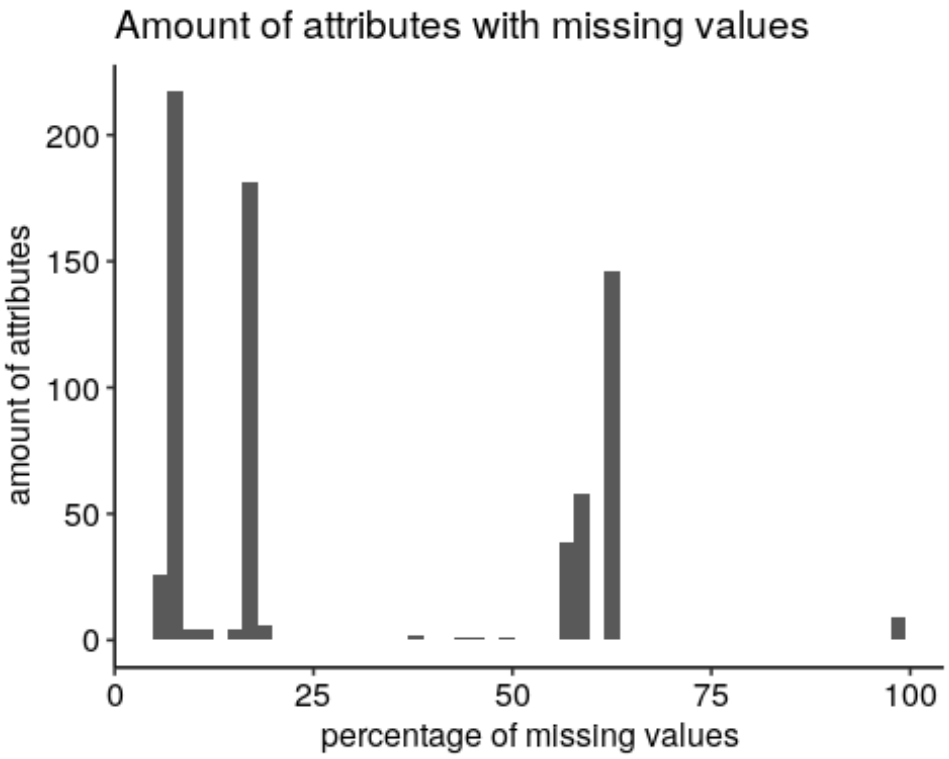


Figure 1: Shows the amount of attributes that miss a certain percentage of values. With the percentage of missing values on the x axis and the amount of attributes on the y. This plot clearly show that a lot of attributes miss 60~70 percent of their values

The data has also been log two transformed so that it would better fit a normal distribution (figure 4) which acts more predictable with most machine learning algorithms. Furthermore the data has been normalized (Figure 2 and 3) with scaling normalization. Scaling normalization. normalizes the data with the standard deviation which has the advantage of being able to scale future measurements even when they range outside of the training data set unlike min-max normalization. This is especially important for this data set since it

contains such few instances which increases the chance that a new instance has a larger range.

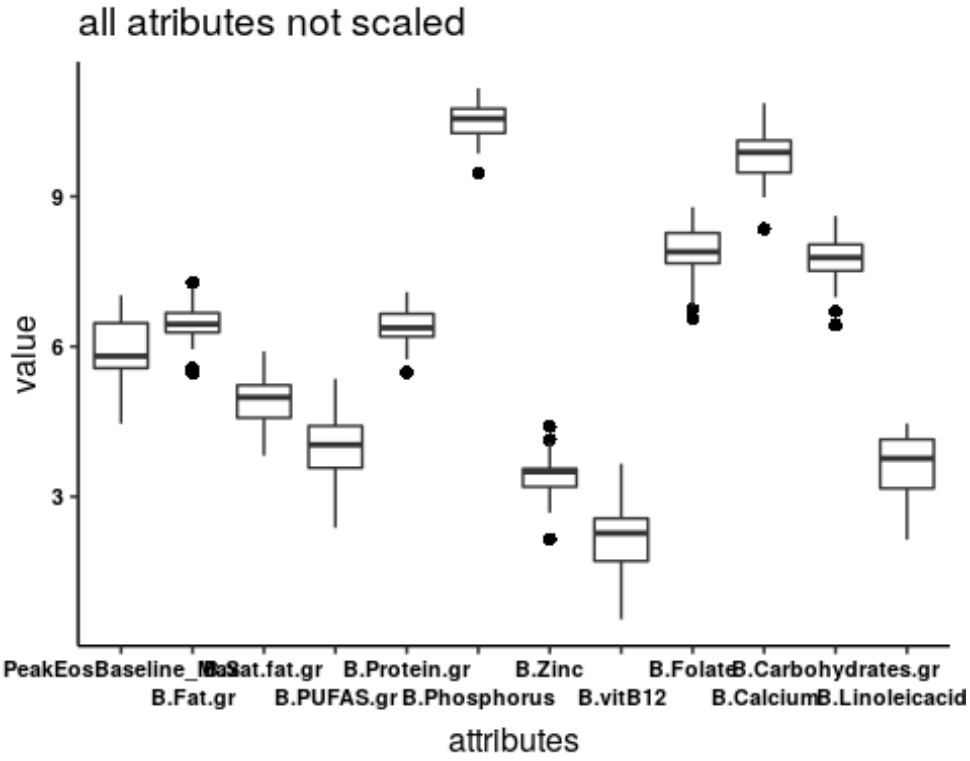


Figure 2: Shows a selection of attributes and their distributions as a box-plot. The x axis shows the attributes and y axis its values those attributes range within. This plot shows that the range varies largely between attributes.

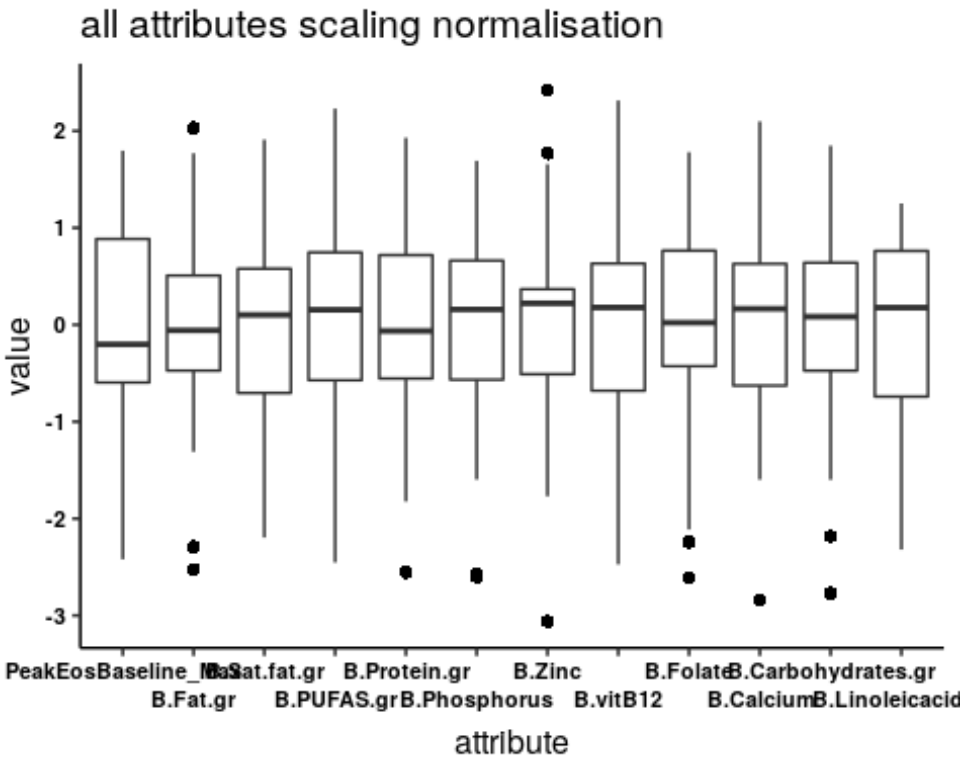


Figure 3: Shows a selection of attributes and their distributions after scaling normalization. as a box-plot. The x axis shows the attributes and y axis its values those attributes range within. This plot shows the distributions are far more comparable then without normalization.

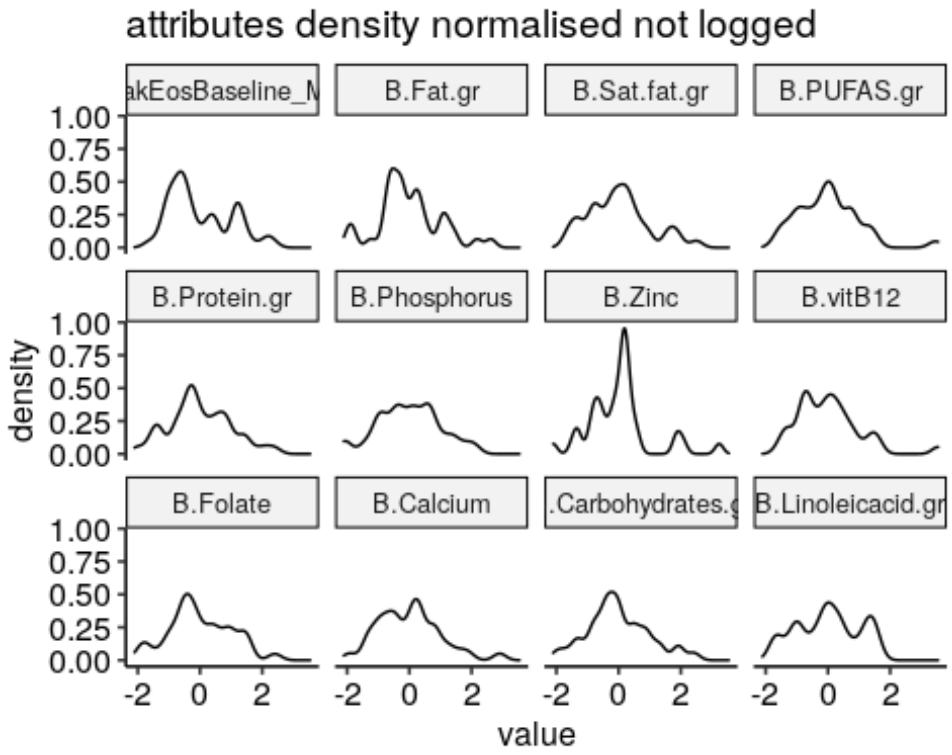
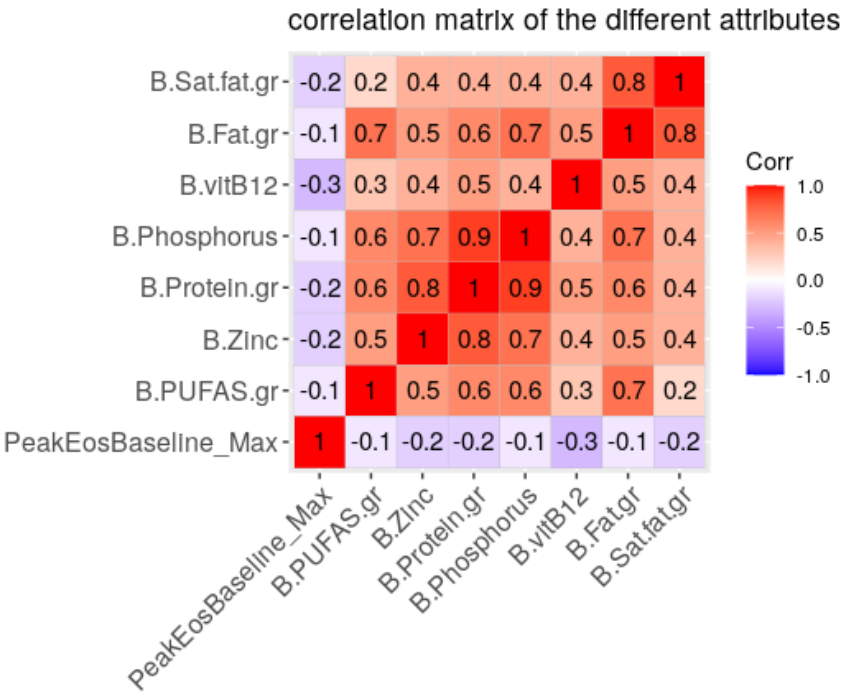


Figure 4: Shows the density of the values of a selection of attributes from the data set. With the density on the y axis and the values on the x axis. This plot shows that the data is now far more normal distributed after scaling and more comparable in its ranges.

It seems that the majority isn't highly correlated which meant that there is enough information between attributes for the algorithm to train on. Some attributes like the fats (B.Sat.fat.gr, B.Fat.gr, etc) are highly correlated (figure 6). So only one of these ended up in the final training data set because the others convey very little information to the algorithm. In this case the polyunsaturated fatty acids (B.PUFAS.gr) are one of the attributes who ended up in the final training data set during the attribute selection.


correlation matrix of the different attributes
B.Sat.fat.gr -0.2 0.2 0.4 0.4 0.4 0.4 0.8 1
B.Fat.gr -0.1 0.7 0.5 0.6 0.7 0.5 1 0.8
B.vitB12 -0.3 0.3 0.4 0.5 0.4 1 0.5 0.4
B.Phosphorus -0.1 0.6 0.7 0.9 1 0.4 0.7 0.4
B.Protein.gr -0.2 0.6 0.8 1 0.9 0.5 0.6 0.4
B.Zinc -0.2 0.5 1 0.8 0.7 0.4 0.5 0.4
B.PUFAS.gr -0.1 1 0.5 0.6 0.6 0.3 0.7 0.2
PeakEosBaseline_Max 1 -0.1 -0.2 -0.2 -0.1 -0.3 -0.1 -0.2
PeakEosBaseline_Max
B.PUFAS.gr
B.Zinc
B.Protein.gr
B.Phosphorus
B.vitB12
B.Fat.gr
B.Sat.fat.gr
Corr
1.0
0.5
0.0
-0.5
-1.0

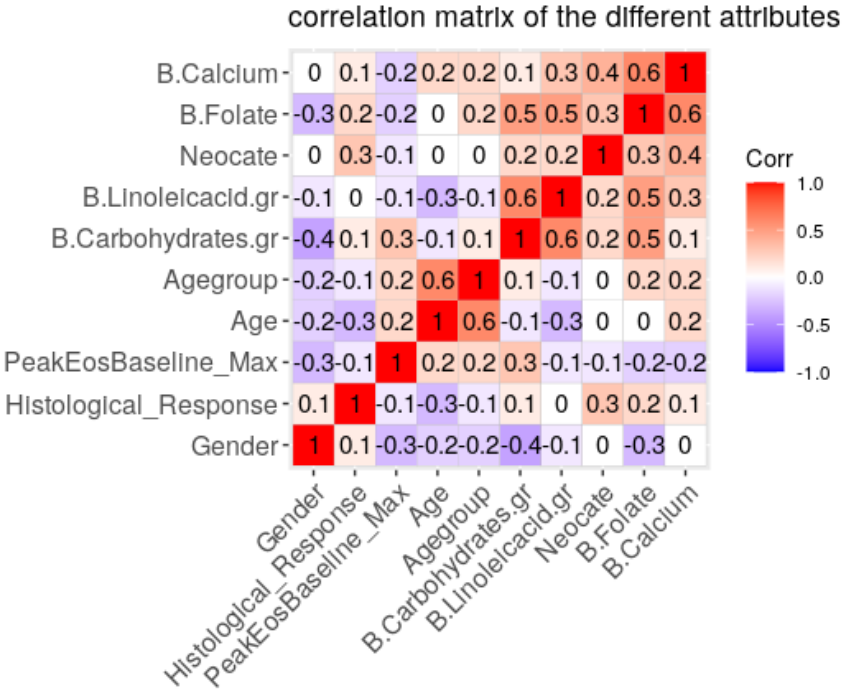

correlation matrix of the different attributes
B.Calcium 0 0.1 -0.2 0.2 0.2 0.1 0.3 0.4 0.6 1
B.Folate -0.3 0.2 -0.2 0 0.2 0.5 0.5 0.3 1 0.6
Neocate 0 0.3 -0.1 0 0 0.2 0.2 1 0.3 0.4
B.Linoleicacid.gr -0.1 0 -0.1 -0.3 -0.1 0.6 1 0.2 0.5 0.3
B.Carbohydrates.gr -0.4 0.1 0.3 -0.1 0.1 1 0.6 0.2 0.5 0.1
Agegroup -0.2 -0.1 0.2 0.6 1 0.1 -0.1 0 0.2 0.2
Age -0.2 -0.3 0.2 1 0.6 -0.1 -0.3 0 0 0.2
PeakEosBaseline_Max -0.3 -0.1 1 0.2 0.2 0.3 -0.1 -0.1 -0.2 -0.2
Histological_Response 0.1 1 -0.1 -0.3 -0.1 0.1 0 0.3 0.2 0.1
Gender 1 0.1 -0.3 -0.2 -0.2 -0.4 -0.1 0 -0.3 0
Gender
Histological_Response
PeakEosBaseline_Max
Age
Agegroup
B.Carbohydrates.gr
B.Linoleicacid.gr
Neocate
B.Folate
B.Calcium
Corr
1.0
0.5
0.0
-0.5
-1.0

Figure 5 & 6: show the correlation between attributes. It seems that the majority isn't highly correlated which might be a good indication that there is enough information between attributes for the algorithm to train on.

The Peak eosinophil baseline max needs to be split into categories because most classical machine learning algorithms can't use a numerical dependent class variable. The Peak eosinophil baseline max is split into three categories: low, mid, high split at 0-49, 49-75, 75-200. This is found to create the most even distribution of instances while also being informative (figure 7).

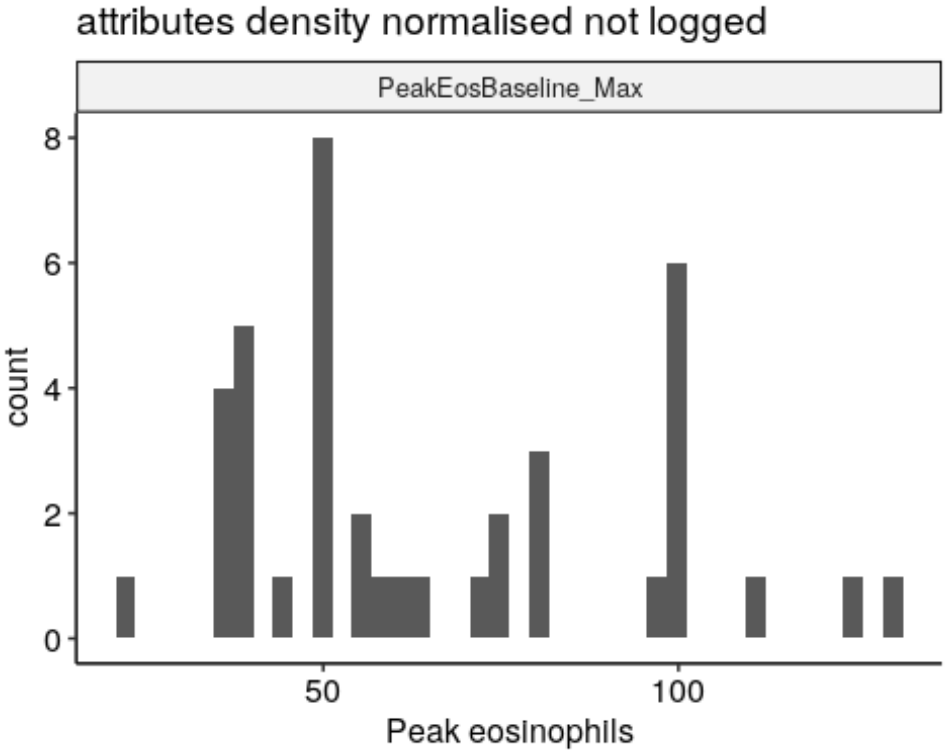


Figure 7: Shows the distribution of the peak max eosinophils with the peak eosinophils on the x axis and its frequency on the y axis. This plot shows that most values are approximately at 50 and 100 with the rest in between.

evaluation metrics

The most important metric to optimize for is the sensitivity. A false positive case is less harmful than a false negative case. This is because a patient that is classified as sick but turns out to be healthy after further inspection, experiences far less harm then those who are classified as healthy but continue suffering from symptoms.
The speed at which algorithm runs at isn't specifically optimized for nor is it measured on larger data sets. However it's main use case would be to classify individual patients which it can do near instantly in a common desktop environments.

Exploring algorithms

Model evaluation is done in the weka experimenter with five data sets and seven algorithms (default setting from weka 3.8.4) and four variations of the stacking meta algorithm with different settings. The best performing algorithm is believed to be the meta stacking algorithm with the correlation-based feature selection data-set. This algorithm has an accuracy of 47%. The meta stacking algorithm consists of random forest as its meta learner and the following algorithms are its ensemble members: J48, Random Forest, naive Bayes and logistic regression. Some algorithms like oneR and k nearest neighbor aren't used since they do not create a true model and therefore aren't useful in this study.

Attribute selection

There are different data sets created from the original data-set. which is logged and normalized. The different data-sets are created by using different attribute selection methods.
These methods are: Classifier attribute evaluation with j48. Done with the setting, leave one out turned on and with it off and with different rank cut off points.
And a data-set. created by correlation-based feature selection. The data set created from the Correlation-based feature selection yielded the best results and is used in the final model.
The attributes selected in this data set are: Neocate, B.Carbohydrates.gr, Gender, B.PUFAS.gr, B.Linoleicacid.gr, B.Calcium

links:

Java wrapper: https://github.com/jimjimvalkema/Predicting-severity-of-Eosinophile-Oesofagitis-with-machine-learning-/tree/master/app
Rmarkdown EDA log: https://github.com/jimjimvalkema/Predicting-severity-of-Eosinophile-Oesofagitis-with-machine-learning-/blob/master/EDA/Eosinofiele-Oesofagitis-dataset-EDA.Rmd

results

The resulting best performing algorithm is a meta stacking algorithm with random forest as its meta-learner and the following ensemble members: J48, Random Forest, naive Bayes and logistic regression. This model is trained on a data-set. that is created from correlation-based feature selection on the original data that is log two transformed, normalized with scaling normalization. and without the attributes that have more than 40% of their values missing.
This model can be used in a command line interface to predict a patient's eosinophils measurements in three categories: low,mid,high based on six attributes with 47% accuracy. This command line interface can process the raw data and can do the necessary pre-processing like normalization. and log transform. It can process csv, arf files and handle a single instance provided in the command line.

discussion

what if new insight with this analysis of ml algo? EOS might be a complex issue? param selection? No weight on false negatives? While the accuracy of the model created by this study isn't that high, it might improve when more data becomes available. There are also a lot more attributes in this data set that might yield a better model and give further insights. Finding the more valuable attribute requires further automation in processing and attribute selection.
It might also be interesting to see if a model can be created to predict the food that triggered the allergy which could help speed up elimination diets. However it is believed that this requires a different data set with more specific information about a patient's diet to do so. There are also a fair amount of missing values removed which might still be valuable because a missing value on its own can also convey information in some cases.

Conclusion

The resulting model is believed to be not accurate enough to be used in a clinical setting. It also found that the data set is too small to evaluate if a proper model can be created.
It is found that scaling by the standard deviation would be the most effective method for normalization for a future proof machine learning model. The data also benefits from a log2 transformation because it seems to be more normal distributed this way.
It is also found that some attributes like the different types of fat are correlated and that most attribute selection methods tend to prefer only one of the sub types of fat since each individual fat type do not convey a lot of information to those attribute selection algorithms.

project proposal

Further improvement of the accuracy of the model could be explored in the high throughput bio-computing minor. A more advanced algorithm like a neural network could be trained to possibly get a higher accuracy. Getting a larger data-set. would also be beneficial to prevent over-fitting. An automated method of selecting and pre processing attributes can then also be developed.

Developing a simple user interface might also be a great improvement if this algorithm is used in a clinical setting. This user Interface could provide a way for the user to provide the data and perhaps a simple way to edit it. Having a graphical representation accompanied with general statistics of the resulting classifications could help users that do not have a deep understanding of machine learning to interpret the output of the algorithm. Developing an application for desktop computing on platforms like linux,mac and windows would be most useful since mobile applications would be more work to develop while not gaining much usability given the context of its future use in a clinical setting. A web app could be useful however the app benefits from being verifiable open source when running locally. This is because researchers need to know exactly what program they are running in order for their work to be reproducible.

references

Eosinophilic Esophagitis:
http://www.pghclinic.com/wp-content/uploads/2019/05/56.pdf