Statistical and Machine learning - Individual assignment

Introduction

The objective of this project is to explain five machine learning algorithms, the main objective, approach, pros, and cons. In addition, to understand better the models and compare them, a benchmarking exercise will be explained by predicting for a particular dataset a binary target by using the five algorithms presented and comparing the results obtained.

I. Machine learning predictive algorithms

Logistic regression

This algorithm used for classification problems, models the probability that the dependent variable (y) belongs to a particular category. Given the fact that the expected output of this model is a probability (values between 0 and 1), the function used to model the target variable is a logistic function (or sigmoid function) (*Figure 1*) based on solving the optimization problem of *maximum likelihood*.

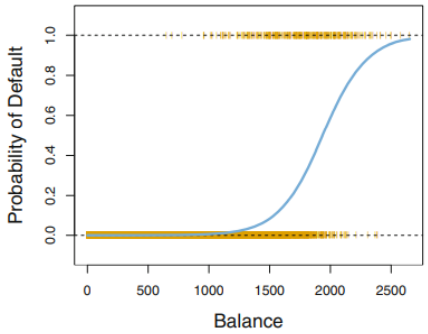

Figure 1. Example logistic function for a classification problem[1]

In order to estimate the probability, the approach of this method consists in estimating the logarithm of the odds using a linear regression model. The logarithm of odds is defined as follows:

$$\ln\left(\frac{p(x)}{(1-p(x))}\right) \qquad (1)$$

The odds are useful to represent the probability that the dependent variable belongs to the category selected as target. A small value of odds represents low probability and high value of odds represents high probability. As it was mentioned previously, the estimation of the probability follows the same principle of linear regression. For this reason, the β coefficients are estimated by maximizing the likelihood because this method is more appropriate than least squares (used in linear regression) considering that the estimated values must be between 0 and 1. The main principle behind this approach is finding the estimated probability that corresponds as closely as possible to the real target variable (0, 1). The likelihood function is defined as follows:

$$l(\beta_0, \beta_1) = \prod_{i:y_i=1} p(x_i) \prod_{i':y_{i'}=0} 1 - p(x_{i'}) \qquad (2)$$

[1] Gareth James, Daniela Witten, Trevor Hastie, Robert Tibshirani. (2013). An introduction to statistical learning : with applications in R. New York :Springer

Once the coefficients are estimated the prediction can be calculated using the formula: $\hat{p}(x) = \frac{e^{\widehat{\beta_0}+\widehat{\beta_1}x}}{1+e^{\widehat{\beta_0}+\widehat{\beta_1}x}}$ (3)

This method can also be used for problems with multiple independent variables or target variables with more than two categories, for the second scenario a few adjustments are needed but the main logic is the same; predicting a binary target comparing pairs of categories (one vs. one) or comparing each category with the rest (one vs. all).

In general, the logistic regression is a very useful model for prediction in classification problems, however, there are some limitations regarding high dimensional datasets. When the number of observations is smaller than the number of features the model will result in overfitting because each feature will represent a dimension and every observation of the training dataset can be represented with one dimension.

On the other hand, the advantages of logistic regression are notable in terms of interpretability and simplicity due to the fact that each coefficient estimated has a magnitude and direction that can be also interpreted in terms of the importance and relation of each independent variable with the target. This is also a simple model that can be easily implemented, calibrated, and updated in case of a decrease in the performance (AUC, accuracy, etc.).

LINEAR DISCRIMINANT ANALYSIS (LDA)

In the same line of the linear models there is a different method called Linear discriminant analysis. This model is also based in the assumption of the existence of a linear relation between the independent and response variables. But in this case the approach is modeling the distribution of the predictors separately in each of the response classes and using Bayes theorem (Conditional probability) to estimate the target probability. The Bayes theorem can be represented as follows, π_k represents the probability that a random observation belongs to the class k and f_k is the probability of (X=x|Y=k):

$$\Pr(Y = k|X = x) = \frac{\pi_k f_k(x)}{\sum_{l=1}^{K} \pi_1 f_1(x)} \qquad (4)$$

Compared with logistic regression, this method is more stable when the classes of the target variable are well separated or when the number of observations is small, and the independent variables are approximately normal distributed in each class. To estimate the probabilities, it is assumed that f_k follows a Normal distribution and the density function would be defined as follows:

$$f_k = \frac{1}{\sqrt{2\pi\sigma_x}} \exp\left(-\frac{1}{2\sigma_k^2}(x-\mu_k)^2\right) \qquad (5)$$

The previous assumption allows to estimate the mean and variance of the variables using the Normal distribution, and the predicted probability can be calculated by using the discriminant function:

$$\widehat{\delta_k}(x) = x\frac{\hat{\mu}_k}{\hat{\sigma}^2} - \frac{\hat{\mu}_k{}^2}{2\hat{\sigma}^2} + \log(\hat{\pi}_k) \qquad (6)$$

This function defines a decision boundary where $x = \frac{\mu_1-\mu_2}{2}$ (For 2 class target), to classify the observations in each class. In the following figure it is possible to observe an example of the decision boundaries defined for a target with two classes

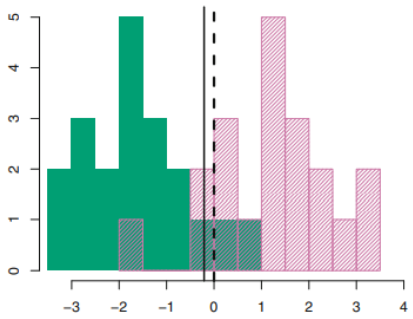

Figure 2 Decision boundary and histogram per class

The process explained previously can be summarized also in the use of information form independent features to create new dimensions that minimize the variance but maximize the distance between classes to classify better. In addition, LDA performs very well when the target classes are well separated and in cases where the target variable is not binary.

Given the fact that this method has a linear decision boundary it is easy to implement and is very useful to mitigate high dimensional problems because it creates new dimensions using the existent variables which can capture the information of the predictors but are represented in less predictors.

A final aspect to consider is the fact that the assumption of a normal distribution in the data can affect the results of the model in case this assumption is not ensured. This aspect can be treated by transforming the data to make it approximately normally distributed.

K-NEAREST NEIGHBORS

A third method useful to classify data is KNN, this is a non-parametric method which does not assume a particular distribution for f(x). In this approach the first part consists in identifying a K number of neighbors in the training data set for each observation in the test set. Then, it is possible to estimate the conditional probability for an observation of being part of a particular category by applying Bayes rule and classifying the observation in the category with the largest probability by using the formula:

$$\Pr(Y = j|X = x_o) = \frac{1}{k}\sum_{i \in No} I(y_i = j) \qquad (7)$$

The selection of K impacts the performance of the model given the fact that a small K gives more flexibility because the decision boundary is more flexible and it is possible to find more patterns in the data. Although, it is important to be careful because the selection of a small K can result in overfitting. For this reason, is always important to evaluate the performance of the model in the test and training datasets.

In the following figure it is possible to observe an example of the model and the definition of de decision boundary using a K equal to 3.

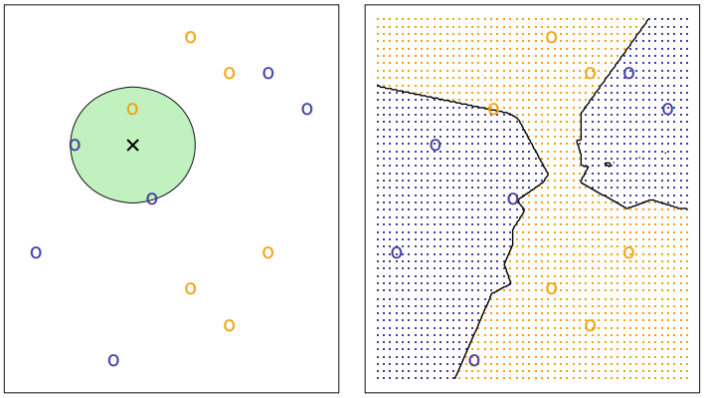

Figure 3 KNN example. Neighbors selection and decision boundary

Given the fact that this algorithm is based on the distance of the observations and the scale of each dimension (feature) impacts those distance it is very recommended to standardize the values of the variables to avoid bias in the case of high magnitude variables. This distance also, can be calculated using different methodologies, for example Euclidean, Manhattan or Minowski distance.

In general, this is a simple and easy to understand model, although it is also important to mention the fact that this algorithm can be affected by the number of features because with a large number of features the complexity of the model increases because the number of dimensions also increases. For example, in cases where the observations are close the distance between them can be affected by the existence of more dimensions and the result will be less accurate.

RANDOM FOREST

Decision trees are based on splitting the predictors into regions that reduce the RSS or classification error rate (in case of a classification problem) by predicting as response the mean of the response variable in the observations belonging to the same region.

Random forest is a decision tree-based algorithm that creates several decision trees on bootstrapped training samples in parallel but when a split is considered in a tree, the of predictors used is a subset of the total number of independent variables and in each split a different set of predictors is used as candidates. Typically, the number of predictors selected is the squared root of the total number of predictors, but this number should be selected according to the correlation of the independent. The reason for this approach is that in some cases there are very strong predictors that can lead to overfitting in the resulting model, this methodology leads to less variable and more reliable trees.

Once the decision trees are created, the final prediction is made by using the majority vote, which is the most common occurring class in the predictions of the trees for the given observation. Regarding this point, it is also important to mention the fact that the number of trees created is also a parameter defined by the modeler and can affect the results obtained.

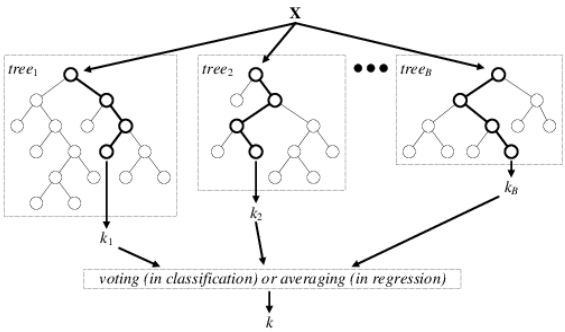

Figure 4 Example majority vote in random forest

This method is very stable in terms of the prediction due to the fact of the bootstrapping in the observations and the random selection of the initial features, for this reason the inclusion of new data will not impact the results obtained in an initial model. A second advantage of this algorithm is based on the type of features that can e included in the model. Given the fact that this is a decision tree-based algorithm it is possible to include categorical variables or missing values, which means that there is no need to encode or create dummy variables in the pre-processing of the data.

On the other hand, a downside of random forest is the complexity and processing time required to estimate a model given the inclusion of bootstrapping and the creation of multiple trees. This aspect can affect the interpretability of the resulting predictions.

SUPPORT VECTOR MACHINE

This method is based in creating a separating hyperplane using the mathematical definition of a hyperplane based in p predictors:

$$\beta_0 + \beta_1 X_1 + \cdots + \beta_P X_P = 0 \quad (8)$$

In addition, for a binary target each one of the classes can be represented with the labels 1 and -1, which helps to classify the observations per category combining the equation of the hyperplane with the Y class

$$y_i(\beta_0 + \beta_1 X_1 + \cdots + \beta_P X_P) > 0 \quad (9)$$

Given the equation 9 it is possible to create a classifier depending on which side of the hyperplane is located the observation to estimate. Another important concept to consider in this model is the distance from the point to the hyperplane because this gives an idea about the confidence of the prediction. In the case of a long distance, the class assignment is more reliable, but in the cases where the distance is close to zero the prediction has more uncertainty.

The svm algorithm maximizes the distance between the observations and the hyperplane. This objective function is called hinge loss and includes a cost regularization parameter to balance this distance and the loss. The cost parameter allows to play with the misclassifications and is related with the tolerance to this scenario.

To include more flexibility in the model it is possible to change the type of kernel in the model. This parameter changes the type of decision boundary obtained, for example linear, radial or polynomial. According to the kernel selected, the model will change the dimensions in the data to project a hyperplane correspondent to the kernel selected.

The svm classifier offers good predictions given its nature and flexibility, however, for this aspect it can also take a long time of training and is sensitive to the type of kernel selected, which means that the results obtained can have significant changes by modifying the type of kernel.

II. Benchmarking experiment

This exercise was made using a database related with direct marketing campaigns of a banking institution. This dataset includes various features regarding the financial situation of the clients with the products in the bank, the number of contacts with the client, demographic information, social and economic context features and the target variable *subscribed* which tells if the client was subscribed or not to the product offered during the campaigns. Thedataset was divided in train (80%) and test(20%).

A first step to conduct this experiment was analyzing the features, their scales, magnitude, and types. In the following table it is possible to observe some summary statistics of the numerical variables:

	age	pdays	previous	emp.var.rate	cons.price.idx	cons.conf.idx	euribor3m	nr.employed	campaign
count	15842.000000	15846.000000	15841.000000	15863.000000	15851.000000	15841.000000	15842.000000	15852.000000	15844.000000
mean	40.084207	961.897892	0.171012	0.095127	93.578654	-40.466189	3.639844	5167.612528	2.589371
std	10.423776	188.341702	0.500488	1.570787	0.580341	4.607290	1.727993	72.093729	2.831599
min	17.000000	0.000000	0.000000	-3.400000	92.201000	-50.800000	0.634000	4963.600000	1.000000
25%	32.000000	999.000000	0.000000	-1.800000	93.075000	-42.700000	1.344000	5099.100000	1.000000
50%	38.000000	999.000000	0.000000	1.100000	93.876000	-41.800000	4.857000	5191.000000	2.000000
75%	47.000000	999.000000	0.000000	1.400000	93.994000	-36.400000	4.961000	5228.100000	3.000000
max	98.000000	999.000000	6.000000	1.400000	94.767000	-26.900000	5.045000	5228.100000	56.000000

Table 1

In this table it is possible to notice that the variable pdays (days after the last contact) presents most of the values in 999 which means that the client was not contacted.

Regarding the number of missing vales, the maximum percentage is 1% which means that all the features have a considerable number of information.

Regarding the categorical variables, in the following table is possible to observe the distribution per category.

	job	
admin.	3985	25.1%
blue-collar	3556	22.4%
technician	2595	16.4%
services	1549	9.8%
management	1139	7.2%
retired	674	4.2%
entrepreneur	586	3.7%
self-employed	557	3.5%
housemaid	431	2.7%
unemployed	374	2.4%
student	304	1.9%
unknown	110	0.7%

	default	
no	12563	79.4%
unknown	3260	20.6%
yes	2	0.0%

	housing	
yes	8388	53.0%
no	7101	44.8%
unknown	348	2.2%

	loan	
no	13053	82.5%
yes	2425	15.3%

	education	
university.degree	4779	30.1%
high.school	3566	22.5%
basic.9y	2340	14.7%
professional.course	2009	12.7%
basic.4y	1660	10.5%
basic.6y	887	5.6%
unknown	617	3.9%
illiterate	7	0.0%

	day_of_week	
thu	3362	21.2%
mon	3309	20.9%

	marital	
married	9559	60.3%
single	4443	28.0%
divorced	1809	11.4%
unknown	30	0.2%

unknown	349	2.2%

	contact	
cellular	9980	63.0%
telephone	5849	37.0%

wed	3146	19.8%
tue	3052	19.3%
fri	2983	18.8%
	poutcome	
nonexistent	13764	86.8%
failure	1551	9.8%
success	546	3.4%

Most of the clients do not have loans, are married and were not included in previous campaigns. Regarding the missing values, the categorical variables are also well informed and can be used in terms of data existence. In terms of the target variable. The 11.2% of the clients were subscribed after the campaigns.

As a data processing step, the observations with missing values were treated as follows:

- **Numeric variables:** Replaced with the mean of the variable (in the training dataset) and a column to indicate this replacement was created for each column
- **Categorical variables:** A new category was created for the observations with missing values in each variable.

After this process, the variable pdays was replaced with a dummy variable given the fact of the low variance (most of the values were 999). The variable age was also treated by creating age categories divided by quartiles according to the distribution in the training set.

As this experiment will include a KNN model, the numeric variables are normalized to avoid bias in terms of the distance calculated for the model. The next step was the creation of dummy variables for the categorical feature taking into a count that some of the algorithms that will be tested cannot handle categorical variables (i.e. logistic regression).

Models benchmark

a. Logistic regression

In order to include relevant variables in the model, the Fisher score is calculated for all the variables described previously. The results for the most important variables are:

predictor	fisherscore
nr.employed	0.748816
euribor3m	0.714965
emp.var.rate	0.677628
contacted	0.473696
poutcome_success	0.454186
previous	0.398336
poutcome_nonexistent	0.384037

Table 2.

The previous variables were chosen to fit the logistic regression algorithm using the training dataset and after the fitting process the results obtained were:

	coef	std err	z	P>|z|	[0.025	0.975]
nr.employed	-3.2548	0.184	-17.696	0.000	-3.615	-2.894
euribor3m	1.0163	0.232	4.384	0.000	0.562	1.471
emp.var.rate	-1.3304	0.232	-5.738	0.000	-1.785	-0.876
contacted	1.3262	0.285	4.653	0.000	0.768	1.885
poutcome_success	0.7000	0.278	2.517	0.012	0.155	1.245
previous	-0.6540	0.359	-1.820	0.069	-1.358	0.050
poutcome_nonexistent	0.3119	0.082	3.800	0.000	0.151	0.473

Table 3.

In the previous summary it is possible to observe the value of the coefficients and the p-values for the significance of the independent variables. Under a significance of 5% all the variables are significant to predict the subscription except for the variable previous which can be removed to improve the results of the model. Although, with an alpha of 10% is significant and for this reason it was not removed from the exercise. Regarding the coefficients, the variables related with the social and economic context have higher impact in the target. For the number of employees and employment variation rate the impact is negative which implies a negative impact in the odds that can be reflected as a decrease in the probability of subscription.

In terms of performance, this model has an accuracy of **89.1%** and an AUC of **75.5%**. The ROC curve can be observed in the following graph and the result of the kfold cross-validation using 5 folds.

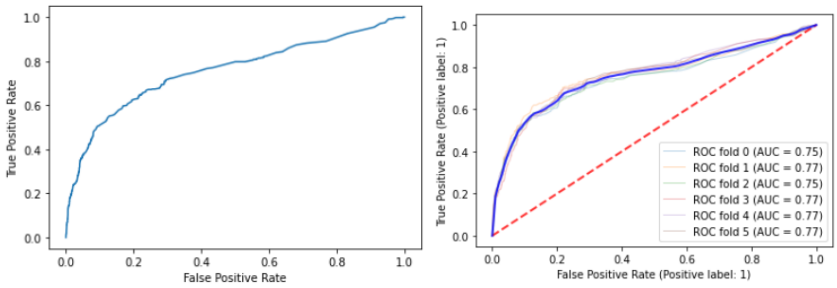

Figure 5.

b. Linear Discriminant analysis

Taking into account that this model is useful for dimensionality reduction, all the features were used as input for the model. Considering that the target variable has 2 classes, the number of components obtained as result after fitting this model is equal to 1 (# classes -1) and this indicates that the variance captured by this component is 100%.

In terms of performance, this model has an accuracy of **87.9%** and an AUC of **75.7%**. The ROC curve can be observed in the following graph and the result of the kfold cross-validation using 5 folds.

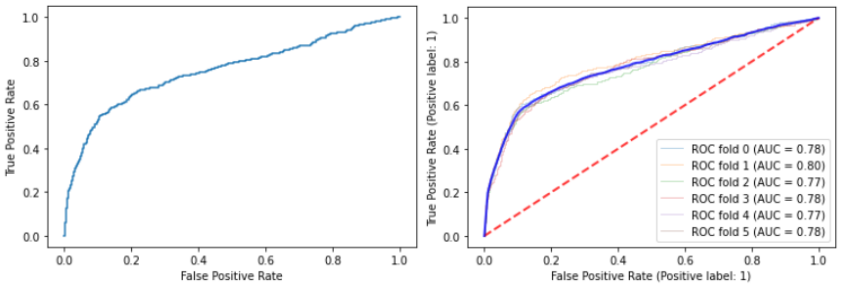

Figure 6.

c. KNN Classifier

Considering the fact that a high number of irrelevant features can affect the performance of this model, a set of features was selected to fit this model. The features selected are base in the correlation with the target variable and those with the strongest correlation are: *'nr.employed','euribor3m','emp.var.rate','contacted','poutcome_nonexistent','contact_telephone','poutcome_success','previous','contact_cellular'.*

The number of K neighbors was evaluated for different values and the one with best results in the test set is k = 10.

Given the previous definitions, the model was fitted using the training dataset, this model has an accuracy of **88.9%** and an AUC of **74.8%**. The ROC curve can be observed in the following graph and the result of the kfold cross-validation using 5 folds.

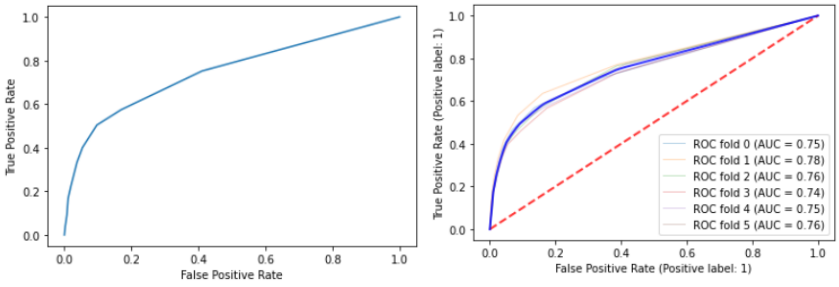

Figure 7.

d. Random forest

For this model the feature selection was based on the results of the feature importance obtained after running the algorithm with all the variables given the selection resulting from the generation of all the trees and the random selection of features in each node. The results obtained were:

nr.employed	0.172447
contacted	0.157440
euribor3m	0.125909
poutcome_success	0.110840
emp.var.rate	0.087674
cons.conf.idx	0.081895
cons.price.idx	0.067843
previous	0.046156
poutcome_nonexistent	0.030244
age	0.022101

After this process, the random forest was fitted using this set of variables setting the random seed to obtain replicable results given the random aspect involved in the iterations of the algorithm.

In terms of performance, this model has an accuracy of **88.9%** and an AUC of **75.7%**. The ROC curve can be observed in the following graph and the result of the kfold cross-validation using 5 folds.

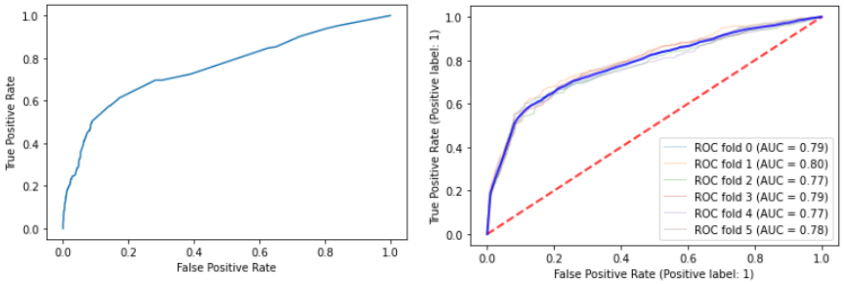

Figure 8.

e. SVM Classifier

Regarding the feature selection, the variables chosen to fit this model were the same resulting form the Fisher score calculated in the logistic regression. In addition, the parameter related to the kernel was modified and the best results were obtained with a linear kernel and the value of the cost parameter was decreased to 0.5 to allow more miss-classifications in the resulting model (Default C = 1).

Given the previous definitions, the model was fitted using the training dataset, this model has an accuracy of **88.8%** and an AUC of **70.3%**. The ROC curve can be observed in the following graph and the result of the kfold cross-validation using 5 folds.

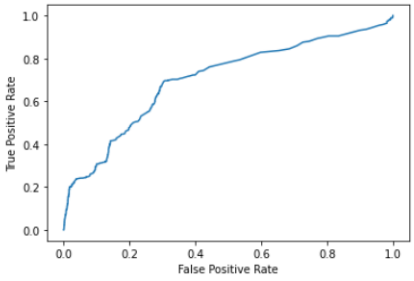

Figure 9.

Conclusions

Considering the results obtained by fitting the different models presented the first aspect to discuss is the performance of the predictions for every algorithm. A comparison is presented in the following table:

	Logistic regression	**LDA**	**KNN**	**Random Forest**	**SVM**
Accuracy	89.10%	87.9%	88.9%	88.9%	88.8%
AUC	75.50%	75.7%	74.8%	75.7%	70.3%

Considering the results of the previous table it is important to consider the percentage of incidence in the target variable which is 88.8% for no subscription. Comparing this value with the accuracy it is possible to see that the improvement in accuracy is small for most of the models. In terms of AUC the results obtained using logistic regression and random forest have the highest values which means that the predictions are more accurate.

An interesting aspect to consider is that the logistic regression is one of the simplest methods but the results show how can still give in some cases better results than more sophisticated methods. A possible explanation for this result can be based on the nature of the relation between the independent variables and the target which can be linear and for this reason well explained by the logit.

Additional References

- https://www.knowledgehut.com/blog/data-science/linear-discriminant-analysis-for-machine-learning
- https://medium.com/mlearning-ai/linear-discriminant-analysis-lda-maximum-class-separation-1c3e2f66d846
- https://www.knowledgehut.com/blog/data-science/knn-for-machine-learning
- https://medium.com/diogo-menezes-borges/random-forests-8ae226855565
- https://www.knowledgehut.com/blog/data-science/bagging-and-random-forest-in-machine-learning
- https://www.knowledgehut.com/blog/data-science/support-vector-machines-in-machine-learning