IÉSEG
SCHOOL OF MANAGEMENT

EMPOWERING CHANGEMAKERS FOR A BETTER SOCIETY

Statistical and Machine Learning

Ahmad Omar Nakib.

Table of Contents

Introduction

The aim of this paper is to go thoroughly through 5 different machine learning delve into the details and understanding of how they operate, their advantages and disadvantages. To understand how these models work, we must understand what statistical learning is and what we want out of it. Statistical learning is the application of building a model to extract either an outcome (or more accurately a prediction) or an inference based on a set of data points or observations that are given.

To proceed with the application, we deal with sets of data called input variables, predictors, independent variables or features. These input variables are denoted as the X of our function to be applied. On the other hand our Y is denoted by the output and is called response or the dependent variable. Mathematically, if we assume that there is some relationship between our X and our Y we can draw the general form of this relationship as $\boldsymbol{y = f(x) + \in}$**.** Where f is an unknown function X1…Xn and ∈ is an error term having mean error = 0.

Back to our first main objective, predicting the outcome based on given data we stumble across a requirement when looking at the previous general formula. This requirement is the function the f in f(x) which connects our input variables to the output variables. Hence, we now have a clear objective, estimate f. Note: *our function may be involved in more than one input variable and force us to go into higher planes.* We need to estimate f for predictions as the sets of X are available but Y isn't, this case calls for the use of the formula $\hat{y} = \hat{f}(X)$ where $\hat{f}$ denotes the estimate of the unknown f and $\hat{y}$ represents the prediction of Y based on $\hat{f}$. In this scenario $\hat{f}$ is a black box meaning we are not concerned with what it is as long as it gives us accurate predictions of Y. Another issue we must take into consideration is the error, since our objective is the accurate prediction of our outcome and looking back to our simplified formula, there is one more factor to take into consideration and that is the error ∈**.** The book "An introduction to statistical learning" refers to two types of errors as reducible errors and irreducible errors. Reducing the first type of error is possible by selecting the best possible model to apply for our estimation and is the objective of this paper (comparing 5 different models and selecting the most appropriate) however, even when choosing the best possible model, we are left with the second type of error which is the irreducible error. This error will always be there because Y is a function of the denoted error in the general formula which cannot be predicted using X. As stated previously statistical learning can also be used to make an inference based on the data points we have. An inference is the interest in assessing the relationship between Y and X and monitoring the way Y is affected by X. In this case we cannot treat $\hat{f}$ as a black box model since we need to know what form it takes. Inference also looks into things like uncovering which predictors are associated with what response, the relationship between each response and each predictor and whether a relationship is too complicated for a linear equation or can it be used to accurately summarize.

Most statistical learning methods used to estimate f are either parametric or non-parametric, parametric meaning we make an assumption about the form of the function such as f is linear and then we fit or train the model. This approach reduced the complexity to estimating a set of parameters based on our assumption. Non-parametric approach does not make an assumption of the shape of the function, instead it estimates as closely as possible the shape of f where it gets close to the datapoints. Each one of these two approaches has its own advantages and disadvantages and is appropriate for a specific problem.

It is also important to note that selecting the appropriate model to estimate the f can be challenging as we have to look at accuracy measures to evaluate said model. In this sense we tend to look at MSE (mean squared error). In real life we do not know what the function looks like, and we have to factors or trade offs we know about when implementing a model. First of all, we have to know our model's flexibility, which means how restricted it is in terms of jumping from one data point to another. Some models are very restrictive such as the linear regression model which will be discussed as part of the 5 predictive models, and some are more flexible. There is a trade off though, when we increase flexibility of a model, we tend to decrease our bias however we increase our variance (variance being the shape of f given more or different data). To get the least possible MSE, we have to find a sweet spot as bias decreases faster than the increase of variance when we increase flexibility but at one point variance starts to increase rapidly while bias stops being affected. Flexible models perform well when our data points are nonlinear whereas restrictive models are better when they are.

Linear Regression Model.

Linear Regression is a form of supervised learning and considered one of the most basic approaches to it. It is useful when used to predict a quantitative outcome. Linear regression can be used to resolve important aspects when it comes to assessing the relationship between the predictors and the output, in particular it can be used to identify if there is a relationship in the first place, the strength of that relationship, which of our predictors contributes to the outcome (this will be addressed in depth as we might have more than one input variable and not all of them might have a relationship with the outcome), the accuracy of our prediction, the accuracy of the effect of each predictor, the form of the relationship (such as linear or nonlinear) and the interaction effect.

The first form to be discussed is the simple linear regression, where it predicts a response in terms of a predictor X and takes the form $y \approx \beta 0 + \beta 1X$ where out betas are unknown that represent the Y intercept and the slope of our line. These are called the model coefficients; we have to use our training data (data used to train the model) to estimate $\widehat{\beta 0}\ and\ \widehat{\beta 1}$ and we use these coefficients to produce a prediction denoted as $\hat{y} = \widehat{\beta 0} + \widehat{\beta 1}X$. Now a deeper look into the coefficient estimates, we have to estimate betas that will fit our data properly so that the line we have is as close as the n (number or data points) as we can. The approach most taken is reducing the least squares this approach involves minimizing the RSS. (Residual some of squares). After obtaining the betas we can assume that increasing X by association will have a $\widehat{\beta 0}.X$ and $\widehat{\beta 1}.X$ increase/decrease in the result. Assessing the accuracy of these coefficients involves looking at the error as the true relationship probably isn't fully linear but our betas represent the least possible error for a linear regression (least square line). Now we have created a statistical hypothesis, a null hypothesis where we assume no relationship and a hypothesis where we assume there is. To determine whether we accept or reject this null hypothesis we must look at the p-value, if the p-value is small enough it means the probability that a chance association is very low and that there is a relationship, so we reject the null hypothesis and proceed. (Note: p-value in the range of 1%). We move on to assess the fit of our data to the model and the underlying errors, we look at our RSE which is the estimate of the standard deviation of the error described in our general formula. This number will indicate the average deviation of the datapoints from the true regression line even if the model and our coefficients are perfect. We are interested in minimizing RSE as much as possible. Next, we look at R^2 which is another measure to assess our fit. It is independent from Y and is between 1 and 0, it is = $1 - \frac{RSS}{TSS}$, TSS being the sum of squares, if it is close to 1 it indicates that large amount of variability has been explained by the model whereas a number close to 0 indicated that we have high error, or the model is wrong.

Multiple linear regression, after discussing the simple version the multiple LR comes in handy when we have multiple predictors, it can help us indicate which predictor is affecting our Y and which might be there but has no relationship. To fit a multiple LR, we give each predictor a separate slope so that it takes the shape $y = \beta 0 + \beta 1X1 + \beta 2X2 + \cdots + \beta pXp$ where p is the number of predictors. We follow the same steps and estimate the coefficients for this model using least squares. In this case we have to check our betas but the difference here is that we have more than the simple LR so our null hypothesis would be if all betas equal 0. We test this null hypothesis using the F-statistic where if it is close to 1 we accept the null hypothesis and we assume no relationship and if it is greater than one (could be much greater) we reject the null hypothesis and assume a relationship. To assess our F-statistic we take a look at n and p where when n is large a F-statistic a bit larger than 1 would suffice and vice versa, if n is small, we need a large F-statistic to assume a relationship. After this we have to look at the p-value of the predictors to determine the association of those predictors with the outcome. After our assessment of the model fit and predictors, we then look at the prediction itself, the prediction by the model in a confidence interval to quantify the uncertainty associated where we assume 95% certainty that the true value of f(x) (recall the general objective) takes the form in this interval. We use prediction intervals to quantify our ‘y’ value or result also, the 2 intervals are usually different than each other indicating the uncertainty range. Advantages of taking this approach are the facts that it is easy to implement, and it is highly interpretable in terms of coefficients. Its complexity is less given that the data has a linear relationship, though overfitting might occur we can overcome it by using dimensionality reduction techniques or cross validation. Though LR has advantages, it comes also with its disadvantages depending on the application, such as the effect of outliers on the outcome, it also assumes a linear relationship between the dependent and independent variables (this might not always be true) and it looks at the mean of the dependent and independent variables which might not always be an accurate description of the data.

Logistic Regression Model.

A logistic regression model makes the assumption that the values we are trying to predict, or our response 'Y' is categorical which means it is qualitative. This type of method is used for classification and predicts the probability for each category in order to assign it. An example would be predicting if the customer will subscribe to a service or not (Assignment task 2) here we need our model to give us a binary prediction 'yes' or 'no'. Logistic regression works by modeling the probability that the outcome belongs to a certain category. In this model since the formula $p(X) = \beta 0 + \beta 1X$ can be greater than 1 or less than 0 for very large and very small numbers respectively, it doesn't make sense (we can't have a negative probability or greater than one). Here we must look at the logistic function $p(x) = \frac{e^{\beta 0+ \beta 1X}}{1+ e^{\beta 0+ \beta 1X}}$. This model forms an S shaped curve where no matter, the probabilities will get very close to 1 and very close to 0 but never cross the boundaries. As stated previously the coefficients are vital to make our predications and are depicted by the betas in the logistic formula, we must estimate them based on the data used to train the model. In this case we use maximum likelihood instead of least squares to fit this model as we are looking for probabilities close to 1 and close to 0. In Logistic regression assessing the model accuracy is quite similar to linear regression, we look first at the z-statistic which indicates similarly to the t-statistic evidence for or against rejecting the null hypothesis (no dependence). A small p-value also indicates the existence of a relationship between our predictors and the outcome. Once we have an estimate of the coefficients the model then calculates the probabilities where the study is based on a threshold, meaning if the probability of an event occurring crosses the threshold, we predict that it will happen and if does not cross we indicate that it will not. In this model we can also use qualitative predictors that can help us estimate the results, in fitting the model we create a dummy variable that takes a binary value for example where 1 indicates a certain characteristic and 0 indicates its absence. We then look to see if the coefficient of this variable is positive, and the p-value is statistically significant to identify if this variable is important for our prediction. If so, we can assume that the presence of this characteristic increases the odds that the event will happen than the ones without the characteristic in question.

A valid question to ask is, what can we do if we have multiple predictors, and we need the same binary response. To tackle this problem, we take a similar approach to the one we took in linear regression and that is implementing a multiple logistic regression model. In this case we use the following formula $\log\left(\frac{p(x)}{1-p(x)}\right) = \beta 0 + \beta 1X1 + \cdots + \beta pXp$ where X to Xp are our multiple predictors, We can now look to see if there is an association between these predictors and the result we are seeking by again looking at the p-value of each of these predictors and assessing their significance. After identifying the associations, we then take another look at the coefficients to see what predictors are leaning towards what result (the existence of characteristic A indicates a higher probability of occurrence than its absence for example). The implementation of the multiple logistic regression model can help us identify critical details in making a decision, as an example, the presence of characteristic A in a normal logistic regression might indicate the occurrence of the event. However, after checking the results of the multiple logistic regression we might have a reason for this scenario, where people with characteristics A might also tend to another characteristic B, and this B characteristic is what is driving the occurrence of the event. To simplify in this case B is driving the event but we would not have this full view before checking in depth. This could be indicated as a con of implementing logistic regression as preforming it on only one predictor might mask the effect of another relevant predictor. Though the multiple logistic regression approach could also be used to assess more than 2 responses, meaning indicate the outcome of 3 classes such as outcome 1,2 and 3, in practice it is not used often for that purpose. Other models are used specifically for that case in real world practices. Logistic regression is a great tool for classifications and has many advantages as it has disadvantages. Logistic regression is easy to implement it is highly interpretable and efficient to train. It also does not make any assumptions on how the classes are divided, when it comes to the coefficient it can tell whether the association in terms of coefficient is positive or negative. The disadvantages of logistic regression include the overfitting scenario, if observations are less than features. Like the linear regression suffers from assuming linearity of the data even if in reality, it is not a linear relationship. As stated previously obtaining a multi-outcome (more than binary) is less efficient than other models such as LDA.

Decision Trees

Decision trees are applicable to both regression and classification scenarios described earlier in this report. Looking first at the regression application, the logic of a regression tree is fairly simple, as it looks at a value and sets a cutoff for that value (observation) in order to categorize, meaning if age is greater than 30 it assigns it to a branch of the tree and in contrast if age is less than 30 it assigns it to another branch of that tree. In order to make a prediction, say the salary, the algorithm looks at the mean response of those observations. Since now the tree is branched, it will look at the mean response of those observations less than 30 years old as a category and will look at the ones over 30 as another category. The general concept of trees is recursive, and its simplicity can be overstated as each branch could be divided into subbranches too, we will illustrate this as number of projects handled where we can now subdivide again these branches into 2, greater than 10 and less than 10 projects. Predicting using stratification implies the division of the possible values into distinct regions where there is no overlapping and each observation, we make the same prediction based on where this observation lies (what cutoff) it is the mean response of the training observation. Taking an example where we split 2 region and the response mean of the training observations of first region is 30 and the response mean of the second region is 60, when we have an observation that belongs to the first region based on the cutoff 30 and for another that belongs to region 2 it will be 60. Now our goal is to minimize the RSS where $\sum_{j=1}^{j} \sum_{i \in Rj}(yi - \hat{y}rj)^2$ where $\hat{y}rj$ is the mean response in this j region. Since we cannot take every possible partition into a defined j region we take an approach known as *greedy* because we look at the best possible split at the time being ignoring steps that might be better in a future step. This is know as recursive binary splitting and in order to do that we select the predictor Xj and a threshold where splitting this predictor will lead to the greatest reduction in RSS where Xj takes a value less than the threshold. We repeat this process to find the values of the threshold and the best possible predictor in each of the created region where RSS is minimized, the only difference is that now we work on each new region meaning we split the predictors of those distinct regions. We continue doing this for each region hence, it is recursive until we reach a defined point such as 10 observations remaining. Once we reach this point, we predict the response of the test observation using the mean of the training depending on which region it belongs to in the newly formed tree. This approach could potentially lead to overfitting of the data so tree pruning could be used to safeguard against it at the cost of higher bias. Tree pruning the is the process of growing the tree and then cutting it to create a subtree that creates the lowest error rate.

Creating a classification tree follows similarly to the previous case, the difference however is that here we are looking at qualitative response. The predicted response for an observation belongs to the most recurring class of training observations in its designated region. Here we are interested in looking at two things, the class prediction indicating a terminal node and the class proportions of the training observations that belong to this region. In regression trees we also use recursive binary splitting the difference, however, is that we need to look at classification error rate instead of the RSS since we are trying to classify where this observation belongs. Classification error is not as suitable for assessment as the Gini index or the entropy, hence they are used in order to assess the splits when building this classification tree. Since we are dealing with qualitative values in this scenario (such as Sex, Education, and other binary/qualitative values) thus when we split, we assign a qualitative value such as male to a branch and female to another branch instead of a threshold in a continues situation. Another aspect we have to look at in classification trees are node purity since some splits might have the same predicted value. Meaning having an answer = yes on both sides. However, the difference is that 100% of the answers for a given qualitative response in 1 node can be ‘yes’, but the other node only has 30% ‘yes’ answers. Certainty in the region with 30% ‘yes’ answers is much lower than the other region, this split improves 2 very important aspects of assessing classification trees, the Gini index and the entropy as both are sensitive to node purity. Decision trees have their advantages, they are very simple to explain and interpret, they are also logical and similar to decision making processes. They also are easy to use when it comes to qualitative variables as no real need for dummy variables is present. However, they also have their disadvantages over other models, they are not as accurate as other models and changes in the data can hugely affect the outcomes when implemented. There are applications however that increase the reliability of decision trees such as boosting and random forests which are our next topics of discussion.

Bagging and Random Forest

To explain bagging and random forests we have to keep in mind that we are still working with decision trees. Recall from the previously explained model that trees suffer from an issue of high variance, thus if we take a dataset split it randomly into training and test sets and apply our tree, if we do this again with a different random split our results will be different. Bagging is a method to reduce the variance by bootstrapping, meaning it takes repeated samples from one training dataset, it generates multiple different bootstrap training datasets that finally result in an average prediction. Bagging is useful for many models but in particular it is useful for regression decision trees since they suffer from this high variance issue. To apply it on a tree, we generate X regression trees using X different bootstrapped training datasets as discussed earlier and get an average of the prediction results. We do not prune these trees we keep them as is so that the bias is low, but we have high variance in between them. After we average these trees, we would have dealt with the variance problem while keeping bias in check. This results in improvement in accuracy. We can also apply the concept of bagging on classification trees too. Recall that in regression trees we asses the situation by looking at the classes the qualitative values belong to. So, after growing multiple classification trees, we simply look at the class that is most common across the many trees we have grown among these predictions. To test the test error of a bagged model, there is no need for cross validation, or a validation set approach. We can look at the out of bags as each bagged tree takes 2/3 of the observations, we use the remaining out of bags to predict the response of the ith observation. We then average the prediction responses obtained to get one single prediction which in turn gets us one final out of bag prediction. Then either MSE or classification error, for regression and classification respectively could be obtained. This error is a good estimate for the error since we did not use the out of bagged observations to fit the model. For bagging, the benefits are clear, and its advantage is the gained increase in accuracy and lower variance however, it does have its disadvantages as it is less interpretable. RSS and the Gin index could still be used to assess the importance of the predictors though.

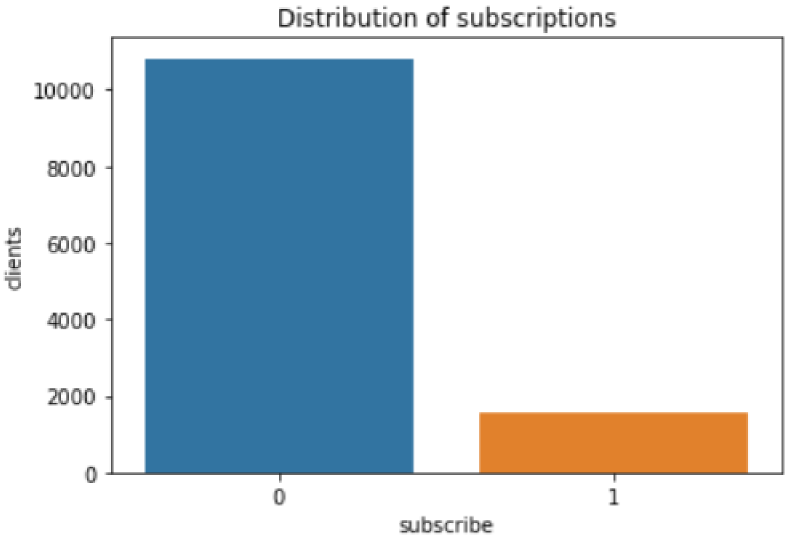
Random forests provided more improvement when compared to bagging as it involves decorrelating the trees. We first follow the same concept of bagging by creating multiple decision trees however, in those trees every time a split happens, a random sample of predictors are used from the full set or predictors. This split is allowed to use one out of all those random predictors. At each split in the tree, a new sample of the random predictors are chosen from the total set of predictors, typically we take the square root of total predictors as a random set. At the beginning of the explanation, we said that random forests cause decorrelation when compared to bagging and this is exactly the reason. Instead of having many similar trees, random forests use random predictors hence, the trees are not correlated with each other. This decorrelation allows the random forest to further decrease variance when predicting as a subset of all predictors are used for each split for each tree respectively. When it comes to the disadvantages of random forests, complexity comes to mind. Another issue which is a derivative of its complexity is a longer training period for random forests.

Boosting

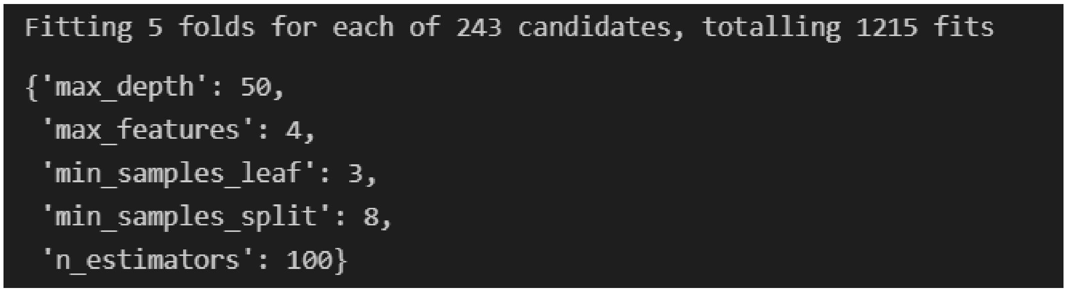
Boosting is yet another method to improve the performance of decision trees. Similar to bagging, boosting could be used with other statistical learning models for both regression and classification. To discuss boosting we have to remind ourselves of how bagging works first, which is by using bootstrapping, creating multiple decision trees using multiple copies of the same training dataset then merging these trees to get one predictive model where each tree is built independently on its bootstrap dataset. Boosting works in a similar fashion but there is a difference, the trees in this case are sequential where each tree is built on the previous copy of the tree. Another major difference is that boosting does not rely on bootstrap sampling rather, it works with modified versions of the main dataset. To avoid the overfitting issue that plagues normal decision trees, applying a boosting method allows the slow process of learning by fitting a decision tree with the residuals. To expand on boosting we must look at the process and the algorithm behind it. The algorithm takes proceeds in a series of steps. This means we use our current residuals instead of using our outcome Y as a response. To proceed we apply this decision tree on the fitted function and update the residuals First, we set $\hat{f}(x) = 0 \; and \; ri = i$ in our training set. Secondly, we fit a tree $\hat{f}^b$ having d splits and d+1 terminal nodes using the training dataset. We then shrink the tree to create a new one implying $\hat{f}(x) < -\hat{f}(x) + \lambda\hat{f}^b(x)$, we then update the residuals as discussed $ri < -ri - \lambda\hat{f}^b(xi)$ and we finally output our boosted model. This process is repeated B times to generate the multiple small trees. These trees used are small with a small number of terminating nodes. This will improve our $\hat{f}$ however this process is slow since the shrinkage (λ) causes different trees to use the residuals. For boosting classification works in a similar way. When it comes to tuning boosting has 3. The first tuning parameter is cross validation to select B. We must understand that boosting as opposed to bagging and random forests is prone to overfitting especially when B is too large (number of recursions that cause trees to be created) and thus cross validation can be beneficial. λ which is the shrinkage parameter controlling the rate at which our model learns, when it is too small, we require a large B to get an acceptable performance out of the model. The number of splits in each tree d (see the algorithm described previously) controls the boosted model's complexity as a high value means a high value of splits in each tree.

Conclusion

To conclude this research paper and what we discussed, it is very important to distinguish what type of problem we are facing to assess the solutions we can use. We have discussed only 5 approaches out of many more but the concept of how to tackle the problem remains the same. We have to look at the situation and ask ourselves, is it a regression problem or is it a classification one. Moving on to the application of task 2, we started of by the preprocessing steps necessary to fit our models properly such as dealing with missing values and other values such as 'unknown' observations. This was followed by the creation of new features such as the creation of age groups. Next, taking a look at the distribution of our target variable which is in this case the subscriptions in a bank dataset. The distribution looked as follows



We see that our distribution is skewed towards the non-subscribers. Then we proceed by scaling our data to accommodate for outliers. After that we collect our categorial variables in order to encode them by using pythons dummies function. After exploring our data making sure there are no null values remaining we set our target variable to ‘subscriptions’ as that is what we are trying to predict. We are now ready to split the data into training and test sets in an 80, 20 formation. We set up the evaluation scheme for ACC,PREC,REC,F1 and AUC. Now we are ready to fit our 5 models, since we are dealing with a classification problem, and as discussed earlier in this paper linear regression is not as efficient as other models so it has been substituted with KNN model. The implementation of decision tree, random forest, logistic regression, KNN and gradient boosting is the next step. We evaluate these models based on the results of the estimators previously mentioned. Since now the bench mark has taken place we need to make our models better, to do this we have to run a grid search to obtain the best parameters to be placed in the model. A grid search for random forest for example results in the following parameters.



```
Fitting 5 folds for each of 243 candidates, totalling 1215 fits

{'max_depth': 50,
 'max_features': 4,
 'min_samples_leaf': 3,
 'min_samples_split': 8,
 'n_estimators': 100}
```

Since we now know what the optimal parameters are, we can implement them on the model. The same is done for the rest. Our prediction results in

Sources

Web sources:

Statistical Learning Theory. Introduction: | by Ken Hoffman | The Startup | Medium

https://www.geeksforgeeks.org/

The Professionals Point: Advantages and Disadvantages of Random Forest Algorithm in Machine Learning

Book sources:

"Introduction to statistical learning" – Gareth James, Daniela Witten, Trevor Hastie and Robert Tishirani.