Applying data mining techniques in order to predict the willingness of a person to volunteer or not

1st Arthur O'Leary
StuID: 18190502

2nd Cem Yilmaz
StuID: 18191681

3rd Michael O'Leary
StuID: 18201881

ABSTRACT

The objective of this study is to better understand what makes people more inclined to volunteer. More specifically, the aim of this study is to determine data mining technique are best suited to predict whether or not a person would volunteer based on specific independent variables. With this in depth understanding, enterprises can apply understandings for business applications, such as in hiring employees. Three separate data mining techniques have been applied to predict whether or not someone would volunteer. From the results gathered, it is evident that the KNN classification technique had the highest accuracy rate followed by logistic regression, and thirdly decision trees. Even though three separate techniques were used to formulate a predictive model to determine whether or not someone would volunteer, it has been concluded that more research and testing is required to generate better performing data mining models.

I. INTRODUCTION

'Volunteering is any activity in which time is given freely to benefit another person, group or cause. Volunteering is part of a cluster of helping behaviors, entailing more commitment than spontaneous assistance but narrower in scope than the care provided to family and friends.'
At a time of social upheaval across the world where public services are being cut by right wing leaning governments, the concept of volunteering has never felt more important. It benefits not only communities and people in need, but has also been proven to benefit the volunteers themselves - it increases confidence and the ability to adapt to new environments and situations. It increases a sense of social responsibility. It also creates a sense of community and embeds the volunteer in the fabric of society
'Positive effects are found for life-satisfaction, self-esteem, self-rated health, and for educational and occupational achievement, functional ability, and mortality. Studies of youth also suggest that volunteering reduces the likelihood of engaging in problem behaviors such as school truancy and drug abuse.' (Annual Reviews. (2019). Volunteering)

There are varying degrees of competency in volunteering, from specifically trained such as in medicine or education, or on a needs-as basis, such as responses to natural disasters. Although it is mainly associated with altruistic, socially engaged young people, individuals between the ages of 35 and 54 are the most likely to volunteer their time, while as much as 47% of people aged 55 to 64, 43% aged 65 to 74, and 37% over the age of 75 volunteered in some capacity in a census taken in the United States. Volunteers are also more likely to donate to charity, and to have a 27% better chance of gaining employment. (US Bureau of the Census, 2019)

The purpose of our predictive modelling on this dataset is to give us a better understanding of the mental attributes associated with someone who volunteers versus someone who does not. This could lead to a more targeted approach when attempting to get people active and involved in the realm of volunteering.

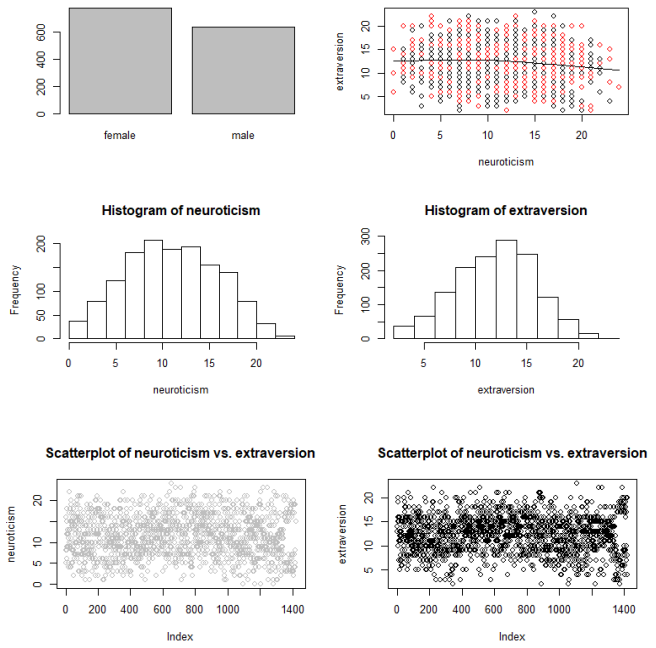
The Eysenck personality is the basis for the differing attributes among participants in judging how likely a person is to volunteer or not.

II. DATASET

For the purposes of this assignment we decided to apply statistical modelling to the Cowles and Davis's Data on Volunteering. This dataset was derived from a study conducted to discover the personality traits of people who are more or less likely to be volunteers based on scales of extraversion and neuroticism. The measure of extraversion and neuroticism was based on the Eysenck personality inventory, which is a self-report instrument designed to measure two central dimensions of personality, extraversion and neuroticism. This classification is comprised of yes/no items and yields total scores for extraversion and neuroticism as well as a validity score (e.g., Lie Scale). Individuals are generally classified as "high" or "low" on these two dimensions (Bodling and Martin, 2011).

The two dimensions further describe four quadrants of human personality;

1. Stable extraverts (sanguine qualities such as outgoing, talkative, responsive, easygoing, lively, carefree, leadership)
2. Unstable extraverts (choleric qualities such as touchy, restless, excitable, changeable, impulsive, irresponsible)
3. Stable introverts (phlegmatic qualities such as calm, even-tempered, reliable, controlled, peaceful, thoughtful, careful, passive)
4. Unstable introverts (melancholic)
(Eysenck,1968)



III. DATA CLEANING/PREPARATION

Check for Null Values

The Cowles data frame has 1421 rows and 4 columns. The first step in the pre-processing phase was to see if there were any Null values in the dataset. A quick check using the *mapply* command showed that there were no null values in the data and it was a clean set, there was no need to amend it any further.

Converting Data Types in the dataset

The machine learning models we were going to use all worked best when we had numeric data types when running the tests, with the exception of the dependant *volunteer* variable we were trying to predict. We checked the structure of the dataset and found that the only column that was an issue was the *sex* variable - This was a factor variable so we converted this to a numeric value in a binary format.

Normalisation

The next step was to normalise the numeric factors. This is an important step before running our predictive models because different variable can have different scales and this can affect the performance of the model by interpreting the importance of one variable over another (eg *neuroticism* has a scale of 0-24, sex 0-1). While understanding that the range for both *neuroticism* and *Extraversion* had the same range and *sex* was now a binary variable, we decided to err on the side of caution and continue with the normalization of all available columns. '[Models] are known to be sensitive to different scales of variables used in a prediction problem. In our case we will normalize the data with the goal of making all variables have a mean value of zero and a standard deviation of one.' (Torgo, 2017).

The *sex* column was already on a scale of 0-1, therefore normalization did not apply here.

Sequence of dataset

While looking through the dataset, we discovered that the data had been grouped by the volunteer variable we were trying to predict. If the running order had been organised sequentially like this then it may have a bearing on our models, because the training and test data might be skewed or biased in a certain way. This might have impacted on the ability to predict the outcome of the test data.

Therefore we randomized the order of the records in the dataset, and set a seed so that the order of the data points would remain the same when other users applied our models, therefore they would be able to produce the same results when running the test at a later time.

Splitting the data into training and test datasets

Finally we needed to split the data into two separate datasets, one to train the machine learning models we had chosen to apply, and another to run the test and see how successful the model was in predicting the outcomes using the other factors. There are various ways in how to best split up the dataset, and whether to include a separate validation set aswell, but we decided that a 70/30 split between the training and test data was the best option for applying the models. This left us with 994 records in the training data and 467 in the test data.

IV. CLASSIFICATION ANALYSIS

A. *Decision Tree*

Decision tree - nonparametric model

A decision tree is a statistical method that makes no assumption about the population distribution of sample size. This means that decision trees can be more flexible, robust and applicable to non-quantitative data in their approach.

There are a number of algorithms underlying the decision tree model. The ones that are most relevant to this report are;

C5.0

C5.0 is the latest iteration of a model originally developed by Ross Quinlan in 1986, the latest version under a proprietary license. It uses less memory and builds smaller rulesets than C4.5 while being more accurate.

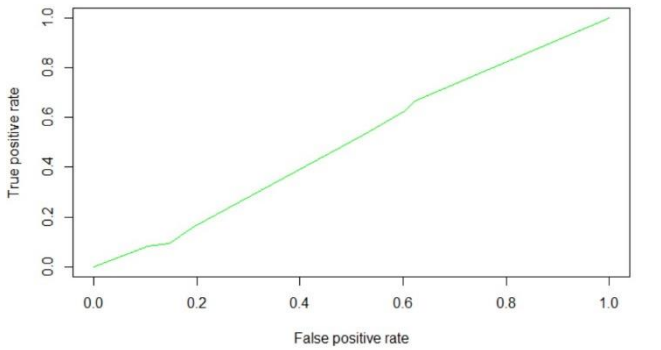
CART

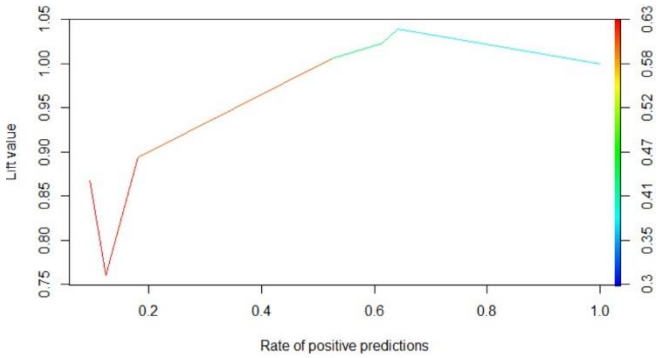
(Classification and Regression Trees) is very similar to C4.5, but it differs in that it supports numerical target variables (regression) and does not compute rule sets. CART constructs binary trees using the feature and threshold that yield the largest information gain at each node. For the purposes of our testing, the CART model is very suitable in its classification of binary rules. (Scikit-learn.org, 2019)

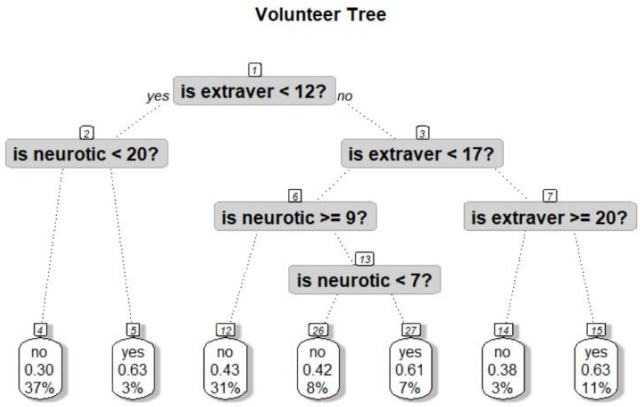
```
Classification tree:
rpart(formula = volunteer ~ ., data = train, method = "class",
    control = rpart.control(minsplit = 20, minbucket = 7, maxdepth = 10,
        usesurrogate = 2, xval = 10))

Variables actually used in tree construction:
[1] extraversion

Root node error: 423/994 = 0.42555

n= 994

        CP nsplit rel error xerror     xstd
1 0.020095      0   1.00000 1.0000 0.036851
2 0.010000      2   0.95981 1.0142 0.036916
```

```
> print(conf.matrix)

            Pred:no Pred:yes
Actual:no       500        77
Actual:yes      289       128
```

```
> perf_val <- performance(pred_val,"auc")#area under curve
> perf_val
An object of class "performance"
Slot "x.name":
[1] "None"

Slot "y.name":
[1] "Area under the ROC curve"

Slot "alpha.name":
[1] "none"

Slot "x.values":
list()

Slot "y.values":
[[1]]
[1] 0.5007985
```







B. KNN

NN classifiers are defined by their characteristic of classifying unlabelled instances by assigning them the class of the most similar labelled instances. 'The KNN simply stores the dataset…given a new test case, its prediction is obtained by searching for similar cases in the training data that was stored.' (Torgo, 2017)

The KNN algorithm is one of the simplest classification algorithms. Even with such simplicity, it can give highly competitive results. The KNN algorithm can also be used for regression problems. The only difference from the methodology detailed below will be using averages of nearest neighbors rather than voting from nearest neighbors (Analytics Vidhya, 2019). It is popular because it is easy to interpret, and calculation times are generally quite low. It is simple yet effective, with a fast training phase.

The major drawbacks with respect to kNN are its low efficiency - being a lazy learning method prohibits it in many applications such as dynamic web mining for a large repository. Its value is also dependent on the selection of a "good value" for k. (Wilson, 2000)

Method

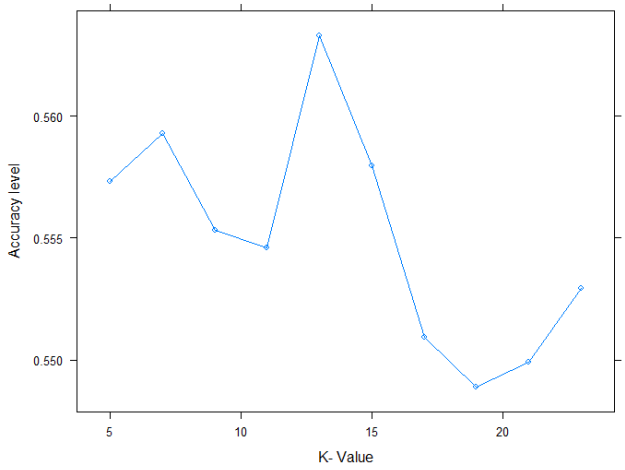
After the pre-processing outlined above we had reached a stage where the training and test data had been separated in a 70/30 split.

Before setting the KNN predictive model on the training dataset, we need to first arrange the *trainControl* method. This will set the parameters of the train method and control how it operates (Datasprint, 2019).

Here we are using repeated cross validation method using *trainContro*l. Number denotes either the number of folds and 'repeats' is for repeated 'r' fold cross validation. In this case, 3 separate 10-fold validations are used. You could use a variations on these settings, but 'It has become common practice to set k to 10 – it tends to produce optimised results without significant compute effort' (Torgo, 2017) , and in my experience, varying these parameters did not have a massive net-gain on the overall performance of the model.

We are using the Caret package in R . Caret is useful for being able to prejudge which K value will be the optimum value for running in the model without having to run several tests to discover the K.

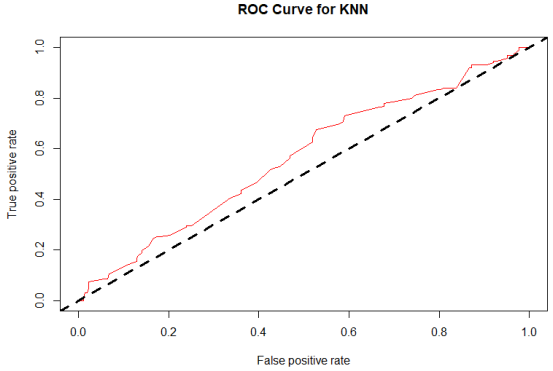
```
Resampling results across tuning parameters:

  k   Accuracy   Kappa
   5  0.5573131  0.08307232
   7  0.5592862  0.08173873
   9  0.5553131  0.06971140
  11  0.5546128  0.06690937
  13  0.5633064  0.08222053
  15  0.5579731  0.06724056
  17  0.5509192  0.05007178
  19  0.5488855  0.04433952
  21  0.5499192  0.04412750
  23  0.5529461  0.05005417

Accuracy was used to select the optimal model using the
largest value.
The final value used for the model was k = 13.
```



Here we can see that the optimum *K* value (ie number of clusters) for this model is 13.
'we can say that a larger value of *k* should be avoided because there is a risk of using cases that are already far away from the test case. (Torgo, 2017)

The next stage was to apply this machine learning to the test data and see how accurate the prediction outcomes was. The test data was run through the predictive KNN model based on probability of the binary Yes or No answer. The table provided answers on a scale of 0-1.

This was depicted on a ROC curve in order to assess how accurate the model had been on the test data.



In this diagram, we would expect the curve to be as close to top left as possible in order to show high accuracy – this is was clearly not the case in this instance. Judging from the diagram we can see that the level of accuracy was barely above 50%, but we can use the Area Under Curve (AUC) calculation to get a better indication.

```
> str(perf.auc)
Formal class 'performance' [package "ROCR"] with 6 slots
  ..@ x.name      : chr "None"
  ..@ y.name      : chr "Area under the ROC curve"
  ..@ alpha.name  : chr "none"
  ..@ x.values    : list()
  ..@ y.values    :List of 1
  .. ..$ : num 0.563
  ..@ alpha.values: list()
> unlist(perf.auc@y.values)
[1] 0.563476
>
```

The AUC calculation confirmed an accuracy level of 56% for this model.

The probability table that was produced above is not ideal for a binary answer, so I tried an alternative which provided a straight yes or no answer.
This option also gave us the opportunity to use a confusion matrix which may produce more detailed information on the results.

```
          Reference
Prediction  no yes
       no  184 106
       yes  80  57

               Accuracy : 0.5644
                 95% CI : (0.5159, 0.612)
    No Information Rate : 0.6183
    P-Value [Acc > NIR] : 0.98998

                  Kappa : 0.0481

 Mcnemar's Test P-Value : 0.06679

            Sensitivity : 0.6970
            Specificity : 0.3497
         Pos Pred Value : 0.6345
         Neg Pred Value : 0.4161
             Prevalence : 0.6183
         Detection Rate : 0.4309
   Detection Prevalence : 0.6792
      Balanced Accuracy : 0.5233

       'Positive' Class : no
```

The final accuracy is the KNN model was 56%, which was disappointing result. There were 186 incorrectly labelled decisions in total. The kappa rating (ie 'correction prediction is chance') rating of 4% is also not a 'close agreement'. I tried changing a lot of the parameters listed previously but there was no real increase the overall performance of the model. I also unable to use k-fold Cross-Validation to optimise performance as this had been utilised earlier as part of the *trainControl* function.

C. *Logistic Regression*

Introduction

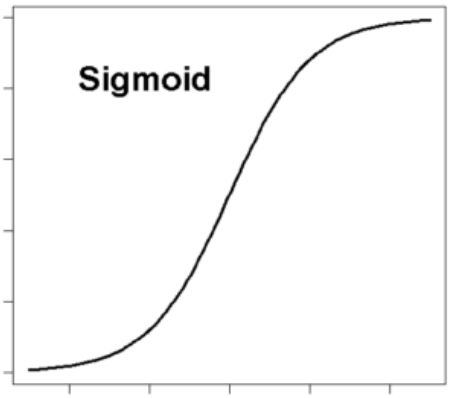
Logistic regression is another selected data mining technique to predict whether or not a person would volunteer based on the impact of the independent variables. Logistic regression is an ideal data mining technique for predicting categorical values such as a yes or no outcome as it produces probabilistic results. Logistic regression can be classed as a classification algorithm or a binary classifying machine learning algorithm.

Formula for Logistic Equation:

$$\frac{p}{1-p} = \exp(b_0 + b_1 x)$$

How Logistic Regression Works

Logistic regression measures the relationship that exists between the independent variables and the dependent variable by using logistic function to estimate the probabilities of a binary result. The conversion of these probabilities into binary values are the task of the underlying logistic function. This particular function is called the Sigmoid-Function. The Sigmoid-Function is an S shaped curve which converts any numeric value into a binary output.



Advantages and Disadvantages

Logistic regression is a widely used technique in the field of machine learning. Running logistic regression is very efficient as it does not require much computational resources to run. Secondly, it is easy to interpret and to conceptualise. However, like linear regression, dependent variables that are either poorly or very highly correlated with the independent variable should be removed from the testing. Another disadvantage to logistic regression is that one cannot solve non-linear problems with it. Another disadvantage to logistic equations is that it cannot be used for predicting continuous values such as temperature as it produces only a binary result.

Testing

Anova test can be applied to understand the impact size of the independent variables on the dependent variables. From running the test, we can see that extraversion has a statistically significant impact on whether or not a person would volunteer. The small p-value suggests that the percentage of predicting these results by chance again would be very slim.

	Df	Sum Sq	Mean Sq	F value	Pr(>F)	
train$sex	1	0.55	0.551	2.326	0.128	
train$extraversion	1	5.13	5.133	21.679	3.66e-06	***
train$neuroticism	1	0.06	0.064	0.268	0.605	
Residuals	990	234.42	0.237			

Logistic Regression Testing

Below is produced given logistic regression equation. The mining technique was applied to the training model which constituted 70% of the total datasets. From applying the regression technique, we can see that extraversion has a very low P-Value which means it is statistically significant and has a large impact on the dependent variable.

Anova Testing

	Estimate	Std. Error	z value	Pr(>|z|)	
(Intercept)	-1.519252	0.291213	-5.217	1.82e-07	***
neuroticism	0.007003	0.013729	0.510	0.610	
extraversion	0.077434	0.016851	4.595	4.32e-06	***
sex	0.175448	0.133538	1.314	0.189	

Anova Chi Squared Testing

	Df	Deviance	Resid. Df	Resid. Dev	Pr(>Chi)	
NULL			993	1344.5		
neuroticism	1	0.0073	992	1344.5	0.9321	
extraversion	1	22.3852	991	1322.1	2.231e-06	***
sex	1	1.7289	990	1320.3	0.1885	

Signif. codes: 0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1

The differences between the deviance number and the residual deviance prevails how the model is performing against the null state of the model. The null state assumes only the intercept. A large P value here suggests that the model could be explained more or less the same in terms of variation without that specific variable. This means that neuroticism has low impact on the dependent variable when compared to the impact of the other independent variables and its impact on the dependant. In other words, there is a 93% probability that neuroticism will have similar impacts on whether or not people would volunteer or not.

Equation

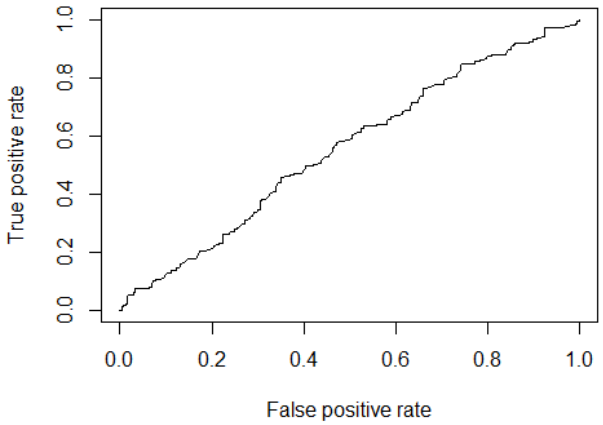
Here below is the logistic regression equation. These are the values assuming the dependent variable is equal to zero.

Coefficients:

(Intercept)	neuroticism	extraversion	sex
-1.519252	0.007003	0.077434	0.175448

Accuracy Testing

0.56 is the accuracy score of the prediction model. This score represents how many predicted results were correctly predicted divided by the total number of predictions. The area under the curve result is 0.5550737.



ROC Curve for Logistic Regression

The 0.84 accuracy on the test set is quite a good result. However, keep in mind that this result is somewhat dependent on the manual split of the data that I made earlier, therefore if you wish for a more precise score, you would be better off running some kind of cross validation such as k-fold cross validation.

CONCLUSION

We conclude that model KNN is the best performing model in terms of an accuracy rate with a score of 58%. The second best scoring model is the logistic regression model. In terms of the area under the curve score, we have concluded that both the logistic regression model and the KNN model produced the same score of 56%. The area under the curve score for the decision tree model is 54%. We can conclude that the KNN model was marginally the best performing machine learning algorithm to predict whether or not someone would volunteer based on specific independent variables, although the overall performance of the models was quite similar. Given the overall scores of the three models, it would be difficult to recommend them for the purposes of concluding who would volunteer and who would not from the factors that were included in the dataset.

REFERENCES

Analytics Vidhya. Available at: https://www.analyticsvidhya.com/blog/2018/03/introduction-k-neighbours-algorithm-clustering/ [Accessed 2 Aug. 2019].

Bodling A.M., Martin T. (2011) Eysenck Personality Inventory. In: Kreutzer J.S., DeLuca J., Caplan B. (eds) Encyclopedia of Clinical Neuropsychology. Springer, New York, NY

Guo, G., Wang, H., Bell, D., Bi, Y. and Greer, K. (2019). KNN Model-Based Approach in Classification.

Burger, E. (2019). 25 Volunteer Statistics That Will Blow Your Mind – VolunteerHub. [online] Volunteerhub.com. Available at: https://www.volunteerhub.com/blog/25-volunteer-statistics/ [Accessed 2 Aug. 2019].

Dataaspirant. (2019). *KNN R, K-Nearest Neighbor implementation in R using caret package*. [online] Available at: https://dataaspirant.com/2017/01/09/knn-implementation-r-using-caret-package/ [Accessed 2 Aug. 2019].

machinelearning-blog.com. (2019). The Logistic Regression Algorithm. [online] Available at: https://machinelearning-blog.com/2018/04/23/logistic-regression-101/ [Accessed 2 Aug. 2019].

Shizukalab.com. (2019). Plotting logistic regression in R - Shizuka Lab. [online] Available at: http://www.shizukalab.com/toolkits/plotting-logistic-regression-in-r [Accessed 2 Aug. 2019].

Scikit-learn.org. (2019). 1.10. Decision Trees — scikit-learn 0.21.3 documentation. [online] Available at: https://scikit-learn.org/stable/modules/tree.html [Accessed 2 Aug. 2019].

Torgo, L. (2017). Data Mining with R. CRC Press LLC.

Wilson, J. (2000). Volunteering. Annual Review of Sociology, 26(1), pp.215-240.