CLASSIFICATION (kNN)

Susana Reche Rodríguez
Student Number: 17165628
Date: 24.07.2020

Data Mining
Higher Diploma in Data Analytics

Table of Contents

Table of Figures

Part 2: 40 Marks

Using a dataset of your choice create a classification or regression model using RStudio. Clearly document the code. The chosen dataset must have a minimum of 10000 records. Provide a report of your finding and an evaluation of the performance of your model. The final report should not include the code and be clearly sectioned, correctly referenced and not exceed 1000 words. Code must be uploaded using the dedicated link.

PART 2: Supervised Machine Learning - Classification – kNN

1. Executive Summary

When a user navigates through the website with the intention to buy have different patterns than a user that is merely navigating with informational intent. The goal of the project is to be able to classify user sessions based on 17 variables in order to know if the user will perform a purchase or not. This information is highly valuable as can help to predict users that are close to a purchase.

The data is first analysed and prepared for exploration. Then a second data manipulation (normalizing and creating dummy variables for categorical data) is needed to be able to apply the kNN algorithm. Finally, the model accuracy is tested to understand if it can be used to classify users based on different variables related to the user session.

2. Data Exploration and Preparation

The data set "Online Shoppers Purchasing Intention Dataset" found at (Kastro, n.d.) contains 12,330 records, each of them being a user session. The data represents one year period and each session is for different users. There are 10 numerical variables and 8 categorical, for more detailed information see [pag.13, Appendix I]. The data set has no empty values [Fig.1].

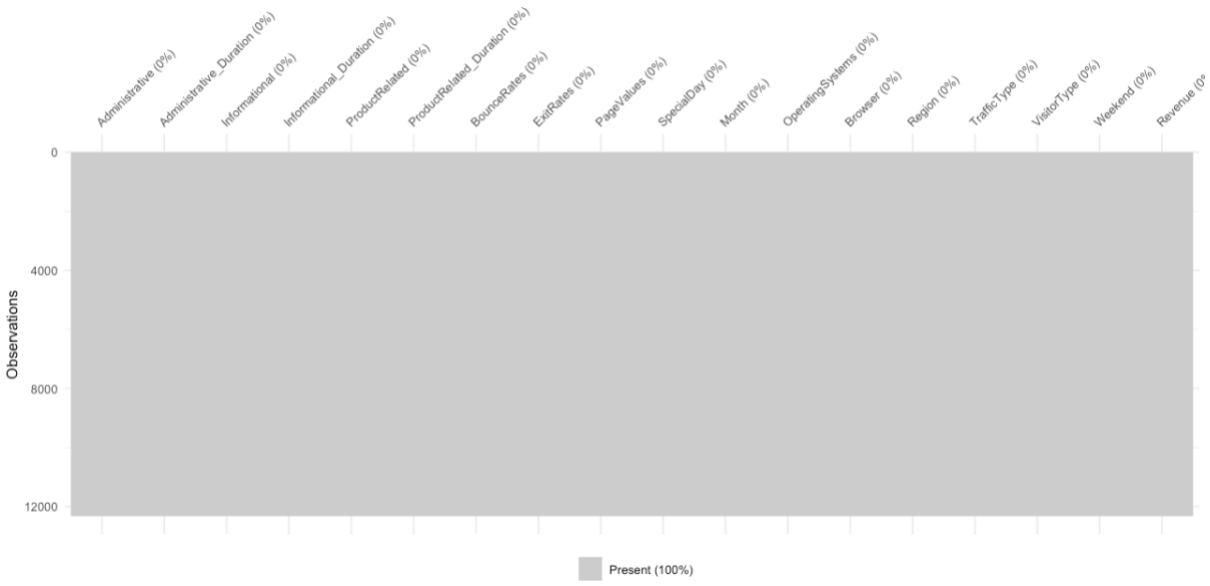


Figure 1: Complete dataset

The categorical variables, which are set as integers, characters and Boolean [Fig.2], are converted into factors [Fig. 3] and then are plotted to see the distribution [Fig. 4]. The data is split into sessions with purchase and sessions without, then the variables are plotted again for each segment [Fig. 5][Fig. 6]. The distribution of the data is similar in both segments with and without purchase.

There are a total number of 10,422 records without purchase and 1,908 records with purchase.

```
'data.frame':   12330 obs. of  18 variables:
 $ Administrative         : int  0 0 0 0 0 0 0 1 0 0 ...
 $ Administrative_Duration: num  0 0 0 0 0 0 0 0 0 0 ...
 $ Informational          : int  0 0 0 0 0 0 0 0 0 0 ...
 $ Informational_Duration : num  0 0 0 0 0 0 0 0 0 0 ...
 $ ProductRelated         : int  1 2 1 2 10 19 1 0 2 3 ...
 $ ProductRelated_Duration: num  0 64 0 2.67 627.5 ...
 $ BounceRates            : num  0.2 0 0.2 0.05 0.02 ...
 $ ExitRates              : num  0.2 0.1 0.2 0.14 0.05 ...
 $ PageValues             : num  0 0 0 0 0 0 0 0 0 0 ...
 $ SpecialDay             : num  0 0 0 0 0 0 0.4 0 0.8 0.4 ...
 $ Month                  : chr  "Feb" "Feb" "Feb" "Feb" ...
 $ OperatingSystems       : int  1 2 4 3 3 2 2 1 2 2 ...
 $ Browser                : int  1 2 1 2 3 2 4 2 2 4 ...
 $ Region                 : int  1 1 9 2 1 1 3 1 2 1 ...
 $ TrafficType            : int  1 2 3 4 4 3 3 5 3 2 ...
 $ VisitorType            : chr  "Returning_Visitor" "Returning_Visitor" "Returning_Visitor" "Returning_Visitor" ...
 $ Weekend                : logi  FALSE FALSE FALSE FALSE TRUE FALSE ...
 $ Revenue                : logi  FALSE FALSE FALSE FALSE FALSE FALSE ...
```

Figure 2: Structure of dataset just imported into R

```
'data.frame':   12330 obs. of  18 variables:
 $ Administrative         : int  0 0 0 0 0 0 0 1 0 0 ...
 $ Administrative_Duration: num  0 0 0 0 0 0 0 0 0 0 ...
 $ Informational          : int  0 0 0 0 0 0 0 0 0 0 ...
 $ Informational_Duration : num  0 0 0 0 0 0 0 0 0 0 ...
 $ ProductRelated         : int  1 2 1 2 10 19 1 0 2 3 ...
 $ ProductRelated_Duration: num  0 64 0 2.67 627.5 ...
 $ BounceRates            : num  0.2 0 0.2 0.05 0.02 ...
 $ ExitRates              : num  0.2 0.1 0.2 0.14 0.05 ...
 $ PageValues             : num  0 0 0 0 0 0 0 0 0 0 ...
 $ SpecialDay             : num  0 0 0 0 0 0 0.4 0 0.8 0.4 ...
 $ Month                  : Factor w/ 12 levels "Jan","Feb","Mar",..: 2 2 2 2 2 2 2 2 2 2 ...
 $ OperatingSystems       : Factor w/ 8 levels "1","2","3","4",..: 1 2 4 3 3 2 2 1 2 2 ...
 $ Browser                : Factor w/ 13 levels "1","2","3","4",..: 1 2 1 2 3 2 4 2 2 4 ...
 $ Region                 : Factor w/ 9 levels "1","2","3","4",..: 1 1 9 2 1 1 3 1 2 1 ...
 $ TrafficType            : Factor w/ 20 levels "1","2","3","4",..: 1 2 3 4 4 3 3 5 3 2 ...
 $ VisitorType            : Factor w/ 3 levels "New_Visitor",..: 3 3 3 3 3 3 3 3 3 3 ...
 $ Weekend                : Factor w/ 2 levels "0","1": 1 1 1 1 2 1 1 2 1 1 ...
 $ Revenue                : Factor w/ 2 levels "0","1": 1 1 1 1 1 1 1 1 1 1 ...
```

Figure 3: Structure of dataset after transformation of categorical variables into factors

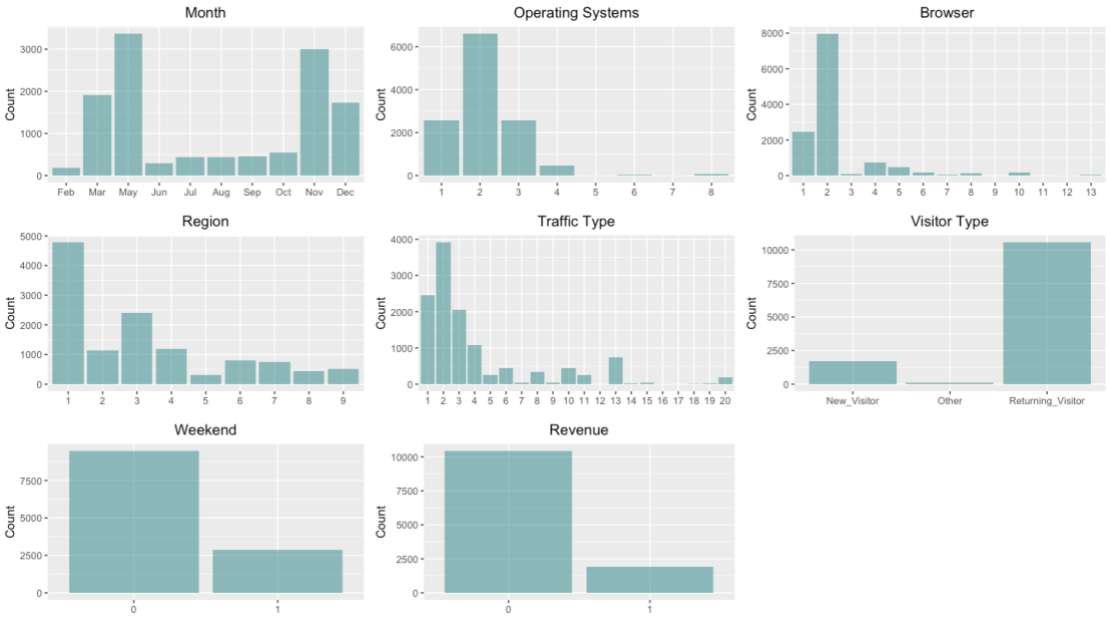


Figure 4: Categorical variables transformed into factors (Bar Plot)

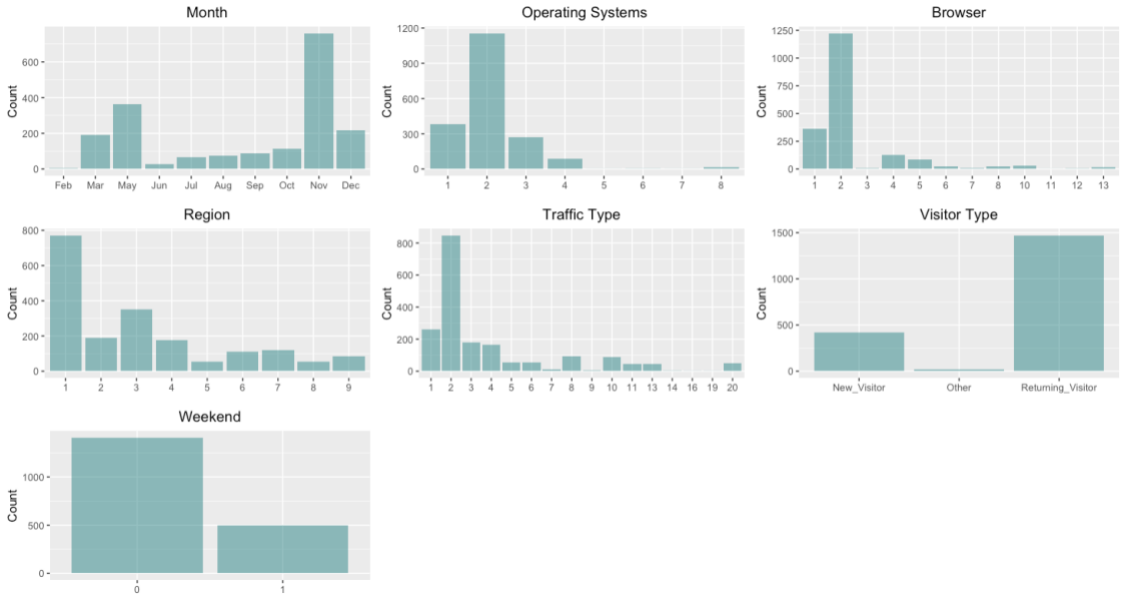


Figure 5: Categorical variables transformed into factors (with Purchase)(Bar Plot)

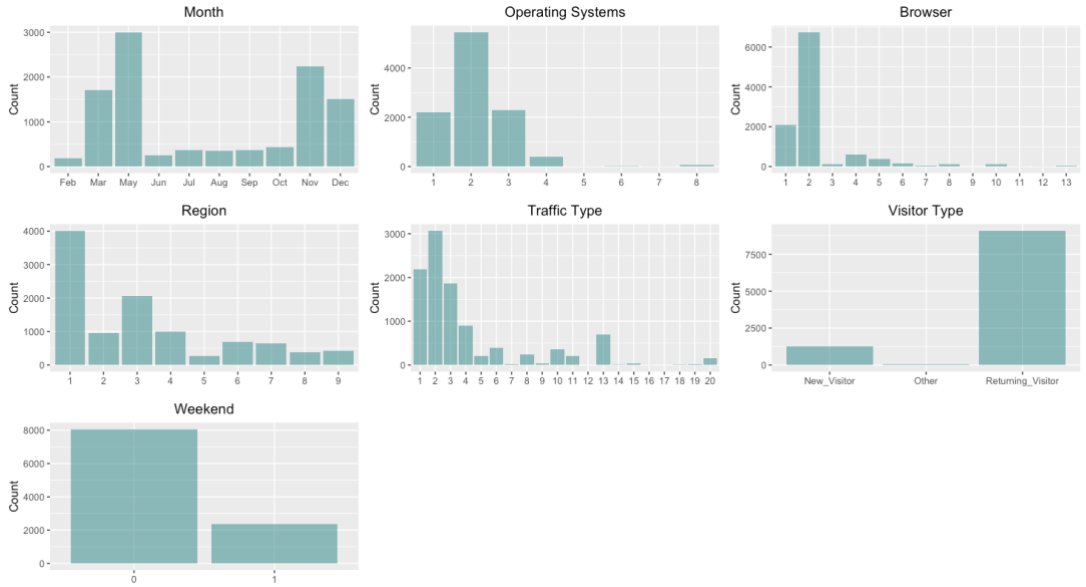


Figure 6: Categorical variables transformed into factors (without Purchase)(Bar Plot)

The distribution of the numerical variables is also analysed using box plots [Fig. 7]. Then the data is split into sessions with and without purchase and plotted again [Fig.8][Fig. 9]. Again doesn't seem to be any major difference between both segments.

Within the categorical variables, some categories have really few sessions and the same happens with the numerical variables, contains many outliers. Even if both segments (with and without purchase) contain outliers, in this case, the consideration is to include those outliers as might be relevant for the categorization.

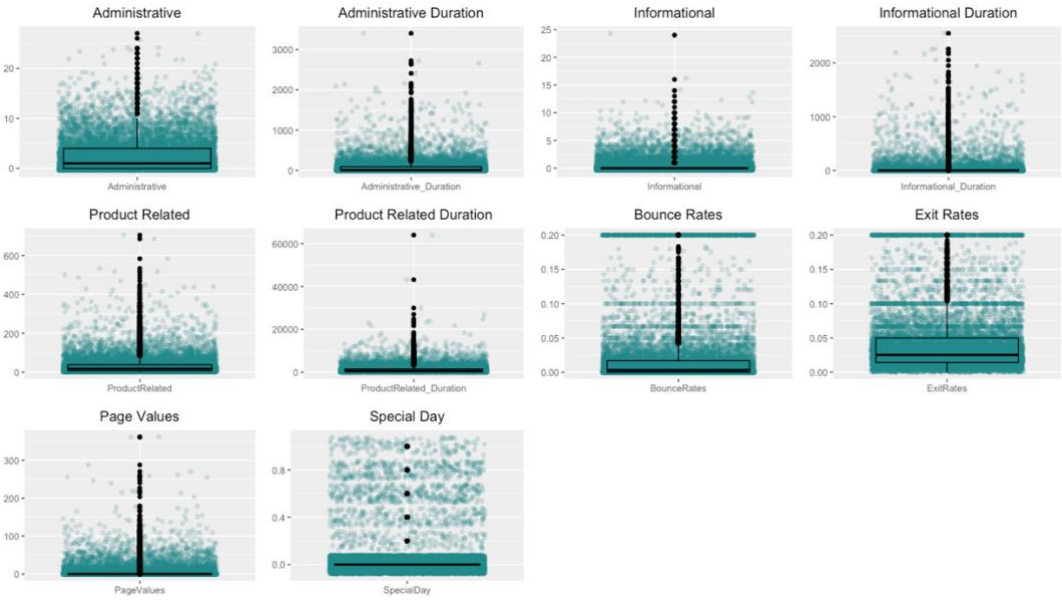


Figure 7: Numerical variables (Box Plot)

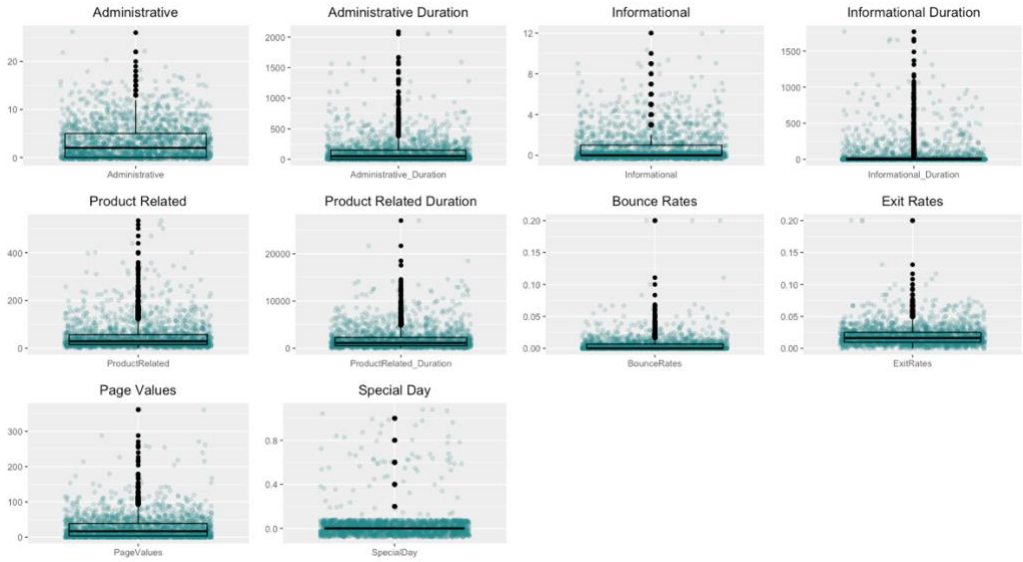


Figure 8: Numerical variables (with Purchase) (Box Plot)

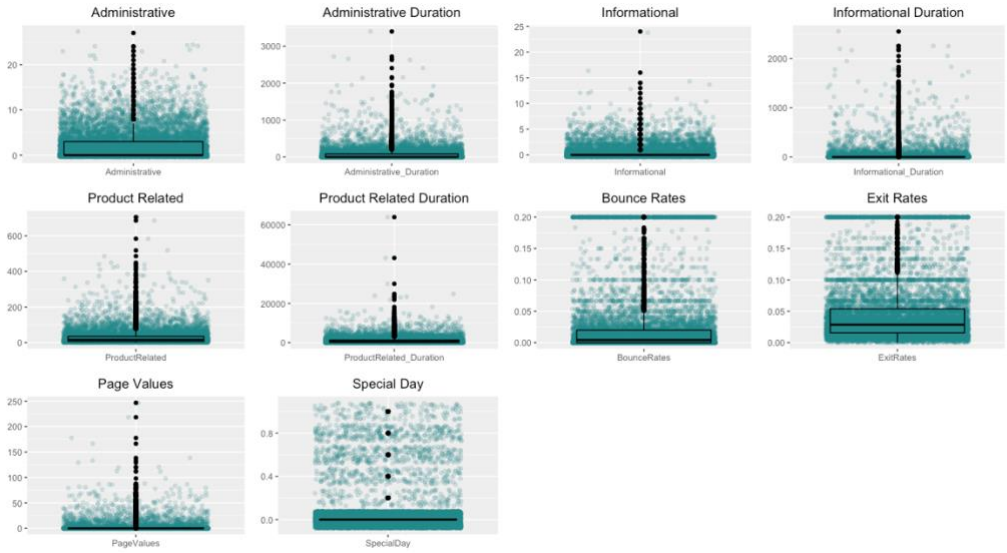
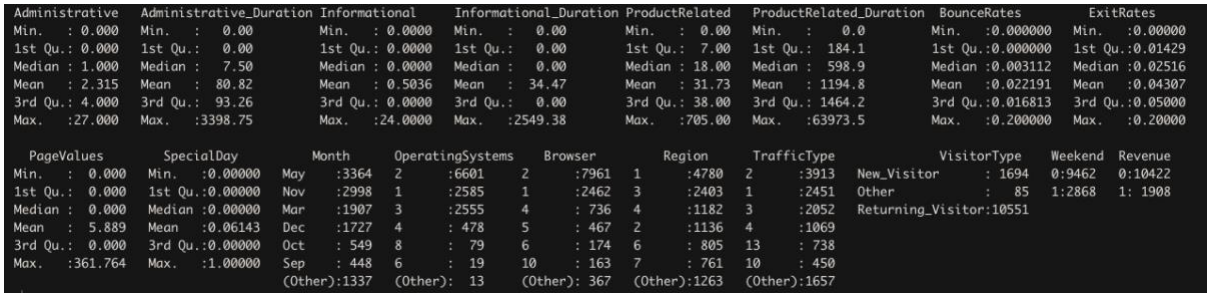


Figure 9: Numerical variables (without Purchase)(Box Plot)

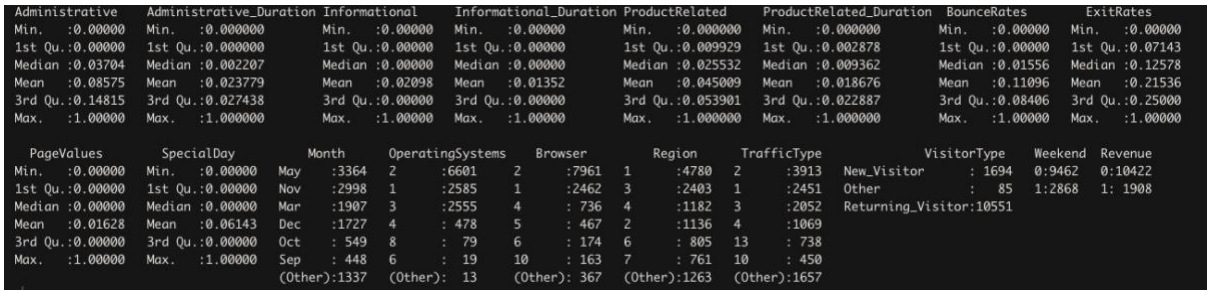


```
 Administrative   Administrative_Duration Informational     Informational_Duration ProductRelated   ProductRelated_Duration  BounceRates         ExitRates
 Min.   : 0.000   Min.   :   0.00         Min.   : 0.0000   Min.   :   0.00        Min.   :  0.00   Min.   :    0.0         Min.   :0.000000   Min.   :0.00000
 1st Qu.: 0.000   1st Qu.:   0.00         1st Qu.: 0.0000   1st Qu.:   0.00        1st Qu.:  7.00   1st Qu.:  184.1         1st Qu.:0.000000   1st Qu.:0.01429
 Median : 1.000   Median :   7.50         Median : 0.0000   Median :   0.00        Median : 18.00   Median :  598.9         Median :0.003112   Median :0.02516
 Mean   : 2.315   Mean   :  80.82         Mean   : 0.5036   Mean   :  34.47        Mean   : 31.73   Mean   : 1194.8         Mean   :0.022191   Mean   :0.04307
 3rd Qu.: 4.000   3rd Qu.:  93.26         3rd Qu.: 0.0000   3rd Qu.:   0.00        3rd Qu.: 38.00   3rd Qu.: 1464.2         3rd Qu.:0.016813   3rd Qu.:0.05000
 Max.   :27.000   Max.   :3398.75         Max.   :24.0000   Max.   :2549.38        Max.   :705.00   Max.   :63973.5         Max.   :0.200000   Max.   :0.20000

   PageValues        SpecialDay          Month      OperatingSystems    Browser         Region      TrafficType             VisitorType    Weekend  Revenue
 Min.   :  0.000   Min.   :0.00000   May    :3364   2      :6601     2      :7961   1      :4780   2      :3913   New_Visitor      : 1694   0:9462   0:10422
 1st Qu.:  0.000   1st Qu.:0.00000   Nov    :2998   1      :2585     1      :2462   3      :2403   1      :2451   Other            :   85   1:2868   1: 1908
 Median :  0.000   Median :0.00000   Mar    :1907   3      :2555     4      : 736   4      :1182   3      :2052   Returning_Visitor:10551
 Mean   :  5.889   Mean   :0.06143   Dec    :1727   4      : 478     5      : 467   2      :1136   4      :1069
 3rd Qu.:  0.000   3rd Qu.:0.00000   Oct    : 549   8      :  79     6      : 174   6      : 805   13     : 738
 Max.   :361.764   Max.   :1.00000   Sep    : 448   6      :  19     10     : 163   7      : 761   10     : 450
                                     (Other):1337   (Other):  13     (Other): 367   (Other):1263   (Other):1657
```

Figure 10: Dataset Summary

As seen in [Fig. 10] the numerical variables have different scales, which can make the classification not accurate, as the distances between elements wouldn't be comparable. In order to overcome this obstacle, the numerical variables are normalized (Data Science Tutorials, 2017).



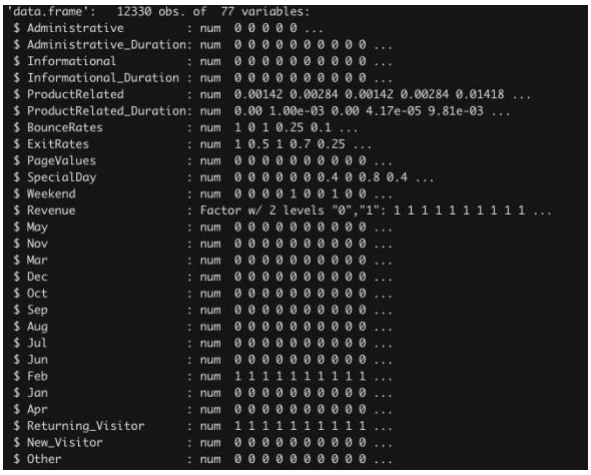
```
 Administrative   Administrative_Duration Informational     Informational_Duration ProductRelated     ProductRelated_Duration  BounceRates         ExitRates
 Min.   :0.00000  Min.   :0.000000        Min.   :0.00000   Min.   :0.00000        Min.   :0.000000   Min.   :0.000000         Min.   :0.00000   Min.   :0.00000
 1st Qu.:0.00000  1st Qu.:0.000000        1st Qu.:0.00000   1st Qu.:0.00000        1st Qu.:0.009929   1st Qu.:0.002878         1st Qu.:0.00000   1st Qu.:0.07143
 Median :0.03704  Median :0.002207        Median :0.00000   Median :0.00000        Median :0.025532   Median :0.009362         Median :0.01556   Median :0.12578
 Mean   :0.08575  Mean   :0.023779        Mean   :0.02098   Mean   :0.01352        Mean   :0.045009   Mean   :0.018676         Mean   :0.11096   Mean   :0.21536
 3rd Qu.:0.14815  3rd Qu.:0.027438        3rd Qu.:0.00000   3rd Qu.:0.00000        3rd Qu.:0.053901   3rd Qu.:0.022887         3rd Qu.:0.08406   3rd Qu.:0.25000
 Max.   :1.00000  Max.   :1.000000        Max.   :1.00000   Max.   :1.00000        Max.   :1.000000   Max.   :1.000000         Max.   :1.00000   Max.   :1.00000

   PageValues        SpecialDay          Month      OperatingSystems    Browser         Region      TrafficType             VisitorType    Weekend  Revenue
 Min.   :0.00000   Min.   :0.00000   May    :3364   2      :6601     2      :7961   1      :4780   2      :3913   New_Visitor      : 1694   0:9462   0:10422
 1st Qu.:0.00000   1st Qu.:0.00000   Nov    :2998   1      :2585     1      :2462   3      :2403   1      :2451   Other            :   85   1:2868   1: 1908
 Median :0.00000   Median :0.00000   Mar    :1907   3      :2555     4      : 736   4      :1182   3      :2052   Returning_Visitor:10551
 Mean   :0.01628   Mean   :0.06143   Dec    :1727   4      : 478     5      : 467   2      :1136   4      :1069
 3rd Qu.:0.00000   3rd Qu.:0.00000   Oct    : 549   8      :  79     6      : 174   6      : 805   13     : 738
 Max.   :1.00000   Max.   :1.00000   Sep    : 448   6      :  19     10     : 163   7      : 761   10     : 450
                                     (Other):1337   (Other):  13     (Other): 367   (Other):1263   (Other):1657
```

Figure 11: Dataset Summary after normalization of numerical variables

As kNN calculate distances between a pair of elements in order to classify them, all the variables need to be numerical.

Weekend is transformed from factor to numerical and the rest of categorical variables with more than 2 levels are transformed into dummy variables (Quant Dev, n.d.). For each level of the category new variables are created with 0 and 1 values. They are transformed outside the data frame and then combined again into it, ensuring the old variable is deleted. Now the data set contains 77 variables [Fig. 12][Fig.13][Fig.14].



```
'data.frame':   12330 obs. of  77 variables:
 $ Administrative         : num  0 0 0 0 0 ...
 $ Administrative_Duration: num  0 0 0 0 0 0 0 0 0 0 ...
 $ Informational          : num  0 0 0 0 0 0 0 0 0 0 ...
 $ Informational_Duration : num  0 0 0 0 0 0 0 0 0 0 ...
 $ ProductRelated         : num  0.00142 0.00284 0.00142 0.00284 0.01418 ...
 $ ProductRelated_Duration: num  0.00 1.00e-03 0.00 4.17e-05 9.81e-03 ...
 $ BounceRates            : num  1 0 1 0.25 0.1 ...
 $ ExitRates              : num  1 0.5 1 0.7 0.25 ...
 $ PageValues             : num  0 0 0 0 0 0 0 0 0 0 ...
 $ SpecialDay             : num  0 0 0 0 0 0 0.4 0 0.8 0.4 ...
 $ Weekend                : num  0 0 0 0 1 0 0 1 0 0 ...
 $ Revenue                : Factor w/ 2 levels "0","1": 1 1 1 1 1 1 1 1 1 1 ...
 $ May                    : num  0 0 0 0 0 0 0 0 0 0 ...
 $ Nov                    : num  0 0 0 0 0 0 0 0 0 0 ...
 $ Mar                    : num  0 0 0 0 0 0 0 0 0 0 ...
 $ Dec                    : num  0 0 0 0 0 0 0 0 0 0 ...
 $ Oct                    : num  0 0 0 0 0 0 0 0 0 0 ...
 $ Sep                    : num  0 0 0 0 0 0 0 0 0 0 ...
 $ Aug                    : num  0 0 0 0 0 0 0 0 0 0 ...
 $ Jul                    : num  0 0 0 0 0 0 0 0 0 0 ...
 $ Jun                    : num  0 0 0 0 0 0 0 0 0 0 ...
 $ Feb                    : num  1 1 1 1 1 1 1 1 1 1 ...
 $ Jan                    : num  0 0 0 0 0 0 0 0 0 0 ...
 $ Apr                    : num  0 0 0 0 0 0 0 0 0 0 ...
 $ Returning_Visitor      : num  1 1 1 1 1 1 1 1 1 1 ...
 $ New_Visitor            : num  0 0 0 0 0 0 0 0 0 0 ...
 $ Other                  : num  0 0 0 0 0 0 0 0 0 0 ...
```

Figure 12: Dataset structure with new dummy variables (1)

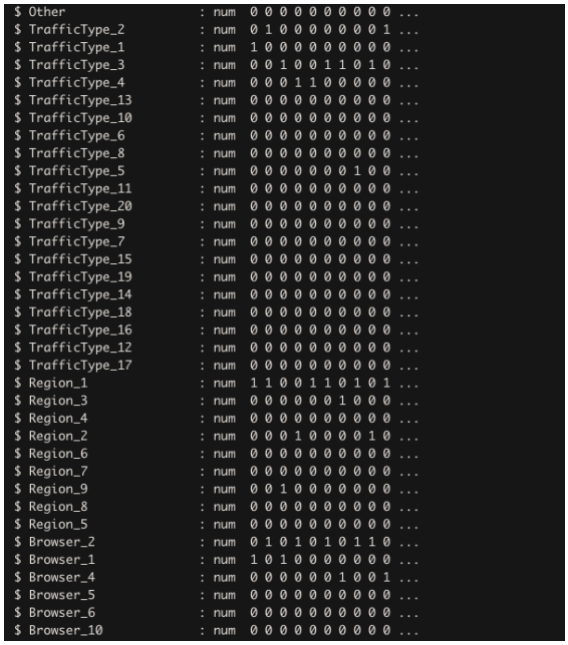


Figure 13: Dataset structure with new dummy variables (2)

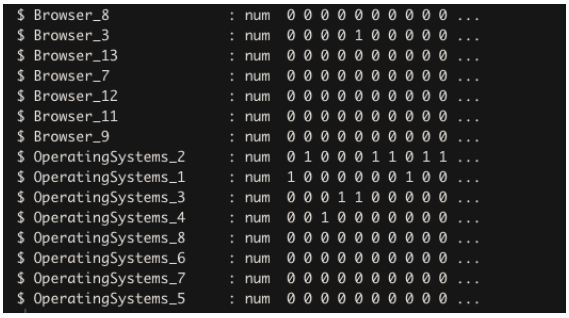


Figure 14: Dataset structure with new dummy variables (3)

The variable used as “class”, in this case, “Revenue” needs to be a factor.

As kNN is a supervised algorithm the dataset needs to be split into two segments, one for training and one for the validation of the model. The chosen threshold is that 80% of the records are used to train the model (8,338 records without purchase and 1,527 with purchase), and the rest 20% to validate it (2,084 records without purchase and 381 with purchase).

3. kNN and Model Evaluation

The chosen method to predict the classification is the non-parametric kNN (k nearest neighbours) algorithm. Receives this name as the classification of an element within the dataset is based in the class of the majority of elements which are nearby, or neighbours. The distance between the element and its neighbours is calculated using the Euclidian distance, as default, even if other distance methods can also be applied instead. The number of neighbours to include can be chosen (k).

Depending on the number of neighbours taken into consideration the result can change. If the K is too big small patterns can be missed and if it is too small irrelevant elements can be taken into consideration. kNN was tested for different values and the one with optimal result, in this case is 13 [Fig. 15].

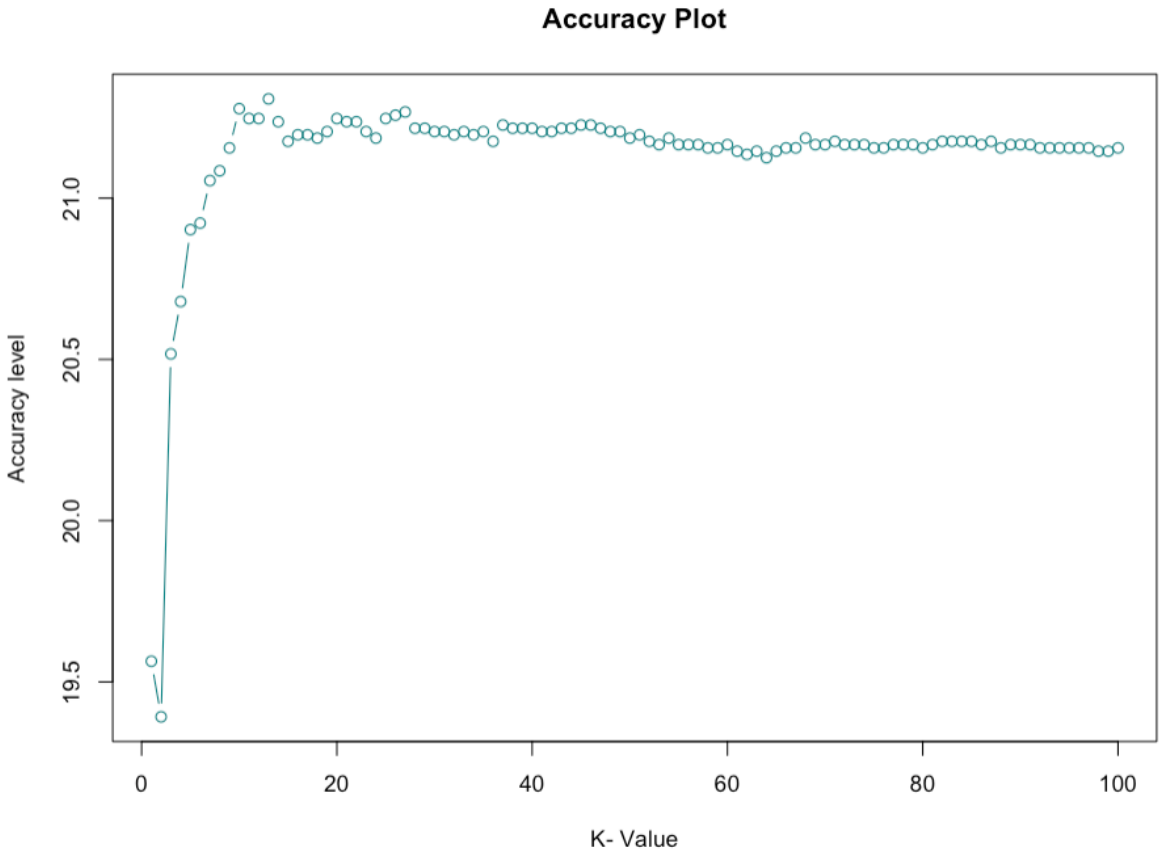


Figure 15: Accuracy Plot

The results of kNN with a K of 13 can be seen in the confusion matrix [Fig. 16].

The overall accuracy is 0.8523 (DataCamp, 2016), which are the correctly classified sessions, 2,066 true negatives and 35 true positives. However, there are 346 false negative (Type II error) and 18 false positive (Type I error). The model has a hard time recognising sessions with Purchase as only predicted correctly 35 out of the total 381 (low sensitivity of 0.09. The original data has a higher number of sessions without purchase, and this can be a reason why the classifier is having a hard time classifying sessions with purchase.

Even though the overall model looks like can predict with accuracy, it would not be a reliable model as the purpose is to predict true positives, which can be confirmed by looking at the 0.13 kappa.

Precision(Pos Pred Value) = TP/(TP+FP) =3 5/(35+18) = 0.66038
Recall (Sensitivity) =TP/(TP+FN) = 35/(35+346) = 0.09186

F-measure = (2* precision * recall)/(recall + precision) =
(2*0.66038*0.09186)/(0.09186+0.66038) = 0.16128

```
Confusion Matrix and Statistics

          Reference
Prediction FALSE TRUE
     FALSE  2066  346
     TRUE     18   35

               Accuracy : 0.8523
                 95% CI : (0.8377, 0.8661)
    No Information Rate : 0.8454
    P-Value [Acc > NIR] : 0.1792

                  Kappa : 0.1284

 Mcnemar's Test P-Value : <2e-16

            Sensitivity : 0.09186
            Specificity : 0.99136
         Pos Pred Value : 0.66038
         Neg Pred Value : 0.85655
             Prevalence : 0.15456
         Detection Rate : 0.01420
   Detection Prevalence : 0.02150
      Balanced Accuracy : 0.54161

       'Positive' Class : TRUE
```

Figure 16: Confusion Matrix kNN

4. Summary

kNN doesn't seem the right choice to be able to predict users which will perform a purchase.

The next steps are trying different classification algorithms as Decision Trees / Random Forest [pag. 13, Appendix II], Naïves Bayes, C4.5 (Karim Baati, 2020), Support Vector Machine or Logistic Regression among others and find out which can be the most efficient. The F-score can be used to compare those different type of models.

Appendix I – About the data

The information about the data set is described at the data set source (Kastro, n.d.).
The data set contains 10 numerical variables and 8 categorical:

Numerical Variables

- "Administrative", "Informational" and "Product Related" represent the number of pages the user visited of this type.

- "Administrative Duration", "Informational Duration" and "Product Related Duration" represent the time spent in each of those category pages.

- "Bounce Rate" represents the users that landed on a page coming from the SERPs and then leave the website without visiting another page.

- "Exit Rate" is a measure applied to a page, it calculates the number of times the page was the exit point for the website divided by the total number of times the page was viewed by a user, as per (Google, n.d.). In this case, probably an average for all the pages visited during the session was calculated.

- Same as "Exit Rate", "Page Value" is a measure applied to a page, it tries to estimate how much value a page has based on how contributes to revenue or other possible established goals for the website, as per (Google, n.d.). In this case, probably an average for all the pages visited during the session was calculated.

- "Special Day", is represented by numerical value which represents the closeness to a special day in the calendar.

Categorical Variables

- "Operating System", informs about the operating systems that the user was using during the session. There are 8 different operating systems within the data set.
- "Browser", informs about the browser that the user was using during the session. There are 13 different browsers within the data set.
- "Region" informs about the Region from the one the user is at the moment of the session. There are 8 different Regions within the data set.
- "Traffic Type", indicate the type of channel the user used to access the site and initiate the session. There are 20 different types as per the data set. Some example of traffic type will be for example organic, paid, social media, …
- "Visitor Type", informs if the user was a "Returning Visitor", a "New Visitor" or "Other".
- "Weekend", informs if the session was during the weekend.
- "Month", reveal the Month when the session was performed.
- "Revenue", indicate if the was a purchase or no during the session.

Appendix II – Decision Tree and Random Forest

Using the same transformed data, used for the kNN model a decision tree is created [Fig. 17]. Some of the key variables seem to be the type of pages visited by the user, especially is the page has a high bounce rate and the value of the page. November seems to be an important month as per the data. Even though in November is the "Black Friday" sales is not related to food, should be studied.

The performance of the decision tree compared to the kNN model is much higher.

F-measure = (2* precision * recall)/(recall + precision) = (2*0.76125*0.57743)/(0.57743+0.76125) = 0.65672

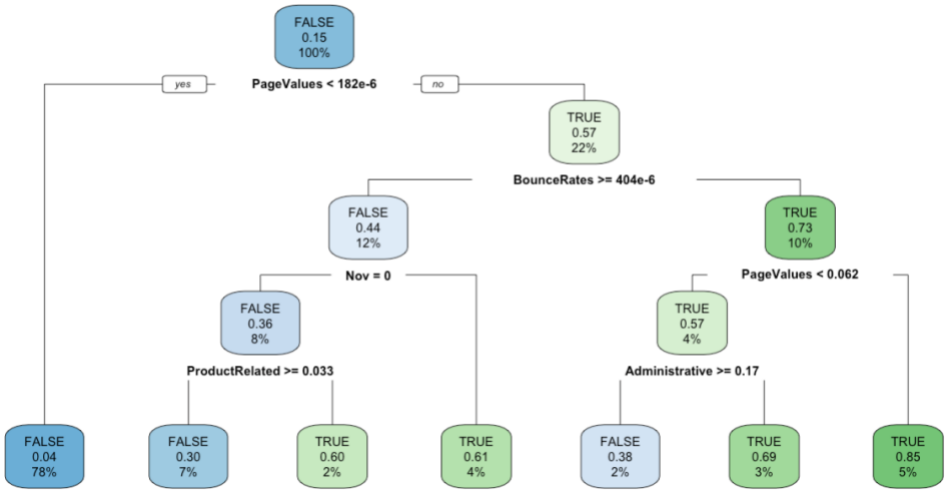


Figure 17: Decision Tree

```
Confusion Matrix and Statistics

          Reference
Prediction FALSE TRUE
     FALSE  2015  161
     TRUE     69  220

               Accuracy : 0.9067
                 95% CI : (0.8945, 0.9179)
    No Information Rate : 0.8454
    P-Value [Acc > NIR] : < 2.2e-16

                  Kappa : 0.6039

 Mcnemar's Test P-Value : 1.969e-09

            Sensitivity : 0.57743
            Specificity : 0.96689
         Pos Pred Value : 0.76125
         Neg Pred Value : 0.92601
             Prevalence : 0.15456
         Detection Rate : 0.08925
   Detection Prevalence : 0.11724
      Balanced Accuracy : 0.77216

       'Positive' Class : TRUE
```

Figure 18: Confusion Matrix Decision Tree

By applying then Random Forest (which uses bagging) to the same partition used for the decision tree, the accuracy of the model improves but not the overall F-measure.

In this case, the page value is the most relevant variable followed by the exit rate.

F-measure = (2* precision * recall)/(recall + precision) = (2*0.82329*0.53806)/(0.53806+0.82329) = 0.65079

```
Confusion Matrix and Statistics

          Reference
Prediction FALSE TRUE
     FALSE  2040  176
     TRUE     44  205

               Accuracy : 0.9108
                 95% CI : (0.8988, 0.9217)
    No Information Rate : 0.8454
    P-Value [Acc > NIR] : < 2.2e-16

                  Kappa : 0.6022

 Mcnemar's Test P-Value : < 2.2e-16

            Sensitivity : 0.53806
            Specificity : 0.97889
         Pos Pred Value : 0.82329
         Neg Pred Value : 0.92058
             Prevalence : 0.15456
         Detection Rate : 0.08316
   Detection Prevalence : 0.10101
      Balanced Accuracy : 0.75847

       'Positive' Class : TRUE
```

Figure 19: Confusion Matrix Random Forest

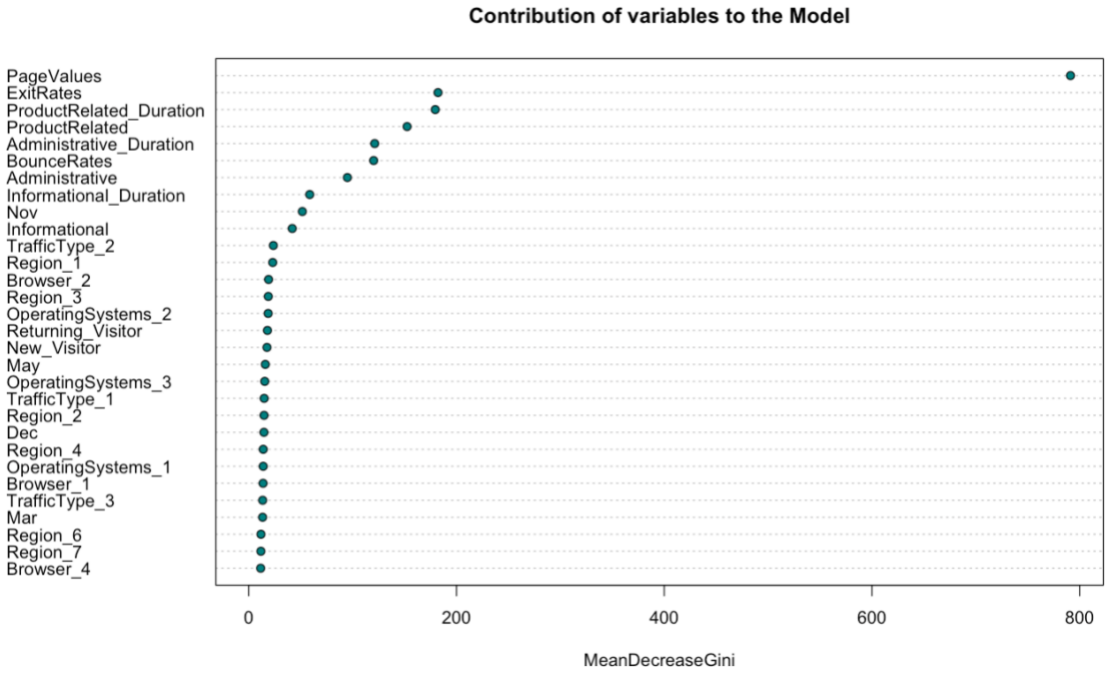


Figure 20: Importance of variables in the performance of the model

Bibliography

Data Science Tutorials, 2017. *K Nearest Neighbor (kNN) Algorithm | R Programming | Data Prediction Algorithm.* [Online]
Available at: https://www.youtube.com/watch?v=lDCWX6vCLFA
[Accessed 16 July 2020].

DataCamp, 2016. *R tutorial: Data splitting and confusion matrices.* [Online]
Available at: https://www.youtube.com/watch?v=Gx3_o1JVkPE
[Accessed 19 July 2020].

Google, n.d. *Exit Rate vs. Bounce Rate.* [Online]
Available at: https://support.google.com/analytics/answer/2525491?hl=en
[Accessed 18 July 2020].

Google, n.d. *How Page Value is calculated.* [Online]
Available at: https://support.google.com/analytics/answer/2695658?hl=en
[Accessed 18 July 2020].

Greenwell, B. B. &. B., 2020. *Bagging.* [Online]
Available at: https://bradleyboehmke.github.io/HOML/bagging.html
[Accessed 23 July 2020].

Harrison, O., 2018. *Machine Learning Basics with the K-Nearest Neighbors Algorithm.* [Online]
Available at: https://towardsdatascience.com/machine-learning-basics-with-the-k-nearest-neighbors-algorithm-6a6e71d01761
[Accessed 16 July 2020].

Jalalian, S., 2019. *Prediction of Online Shopper's Intention.* [Online]
Available at: https://fsocietysj.com/prediction-of-online-shoppers-intention/
[Accessed 18 July 2020].

Jorgesen, T., 2014. *Classification Trees in R.* [Online]
Available at: https://www.youtube.com/watch?v=3TbzO5vep20
[Accessed 24 July 2020].

Kammar, N., 2020. *Predicting Online Shopper's Intention.* [Online]
Available at: https://rpubs.com/naveen_kammar/shoppers_intention
[Accessed 18 July 2020].

Karim Baati, M. M., 2020. *Real-Time Prediction of Online Shoppers' Purchasing Intention Using Random Forest.* [Online]
Available at: https://link.springer.com/chapter/10.1007/978-3-030-49161-1_4
[Accessed 19 July 2020].

Kastro, C. O. S. &. Y., n.d. *Online Shoppers Purchasing Intention Dataset Data Set.* [Online]
Available at:
http://archive.ics.uci.edu/ml/datasets/Online+Shoppers+Purchasing+Intention+Dataset
[Accessed 16 July 2020].

L., F. L. A., 2019. *Clasificación de canciones en base al artista aplicando KNN sobre un FMA Dataset.* [Online]
Available at: https://medium.com/@freddy.abadl/clasificaci%C3%B3n-de-canciones-en-base-al-artista-aplicando-knn-sobre-un-fma-ce9ce44eb98d
[Accessed 16 July 2020].

Lantz, B., 2015. *Machine Learning with R.* 2nd Edition ed. Birmingham: Packt Publishing Ltd..

Lateef, Z., 2020. *KNN Algorithm: A Practical Implementation Of KNN Algorithm In R.* [Online]
Available at: https://www.edureka.co/blog/knn-algorithm-in-r/
[Accessed 16 July 2020].
Le, J., 2018. *Decision Trees in R.* [Online]
Available at: https://www.datacamp.com/community/tutorials/decision-trees-R
[Accessed 23 July 2020].
Quant Dev, n.d. *k-Nearest Neighbor: An Introductory Example.* [Online]
Available at: https://quantdev.ssri.psu.edu/sites/qdev/files/kNN_tutorial.html
[Accessed 16 July 2020].
Sharma, R., n.d. *Online Shopper's Intention - Data Mining, Clustering, Classification.* [Online]
Available at: https://www.kaggle.com/roshansharma/online-shoppers-intention
[Accessed 18 July 2020].
Sirohi, K., 2018. *K-nearest Neighbors Algorithm with Examples in R (Simply Explained knn).* [Online]
Available at: https://towardsdatascience.com/k-nearest-neighbors-algorithm-with-examples-in-r-simply-explained-knn-1f2c88da405c
[Accessed 15 July 2020].
Stackoverflow, 2015. *R - convert from categorical to numeric for KNN.* [Online]
Available at: https://stackoverflow.com/questions/30058362/r-convert-from-categorical-to-numeric-for-knn
[Accessed 16 July 2020].
Stakoverflow, 2013. *Add a prefix to column names.* [Online]
Available at: https://stackoverflow.com/questions/14872081/add-a-prefix-to-column-names
[Accessed 18 July 2020].
Suchaga, G. S.-K. M. &. P. A., 2015. *A k-Nearest Neighbors Method for Classifying User Sessions in E-Commerce Scenario.* [Online]
Available at: https://yadda.icm.edu.pl/baztech/element/bwmeta1.element.baztech-40e29335-8f5f-4d8c-aa93-8c13a90d1b2d
[Accessed 16 July 2020].
Tierney, N., 2020. *Gallery of Missing Data Visualisations.* [Online]
Available at: https://cran.r-project.org/web/packages/naniar/vignettes/naniar-visualisation.html
[Accessed 18 July 2020].