Classification analysis on receipt data of a big chain of clothing supermarkets

Evaluation of Classification Algorithms

Khan Rashed Ziyaulla Khan
Data Mining & Machine Learning
Applied Research in Computer Science
WS 2021-22, Hof University of Applied Science

ABSTRACT

All industries and organizations are striving to become profitable by attracting more customers. In order to achieve this goal, they must understand the priorities of the customer. By applying data mining & machine learning techniques, one can better understand consumer behaviour, determine purchase patterns, and improve customer service. This study aims to identify the likely factors that influence customer buying decisions and to classify the buying patterns by using single and ensemble classification algorithms. According to the experimental results, Random Forest algorithm performed the best in terms of accuracy and other evaluation parameters. As a result, the company can select the appropriate model to use in order to lower time and cost when providing a more personalized customer experience based on the results of the new model. In this paper, a predictive and strategy model is developed using historical data and machine learning algorithms which will do binary classification on given data.

CSS CONCEPTS

• Computing methodologies → Machine learning; Machine learning algorithms.

KEYWORDS

Supervised Learning, Classification, Naïve bayes, Random Forest, Decision Tree, Support Vector Machine.

1. Record

In this study, "Receipt data of a big chain of clothing supermarkets" dataset is used in which it includes information about clothing item purchase receipts. This dataset includes monthly records from different countries in different feather files.

Dataset consist of fourteen attributes for each record with both numerical and categorical features. Furthermore, this data is free from any tendency towards special day sale, or user profile and specific campaign.

(Table 1) shows the descriptions of the variables present in the dataset.

Attribute Name	Attribute Description
dwh_ka_bon_head_id	unique identifier of bon
dwh_store_no	unique identifier of store
doc_num_n	bon line number
dim_cash_movement_id	type of transaction
dwh_article_no	unique identifier of article
mpg_no	unique identifier of the main product group
pg_no	unique identifier of the product group
ag_no	unique identifier of article group
article_size	unique identifier of article size
dwh_tax_percentage_no	VAT percent
bon_amount	number of items of the article
bon_timestamp	timestamp
Country_short_name	name of the country
weekday	day of the week

Table 1: Description of attributes

2. Goals and assumptions

The objective of this project is to use machine learning to identify the customer buying patterns based on the categorical attributes such as sale weekday, sale time, region present in the dataset. Four different classification algorithms are used to demonstrate which aspects yield the best results. Furthermore, it also demonstrates how precisely a model can classify clothing article purchase records based on their characteristics.

Hypothesis

1. Seasonal differences in buying patterns of clothing articles are apparent (for example, in summer, a specific type of outfit is more in demand than in winter).

2. The algorithms should be able to predict when the article was purchased based on the season.

dwh_ka_bon_head_id	dwh_store_no	doc_num_n	dim_cash_movement_id	dwh_article_no	article_size	dwh_tax_percentage_no	bon_timestamp	mpg_no	pg_no	ag_no	country_short_name	bon_amount	weekday
453089135	2772	5	1500002000	5158726	0	1900	2019-06-15 16:44:00	89	39	1	DE	2	Sa
434790672	2578	2	1500002000	1682379	218	1900	2019-06-11 13:41:00	42	11	1	DE	1	Di
522119877	2286	2	1500002000	5155547	255	1900	2019-06-28 16:08:00	52	7	1	DE	1	Fr
525830685	5405	1	1500002000	1689941	502	1900	2019-06-29 17:45:00	31	2	11	DE	2	Sa
465447836	1292	1	1500002000	1708373	0	1900	2019-06-18 11:52:00	90	15	2	DE	1	Di
486949295	1807	1	1500002000	5156841	254	1900	2019-06-25 20:01:00	53	2	1	DE	1	Di
417192234	1559	6	1500002000	5163036	254	1900	2019-06-04 16:26:00	53	7	5	DE	1	Di
487055127	1305	2	1500002000	5157514	252	1900	2019-06-25 15:38:00	52	6	1	DE	1	Di
522135641	989	3	1500002000	5156482	254	1900	2019-06-28 12:24:00	52	2	1	DE	1	Fr
491032161	1571	1	1500002000	5155798	48	1900	2019-06-24 16:47:00	53	3	1	DE	1	Mo
418213747	2547	5	1500002000	5156835	253	1900	2019-06-12 11:16:00	53	2	3	DE	1	Mi
453215961	1686	6	1500002000	5161413	14	1900	2019-06-15 13:08:00	86	1	1	DE	1	Sa
518393071	2563	1	1500002000	5160392	252	1900	2019-06-27 11:27:00	52	3	2	DE	1	Do
522124334	1765	1	1500002000	5153898	253	1900	2019-06-28 11:29:00	53	7	4	DE	1	Fr
449482329	1330	3	1500002000	5159589	254	1900	2019-06-17 18:13:00	52	2	1	DE	1	Mo
434386691	558	3	1500002000	5156517	0	1900	2019-06-06 16:06:00	89	33	1	DE	4	Do
518528038	1553	3	1500002000	5162644	253	1900	2019-06-27 20:10:00	53	7	5	DE	1	Do

Figure 1: Snapshot of receipt data

3. Supervised Learning

The difference between supervised and unsupervised machine learning algorithms lies in the fact that supervised machine learning relies on labelled input data to build a function that produces an appropriate output when given a new unlabelled data.

The process of supervised learning is used to create artificial intelligence (AI), where a computer algorithm is trained using input data that has been labelled for a particular output [1]. A model is trained until it can detect the underlying patterns and relationships among the input data and the output labels enables it to produce accurate labelling results when presented with data it has never seen before [2].

The goal of supervised learning is to interpret data within the context of a specific question. Unlike all machine learning algorithms, supervised learning is based on training. In this phase, the system is fed a set of labelled data sets, which tell it what output is associated with each input value [3]. The goal is to determine how accurately the algorithm will perform on unlabelled data.

4. Classification

A classification algorithm sort data into a specific number of classes or categories based on labelled training data [4]. Classification algorithms can be used for binary classifications as well as multiclass classification such as filtering email into spam or non-spam and categorizing set of images of fruits into various class.

Speech recognition, face detection, handwriting recognition, document classification are some of the most common classification problems. Linear classifiers, support vector machines (SVM), decision trees, k-nearest neighbour, and random forest are some of the most common classification algorithms [5].

4.1. Naïve Bayes

Naive Bayes is a classification approach that is based on the Bayesian principle of class conditional independence. This means that one feature does not change the probability of another, and that each of the features has an equal impact on the probability of an outcome [6].

A Naive Bayes classifier is among the simplest probabilistic classifiers [7]. It often shows amazingly good results in many real-world applications, despite its strong assumptions that all features are provisionally independent.

In Naïve Bayes, if there is no association between a class label and an attribute value, then the frequency-based probability estimate will be zero. And this will also result in a zero when all the probabilities are multiplied [8]. If a particular attribute value is not associated with every class value, then adding one to the count is an approach to combat the 'zero-frequency problem' in a Bayesian environment.

4.2. SVM

The goal of support vector machines algorithm is to find a set of N-dimensional hyperplanes (N is the number of features) that can clearly classify the data points.

In order to separate the two classes of data points, there are many possible hyperplanes that can be chosen. The objective is to identify the plane that has the maximum margin, i.e., the distance between data points of both classes [9]. The maximum margin distance provides some reinforcement so that future data points can be classified with more accuracy.

Hyperplanes serve as decision boundaries that categorize data points. Data points that fall on either side of the hyperplane can be categorized. In addition, the dimensions of the hyperplane depend on the number of input features. If there are only two input features, then the hyperplane is just a line [10]. If there are three input features, then the hyperplane becomes a two-dimensional plane. The difficulty increases when the number of features exceeds three.

4.3. Decision Tree

Decision trees can be used to represent decision making processes visually and explicitly. A decision tree is an abstract model that relies on a tree-like structure. This is a commonly used tool for deriving strategies to reach a particular goal in data mining, but it is also widely used in machine learning [11]. Trees have roots, branches, and leaves. The decision tree follows the same structure, containing nodes at the root, branches, and leaves.

A test of an attribute appears on each internal node, the result of the test appears on the branch, and the class label resulting from the test appears on the leaf node. When a problem has a large set of features, it usually leads to a large number of splits, which results in a huge tree, which is complex and can lead to overfitting [12].

4.4. Radom Forest

Random forests are composed of many individual decision trees that operate as an ensemble. Each tree in a random forest generates an output and the output with the most votes become the prediction of the model. As a group, the trees prevent each other from making errors. While some trees may be wrong, a lot of other trees can be right. Through this, as a group, the trees are able to move in the right direction.

In random forest algorithms, there are three main parameters, which need to be set prior to training. The three parameters are node size, the number of trees, and the number of sampled features [13]. A random forest classifier can then be used to solve regression or classification problems [14]. To ensure that random forest can perform well, the following conditions should be met:

1. In order for models to do better than random guessing, we need to have some actual signal in our features.
2. There should be low correlations between the predictions made by the individual trees.

5. Data Pre-processing

Various investigations are conducted using only a subset of 2 months (June and November) of records in this study in order to keep the time frame within reasonable limits. A careful selection is be made to ensure that these hypotheses are tested.

In machine learning algorithms, diversity in scale could cause huge issues. To solve this problem, feature scaling is used to ensure that the variables are in the same scale and avoid from further complexities caused by the large datasets. During the pre-processing of the data before building a machine learning model, feature scaling is one of the most critical steps because it enables the algorithm to converge faster, reduce the time spent finding support vectors and avoid saturation.

Using min-max scaling is a common technique that is useful when we need values within a bounded interval. Standardization is more practical for many machine learning algorithms [15].

Nonetheless, normalization is commonly used in order to deal with the problem of variables not falling into a similar range. In order to normalize, subtract the mean from the value and divide by the standard deviation [16]. This results in variables that all have a mean of 0 and a standard deviation of 1.

In this study, only ten thousand records are used from each month receipt data. Special care has taken to eliminate the class sampling issues. New column is introduced as season where all June month record value is 1 (summer) and all November month record value is 2 (winter). Furthermore, 90% of the records were used for training purpose and rest 10% were used for validation.

6. Training Classification

All four models are trained on the receipt data. Several result for accuracy and loss from a different algorithm are compared.

For this study of different model, only one R script is written, in which four different models were built for all four classification algorithms. These, models were trained with training data and validated by testing data and calculated the evaluation parameters.

After plotting the trained Naïve bayes model, there was a significant difference between mpg_no, pg_no and ag_no parameters for different months.

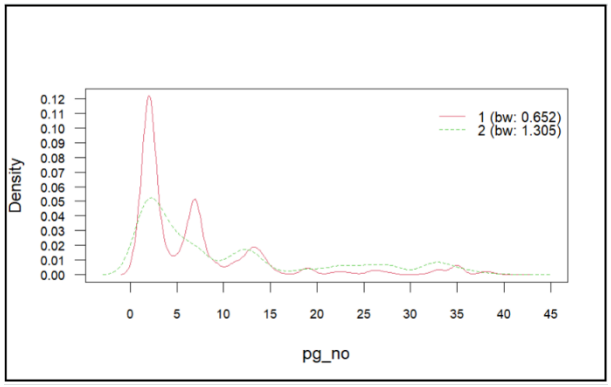

Figure 2: pg_no density graph

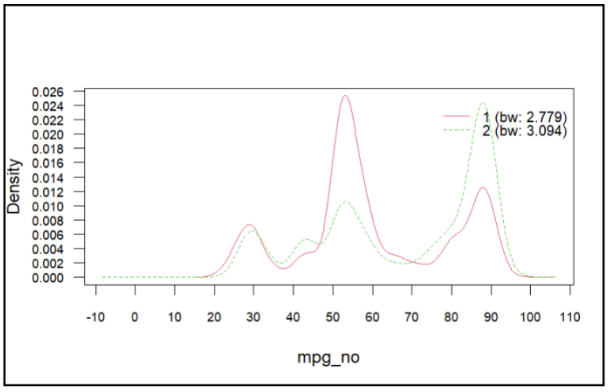

Figure 3: mpg_no density graph

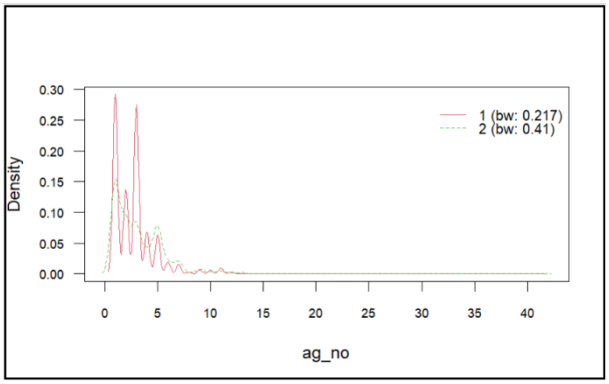

Figure 4: ag_no density graph

Also, Decision tree model used the mpg_no, pg_no and ag_no to build the decision tree. Where it created various branches and classified the records into class.

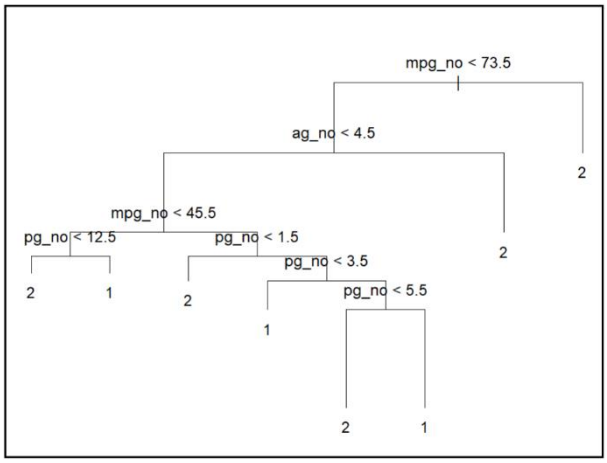

Figure 5: Decision Tree

7. Evaluation of Algorithms

The following sections show the results for the previously selected records and what further steps were taken to evaluate the models. In this study, after training the model various evaluation parameters result are created. Following is the sample evaluation results for random forest algorithm.

```
          Sensitivity : 0.7375
          Specificity : 0.8308
       Pos Pred Value : 0.8151
       Neg Pred Value : 0.7579
           Prevalence : 0.5028
       Detection Rate : 0.3708
 Detection Prevalence : 0.4549
    Balanced Accuracy : 0.7841
```

Figure 6: Confusion matrix for random forest algorithm

When the performances of the algorithms are evaluated in the dataset in terms of Accuracy performance metric, Random Forest classifier can be seen to give the best result of 0.78 (Figure 7). Decision tree, SVM and Naïve Bayes following this algorithm.

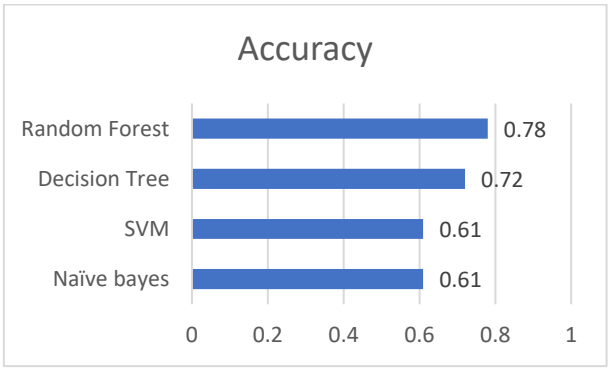

Figure 7: Accuracy of all classification algorithm

The Naïve Bayes classifier algorithm gave the best value of 0.84 for the recall performance metric. The random forest is the second algorithm with 0.74 recall value. SVM and Decision Tree algorithms followed, respectively, (Figure 8).

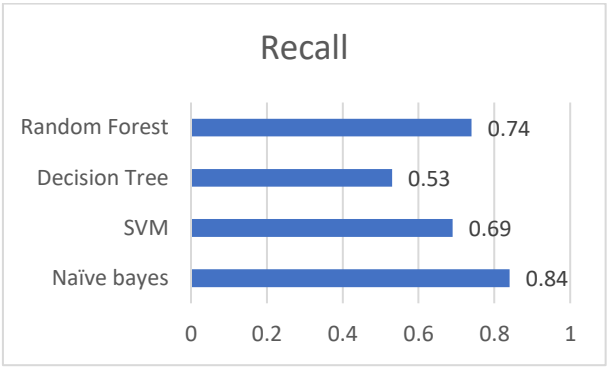

Figure 8: Recall of all classification algorithm

Additionally, the Random Forest classifier algorithm gave the best value of 0.78 for the F-measure performance metric, which is the harmonic mean of the precision and recall metrics. The Naïve bayes is the second algorithm with 0.69 F-measure value. SVM and Decision Tree algorithms followed, respectively, the naïve bayes algorithm according to F-measure values (Figure 9).

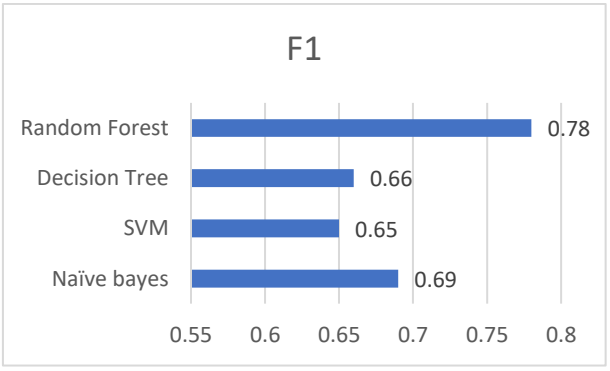

Figure 9: F1- measure of all classification algorithm

The Precision performance metric values obtained by machine learning methods on the dataset are shown in Figure 10. When (Figure 10) is examined, it can be seen that the Decision Tree classifier is given the maximum precision 0.85 in dataset. Random Forest algorithm is the second algorithm with 0.82 precision, and it is followed by SVM and Naïve Bayes respectively.

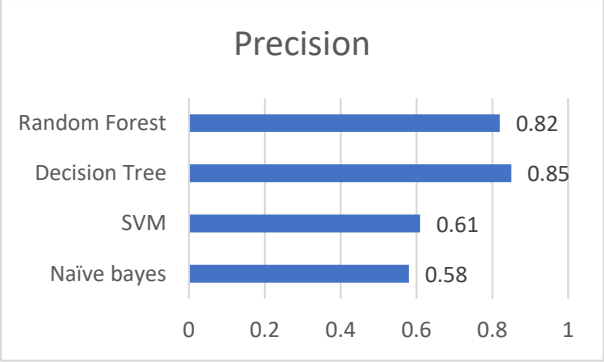

Figure 10: Precision of all classification algorithm

8. Summary

A major requirement for effective and successful marketing analytics is examining and analysing the valuable data of customer purchase records. Large volumes of data have become fundamental elements of modern societies. Its ability to analyse, solidify, and learn from them has become a useful aspect of competition and innovation.

In spite of the fact that big data can be considered useful for decision-making, large data does not necessarily lead to better marketing because it does not address some of the key challenges (such as the lack of feature engineering, the lack of effective analysis, etc.). Therefore, the prediction of customer purchasing behaviour can be greatly aided by machine learning techniques [17][18].

In this study using the machine learning techniques, the data were classified by using four different machine learning algorithms. According to the results, predictions can be done with around 78% accuracy.

The experimental results indicated that highest accuracy of receipt data classification was achieved by Random Forest Algorithm. Based on initial investigation, it is cleared that model can predict that when the clothing item was purchased on data. In future study, data from different months and countries will be used and classification model will be trained to provide multiclass classification for all seasons with maximum accuracy.

REFERENCES

[1] Liu, Qiong & Wu, Ying. (2012). Supervised Learning. 10.1007/978-1-4419-1428-6_451.

[2] IBM Cloud Education "Supervised Learning" [Online] https://www.ibm.com/cloud/learn/supervised-learning

[3] Jason Brownlee "Supervised and Unsupervised Machine Learning Algorithms" [online] https://machinelearningmastery.com/supervised-and-unsupervised-machine-learning-algorithms/

[4] Kotsiantis, Sotiris & Zaharakis, I. & Pintelas, P.. (2006). Machine learning: A review of classification and combining techniques. Artificial Intelligence Review. 26. 159-190. 10.1007/s10462-007-9052-3.

[5] S. Chowdhury and M. P. Schoen, "Research Paper Classification using Supervised Machine Learning Techniques," 2020 Intermountain Engineering, Technology and Computing (IETC), 2020, pp. 1-6, doi: 10.1109/IETC47856.2020.9249211

[6] Rish, Irina. (2001). An Empirical Study of the Naïve Bayes Classifier. IJCAI 2001 Work Empir Methods Artif Intell. 3.

[7] Kaviani, Pouria & Dhotre, Sunita. (2017). Short Survey on Naive Bayes Algorithm. International Journal of Advance Research in Computer Science and Management. 04.

[8] J. Wu, Z. Cai and X. Zhu, "Self-adaptive probability estimation for Naive Bayes classification," The 2013 International Joint Conference on Neural Networks (IJCNN), 2013, pp. 1-8, doi: 10.1109/IJCNN.2013.6707028

[9] Jair Cervantes, Farid Garcia-Lamont, Lisbeth Rodríguez-Mazahua, Asdrubal Lopez, A comprehensive survey on support vector machine classification: Applications, challenges and trends, Neurocomputing, Volume 408,2020, Pages 189-215, ISSN 0925-2312, https://doi.org/10.1016/j.neucom.2019.10.118.

[10] Kirchner, Antje, and Curtis S. Signorino. 2018. "Using Support Vector Machines for Survey Research." Survey Practice 11 (1). https://doi.org/10.29115/SP-2018-0001

[11] Jijo, Bahzad & Mohsin Abdulazeez, Adnan. (2021). Classification Based on Decision Tree Algorithm for Machine Learning. Journal of Applied Science and Technology Trends. 2. 20-28.

[12] Patel, Harsh & Prajapati, Purvi. (2018). Study and Analysis of Decision Tree Based Classification Algorithms. International Journal of Computer Sciences and Engineering. 6. 74-78. 10.26438/ijcse/v6i10.7478.

[13] IBM Cloud Education "Random Forest" [Online] https://www.ibm.com/cloud/learn/random-forest

[14] Ali, Jehad & Khan, Rehanullah & Ahmad, Nasir & Maqsood, Imran. (2012). Random Forests and Decision Trees. International Journal of Computer Science Issues(IJCSI). 9.

[15] Muhammad Ali, Peshawa & Faraj, Rezhna. (2014). Data Normalization and Standardization: A Technical Report. 10.13140/RG.2.2.28948.04489.

[16] Aniruddha Bhandari "Feature Scaling for Machine Learning: Understanding the Difference Between Normalization vs. Standardization" [Online] https://www.analyticsvidhya.com/blog/2020/04/feature-scaling-machine-learning-normalization-standardization/

[17] Das, Tapan. (2015). A customer classification prediction model based on machine learning techniques. 321-326. 10.1109/ICATCCT.2015.7456903.

[18] Choudhury, Adil & Nur, Kamruddin. (2019). A Machine Learning Approach to Identify Potential Customer Based on Purchase Behavior. 10.1109/ICREST.2019.8644458.

Other references for the creation and application of the model

Naive Bayes Classification in R https://www.r-bloggers.com/2021/04/naive-bayes-classification-in-r/

Joseph Rickert "Naive Bayes: A Generative Model and Big Data Classifier" https://rviews.rstudio.com/2016/11/02/naive-bayes-a-generative-model-and-big-data-classifier/

"Creating Visualizations" https://db.rstudio.com/best-practices/visualization/

James Le,"Support Vector Machines in R" https://www.datacamp.com/community/tutorials/support-vector-machines-r

"Decision Trees in R" https://www.r-bloggers.com/2021/04/decision-trees-in-r/

Cory Maklin "Random Forest In R" https://towardsdatascience.com/random-forest-in-r-f66adf80ec9

Github https://github.com/topics/random-forest-classification