Online purchase prediction

\*MohamedNurAbshir

\*Dept. of CSE, \*\*Student Dept. of CSE

**Abstract**. Due to today’s transition from visiting physical stores to online shopping, predicting customer behavior in the context of e-commerce is gaining importance. It can increase customer satisfaction and sales, resulting in higher conversion rates and a competitive advantage, by facilitating a more personalized shopping process. By utilizing clickstream and supplementary customer data, models for predicting customer behavior can be built. This study analyzes machine learning models to predict a purchase, which is a relevant use case as applied by a large German clothing retailer. Next, to comparing models this study further gives insight into the performance differences of the models on sequential clickstream and the static customer data, by conducting a descriptive data analysis and separately training the models on the different datasets. The results indicate that a Random Forest algorithm is best suited for the prediction task, showing the best performance results, reasonable latency, offering comprehensibility and a high accuracy. Regarding the different data types, models trained on sequential session data outperformed models trained on the static customer data by far. The best results were obtained when combining both datasets.

**INTRODUCTION**

Predicting customer behavior in the context of e-commerce is becoming more important nowadays. It increases customer satisfaction and sales, by facilitating an increase of customer experience through personalization, recommendations and special offers. By utilizing clickstream and additional customer data, predictions can be carried out, ranging from customer classification, purchase prediction, and recommender systems to the detection of customer churn. A variety of machine learning models and data are available to conduct these kinds of predictions.

**RESEARCH METHOD**

This research was structured based on the Cross Industry Standard Process for Data Mining (CRISP-DM) methodology. Suitable models, being boosted three algorithms, Random Forest (RF), Support Vector Machine (SVM), Naïve Bayes (NB), were identified through a literature research. Following, algorithms were trained on three two datasets, the sequential session data, the static customer data and a combined dataset, then evaluated and compared based on different performance metrics, and which one has best accuracy. Random Forest algorithm is best suited for the prediction task, showing the best performance results, reasonable latency, offering comprehensibility and a high accuracy. All algorithms as well as the evaluation and comparison were implemented in Python.

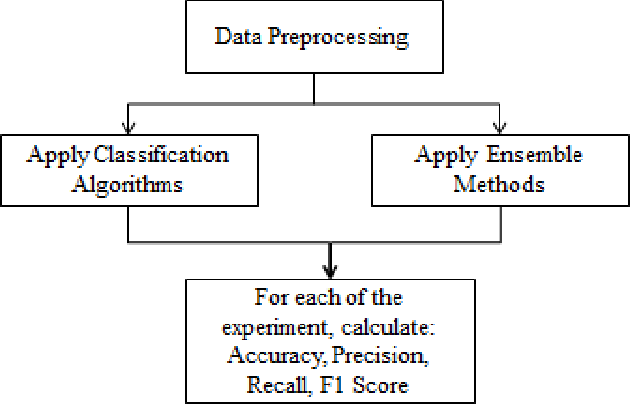
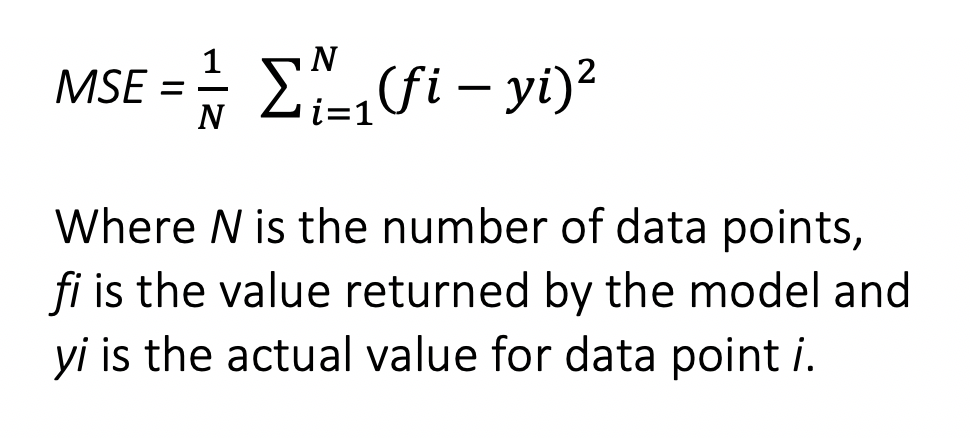


Fig. 1. Online purchase predection

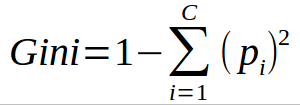
**LEARNING ALGORITHMS EVALUATED**

1. Random Forest (RF): The Random Forest Algorithm is composed of different decision trees, each with the same nodes, but using different data that leads to different leaves. It merges the decisions of multiple decision trees in order to find an answer, which represents the average of all these decision trees. This algorithm used to solve both regression and classification.

* Regression Problems: When using the Random Forest Algorithm to solve regression problems, you are using the mean squared error (MSE) to how your data branches from each node.

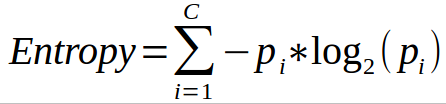
This formula calculates the distance of each node from the predicted actual value, helping to decide which branch is the better decision for your forest. Here, yi is the value of the data point you are testing at a certain node and fi  is the value returned by the decision tree.

* Classification problems: When performing Random Forests based on classification data, you should know that you are often using the Gini index, or the formula used to decide how nodes on a decision tree branch.

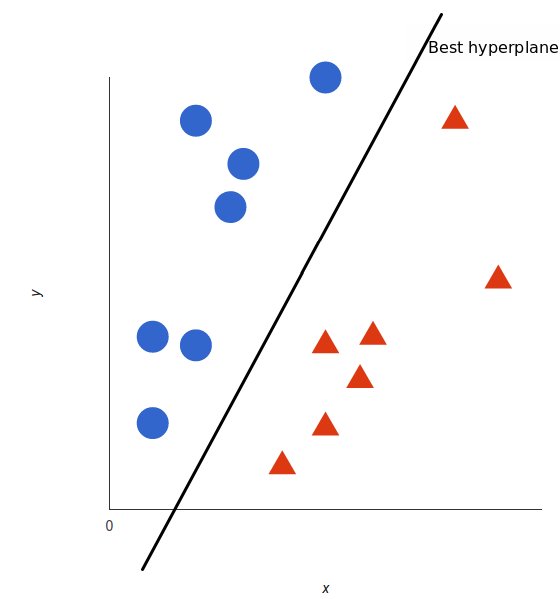


This formula uses the class and probability to determine the Gini of each branch on a node, determining which of the branches is more likely to occur. Here, pi represents the relative frequency of the class you are observing in the dataset and c represents the number of classes.

We can also use entropy to determine how nodes branch in a decision tree.

****

Entropy uses the probability of a certain outcome in order to make a decision on how the node should branch. Unlike the Gini index, it is more mathematical intensive due to the logarithmic function used in calculating it.

1. Support Vector Machine(SVM): Support Vector Machine (SVM) is a supervised machine learning algorithm that can be used for both classification or regression challenges. However,  it is mostly used in classification problems. In the SVM algorithm, we plot each data item as a point in n-dimensional space (where n is a number of features you have) with the value of each feature being the value of a particular coordinate. Then, we perform classification by finding the hyper-plane that differentiates the two classes very well.

* Here’s a trick: SVM doesn’t need the actual vectors to work its magic, it actually can get by only with the dot products between them. This means that we can sidestep the expensive calculations of the new dimensions. Imagine the new space we want:

**z = x² + y²**

* Figure out what the dot product in that space looks like:

**a · b = xa · xb  +  ya · yb  +  za · zb**

**a · b = xa · xb  +  ya · yb +  (xa² + ya²) · (xb² + yb²)**

* Tell SVM to do its thing, but using the new dot product — we call this a karnel function.

That’s it! That’s the **kernel trick**, which allows us to sidestep a lot of expensive calculations. Normally, the kernel is linear, and we get a linear classifier. However, by using a nonlinear kernel (like above) we can get a nonlinear classifier without transforming the data at all: we only change the dot product to that of the space that we want and SVM will happily chug along.

**RESULT AND DISCUSSION**

The obtained results indicate that the RF performed best while showing reasonable prediction latency. Regarding the comprehensibility, no difference between the different algorithms was observed. The performance of different datasets shows that a combined dataset leads to the best results, where customer information enhances the results only slightly. An overview of the results regarding the different datasets and algorithms concerning the ROC AUC value can be observed in Table 1. Further, a promising effect regarding time-consuming feature engineering was observed for the RF, where fewer and less engineered features led to better results than a larger amount of more heavily engineered features as used for the other algorithms.

Table 1: results for all algorithms and datasets.

|  |
| --- |
| Algorithm Combined Data Sequence Data Customer Data Accuracy |

RF 0.82 0.81 0.67 0.98

SVM 0.67 0.74 0.39 0.93

|  |
| --- |
|  |

NB 00 00 00 00

**CONCULATION**

This study shows that web shop sessions can be well categorized as buying or no buying sessions, with an RF showing the best performance. Further, by training on different datasets this study was able to emphasize that session based data, mainly generated from the customer clickstream, is most important for predicting purchase probabilities. This indicates that personal customer information, often associated with privacy concerns and regulations, is not necessarily needed to predict customer behavior well. Additionally, by training an RNN on less engineered features, it was displayed that stateful models perform well while requiring less time-consuming feature engineering, when detecting sequential patt