Machine Learning with Applications 7 (2022) 100196

Contents lists available at ScienceDirect

Machine Learning with Applications

journal homepage: www.elsevier.com/locate/mlwa

Enhancing the pattern recognition capacity of machine learning techniques: The importance of feature positioning

Debora Di Caprio [a,*,1], Francisco J. Santos-Arteaga [b,1]

[a] Department of Economics and Management, University of Trento, Trento, Italy
[b] Faculty of Economics and Management, Free University of Bolzano, Bolzano, Italy

ARTICLE INFO

Keywords:
Feature positioning
Information retrieval
Pattern recognition
Machine learning
Decision trees

ABSTRACT

We design several algorithms representing evaluation processes of different complexity, ranging from basic environments based on a predetermined number of features to complex structures involving alternatives defined through decision trees whose number of nodes is determined by the cardinality of the respective power sets. The sequential structure of these evaluation processes builds on the information retrieval behavior of users in online search environments. The algorithms generate two strings of data, namely, numerical evaluations determining the retrieval behavior of users and the subsequent choices made by the latter. The way the output obtained from the algorithms is inputted within the vectors summarizing the complexity of the evaluation processes conditions the capacity of machine learning techniques to categorize them correctly. The main purpose of the research is to illustrate numerically two main results. First, machine learning techniques categorize processes correctly even if their characteristic features are presented in a way that prevents their identification using standard statistical techniques. Second, the accuracy of the categorization capacities of these techniques can be substantially enhanced by describing the retrieval processes in the way required to implement standard statistical analyses. We perform a battery of tests using machine learning techniques to demonstrate and analyze these results. Their applicability to classification and prediction problems in medical environments, particularly those constrained by the quality of the data available, is emphasized.

1. Introduction

The current study builds on the online information retrieval processes of decision makers (DMs) analyzed by decision theorists and management scholars. The raw data available describing the retrieval behavior of DMs provides information on the pages clicked and the subsequent click through rates (CTRs) generated through the different queries (Chitika, 2013; Dean, 2019). This information, essential to understand the potential retrieval patterns of DMs, is generally used to highlight the importance of ranking positioning within the results displayed by a search engine (Epstein & Robertson, 2015; Pan et al., 2007). Researchers dealing with decision theoretical models have focused on extrapolating the utility functions leading to this type of observed behavior (Basu, 2018; Victorelli, Dos Reis, Hornung, & Prado, 2020). However, as intuition suggests, the behavioral patterns emerging from the data may be generated by different sequential information retrieval strategies of varying complexity.

An important observation is due here. The main empirical studies describing the retrieval behavior of DMs do not provide any details regarding how the data has been inputted before being analyzed (Chitika, 2013; Dean, 2019). That is, the order of the entries composing the data vectors, namely, the alternatives clicked from the page of results displayed by the engine, is not relevant when analyzing variables that define averages across observations such as the CTRs. As a result, this information is not reported when considering research papers that analyze trends in the information retrieval behavior of DMs performing online queries (Li, Duan, Zheng, Wang, & Wang, 2020). However, it is generally understood that this information is extremely important when considering standard statistical analyses. This is the case since the order of the elements composing the entries defining the data vectors conditions the results derived from the subsequent regressions.

The main objective of the current research is to analyze the identification capacities displayed by machine learning techniques when categorizing evaluation patterns of different complexity. We illustrate how the identification capacities of these techniques are determined, and can be enhanced, by the way the features are ordered within the vectors defining the alternatives. The novel characteristics of the proposed approach can be summarized as follows

* Corresponding author.
E-mail addresses: debora.dicaprio@unitn.it (D. Di Caprio), fsantosarteaga@unibz.it (F.J. Santos-Arteaga).
[1] Both authored the manuscript to which they have contributed equally.

https://doi.org/10.1016/j.mlwa.2021.100196
Received 29 May 2021; Received in revised form 20 October 2021; Accepted 24 October 2021
Available online 6 November 2021

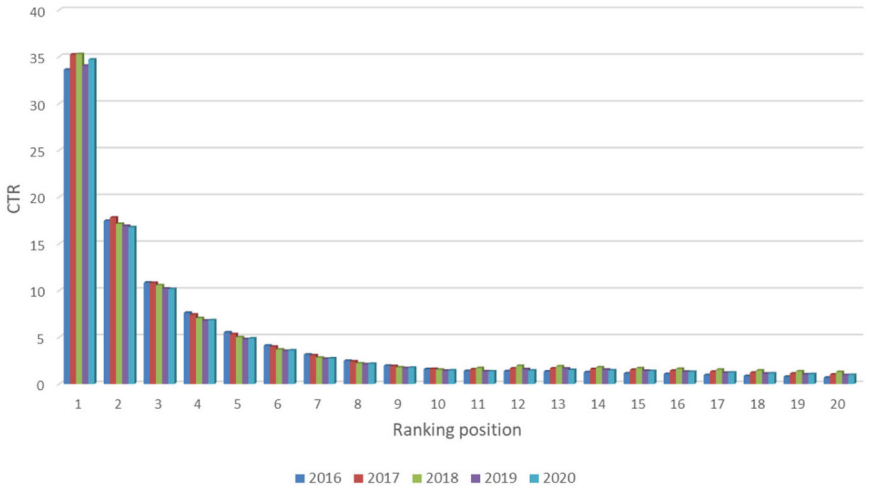

Fig. 1. Annual organic CTRs for international desktop searches.
Source: Advanced Web Ranking (2021).

1. We define sequential evaluation processes of different complexity whose structure builds on the information retrieval behavior of users in online search environments.
2. We illustrate how these evaluation processes can be simulated by algorithms of different complexity.
3. We demonstrate numerically how the way the output obtained from the algorithms is inputted within the vectors summarizing the evaluation processes conditions the categorization capacity of machine learning techniques. This result is illustrated using a generic numerical example as well as calibrating the behavior of DMs to the CTRs observed in real life evaluation environments.
4. We highlight the superior categorization capacities of machine learning techniques relative to standard statistical analyses, particularly when considering misprints or errors in the matrices defining the input or independent variables.

This last quality is extremely important in areas such as medicine, where a general distrust in artificial intelligence and machine learning techniques is leaning the profession towards the use of standard statistical tests (Bae et al., 2020; Lancet, 2021; Wynants et al., 2020). The capacity of machine learning techniques to overcome the potential identification problems arising from the existence of data misprints should provide a powerful argument in support of their use.

1.1. Analyzing the information retrieval behavior of DMs

The reference framework on which we build the algorithms designed to test the identification capacities of different machine learning techniques is that of online information retrieval. This choice is justified by the fact that algorithms based on retrieval patterns corresponding to users exhibiting different degrees of behavioral complexity can be simulated and framed within an intuitive presentation framework.

The information assimilation capacities of DMs constitute one of the main research topics in decision theory and have been analyzed from a variety of academic perspectives, ranging from operations research to psychology (Schwartz, 2015; Tavana, Caprio, & Santos-Arteaga, 2017). In this regard, the availability of data describing the CTRs that result from the searches performed by DMs within an unmonitored environment provides a substantial amount of intuition regarding the complexity of their retrieval behavior.

Fig. 1 presents the annual organic CTRs derived from international desktop searches over the last five years. The observed CTRs are consistent both across countries and through time, with users concentrating their searches on the initial alternatives composing the first page of results provided by search engines. This behavior validates the results

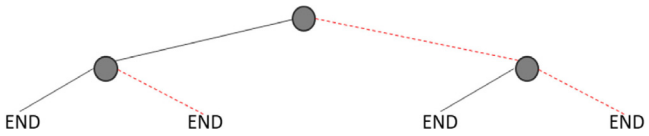

Fig. 2. Decision tree describing the evaluation of two alternatives.

```
nrows = 20;
ncols = 1000000;
A = zeros(nrows,ncols);
for i = 1
   for j = 1:ncols

    A(i,j)=rand(1);
     if A(i,j)>0.5
     A(11,j)=1;
     A(i+1,j)=rand(1);

      if A(i+1,j)>0.5
      A(12,j)=2;
      end

     else
     A(i+1,j)=rand(1);
      if A(i+1,j)>0.5
      A(11,j)=2;
      end
    end

   end
end
```

Fig. 3. Code of an ordered algorithm where DMs consider evaluating two alternatives.

obtained in multiple empirical experimental studies illustrating the sequential evaluation processes followed by DMs when retrieving information form the ranking of alternatives provided by a search engine (Epstein & Robertson, 2015; Lewandowski & Kammerer, 2020).

Consider now the standard definition of CTR

$$\text{CTR of alternative } i = \frac{\text{Number of users clicking on the link to alternative } i}{\text{Number of users performing a search}} \tag{1}$$

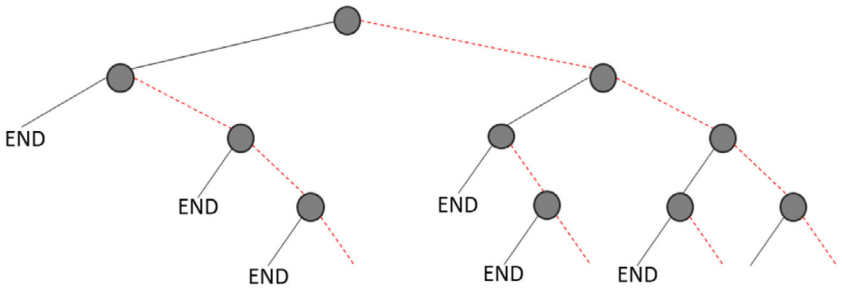

Fig. 4. Initial section of the decision tree describing the search for two satisfying alternatives.

Given this definition and the conditioning of the sequential evaluation processes on the ranking of the alternatives, two main completely different types of DMs can be assumed to generate the CTRs observed. On one hand, DMs could lack a structured evaluation process and base their behavior on a predetermined set of acceptance probabilities defined for each of the alternatives composing the ranking. On the other hand, DMs could follow a completely structured search process aiming to observe the largest potential number of alternatives satisfying a given set of requisites determining their acceptance probabilities. In other words, the value of the CTRs does not reflect whether DMs follow highly elaborated evaluation patterns or implement heuristic mechanisms.

When designing algorithms to simulate the retrieval behavior of DMs, each run can be assumed to represent a search query. As a result, the law of large numbers implies that a trivial algorithm defining ten independent realizations from ten random variables leads to a behavior identical to the one observed in the data. The algorithm only requires a set of acceptance thresholds determined by the corresponding CTRs. At the same time, a complex algorithm accounting for each and every potential combination that may be followed by a user as he retrieves information leads to the same result. In this case, the user considers evaluating the whole set of alternatives composing the first page of results based on a set of predetermined acceptance thresholds.

We elaborate on these scenarios and their consequences relative to the capacity of machine learning techniques to differentiate between both types of DMs through the next section.

2. Contribution

Consider the case where a DM evaluates the first two alternatives from a list and stops afterwards. The DM selects the alternatives performing above a given threshold defined by a predetermined selection criterion. This scenario differs substantially from one where the DM sets out to find two alternatives from a given list satisfying a given selection criterion. In this case, the alternatives are not necessarily the first two but can be located anywhere within the list.

The differences in retrieval abilities between both scenarios and the complexity of the resulting algorithms designed to simulate both sequential processes are substantial.

- In the first case, we can define a simple decision tree consisting of three decision nodes that determine the retrieval behavior of DMs based on two potential realizations per node and the corresponding threshold values. Fig. 2 illustrates a binary decision tree representing this type of evaluation scenario, while Fig. 3 presents a basic algorithm describing the resulting information retrieval behavior. The continuous lines within the tree represent realizations from alternatives located above the predetermined threshold, while the dotted ones refer to evaluations underperforming relative to the threshold value. The code defined in Fig. 3 describes this retrieval process, imposing a threshold of 0.5 for stochastic realizations of the evaluations uniformly distributed within the interval [0, 1]. A more detailed description of the retrieval process coded in Fig. 3 will be provided in Fig. 6 within the next section.
- In the second case, the information retrieval behavior is considerably more complex, since DMs must consider the potential evaluations arising from the whole set of alternatives through the retrieval process. If the list consists of ten alternatives, as is the case when retrieving information from an online search engine, the number of decision nodes composing the subsequent decision tree equals 55. Fig. 4 illustrates the initial section of a decision tree describing the information retrieval process of a DM who aims at finding two satisfying alternatives from a given list. A more detailed description of this type of retrieval process will be provided in Fig. 7 within the next section.

The importance of feature positioning within the vectors describing the retrieval processes of DMs becomes evident now. Clearly, if we input the alternatives clicked according to the position in which they are located within the ranking, the behavior of DMs can be differentiated by machine learning techniques simply based on positioning. That is, the way the features characterizing the retrieval processes are inputted can be used to enhance the categorization capacity of machine learning techniques. The main contribution of the current paper consists in illustrating the extent of this enhancement when dealing with different types of retrieval scenarios characterized by the assimilation capacities of DMs and the number of alternatives defining the search.

2.1. A numerical illustration

Five evaluation processes derived from a retrieval scenario accounting for six alternatives with a threshold value of 0.50 have been described within the first set of columns of Table 1. Note that each column vector describes the evaluation process of a DM and has been divided in two distinct sections. The upper one, denoted ‘initial evaluation’, presents the realizations observed for each of the alternatives composing the ranking, which are drawn from a uniform distribution defined on [0, 1]. The lower section, denoted ‘threshold satisfying alternatives’, identifies the alternatives clicked by the DMs as determined by their realizations and the value of the threshold. The corresponding MATLAB code is provided in Figure A1 within Appendix A section.

In this case, denoted ‘ordered evaluation process’, the alternatives satisfying the threshold requirement have been reported according to the position where they are located within the first ten entries of the vector defining the information retrieval process. The remaining entries of the vector have been defined by zeros. Fig. 5 describes the retrieval framework corresponding to an ordered evaluation process accounting for two alternatives. The retrieval of information is defined by two independent and unrelated evaluations, i.e., x_1 and x_2. That is, the evaluations obtained when observing the first alternative, do not condition the inputs of the subsequent binary nodes. The final nodes resulting from the retrieval process correspond to those derived from each binary node for each alternative composing the ranking.

The second set of columns within Table 1 represents a binary evaluation setting where the alternatives satisfying the threshold requirement have been reported in the initial entries composing the lower section of the vector. Consequently, this case has been denoted

Table 1
Characterizing the retrieval behavior of DMs across evaluation processes: The case with six alternatives.

	TYPE OF EVALUATION PROCESS														
	ORDERED (ORD)					GROUPED (GRP)					COMPLETE				
	DM1	DM2	DM3	DM4	DM5	DM1	DM2	DM3	DM4	DM5	DM1	DM2	DM3	DM4	DM5
	0.7094	0.1190	0.7513	0.5472	0.8143	0.0573	0.8257	0.4513	0.6483	0.2288	0.6977	0.6533	0.4911	0.7755	0.1286
	0.7547	0.4984	0.2551	0.1386	0.2435	0.3010	0.4445	0.2500	0.6147	0.4235	0.8935	0.7541	0.4035	0.6379	0.2208
	0.2760	0.9597	0.5060	0.1493	0.9293	0.5217	0.9821	0.9554	0.4697	0.2736	0.7012	0.2800	0.8774	0.7251	0.4536
	0.6797	0.3404	0.6991	0.2575	0.3500	0.5619	0.5783	0.1427	0.5778	0.4446	0.9373	0.6031	0.7082	0.8946	0.4653
Initial	0.6551	0.5853	0.8909	0.8407	0.1966	0.2416	0.2344	0.5126	0.9113	0.6275	0.0109	0.8888	0.8265	0.7680	0.0898
evaluation	0.1626	0.2238	0.9593	0.2543	0.2511	0.9127	0.8106	0.9719	0.3762	0.5346	0.8197	0.6432	0.0107	0.2068	0.5067
	0	0	0	0	0	0	0	0	0	0	0.0343	0.7861	0.0956	0.0433	0.0765
	0	0	0	0	0	0	0	0	0	0	0.5162	0	0.5433	0.1536	0.1665
	0	0	0	0	0	0	0	0	0	0	0	0	0.8562	0.7183	0.7477
	0	0	0	0	0	0	0	0	0	0	0	0	0.2644	0	0.3346
	1	0	1	1	1	3	1	3	1	5	1	1	3	1	6
	2	0	0	0	0	4	3	5	2	6	2	2	4	2	9
	0	3	3	0	3	6	4	6	4	0	3	4	5	3	0
Threshold	4	0	4	0	0	0	6	0	5	0	4	5	8	4	0
satisfying	5	5	5	5	0	0	0	0	0	0	6	6	9	5	0
alternatives	0	0	6	0	0	0	0	0	0	0	8	7	0	9	0
	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

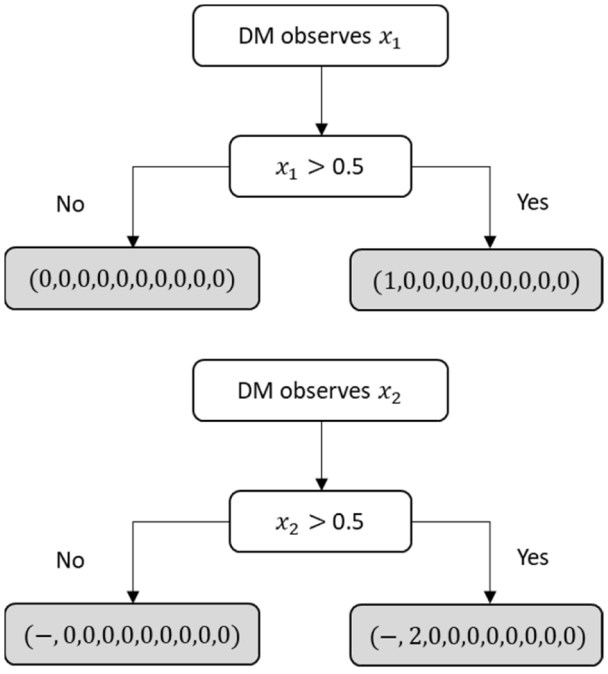

Fig. 5. Retrieval framework of the ordered evaluation process with two alternatives.

'grouped evaluation process'. The corresponding algorithm is described in Figure A2 within Appendix A. Fig. 6 illustrates the basic structure of a grouped evaluation process accounting for two alternatives. That is, a binary tree based on the two evaluations defining the retrieval process, x_1 and x_2. Note how the realizations of the previous evaluations determine the entries inputted in the vectors. These interactions define an interrelated evaluation framework that requires enhanced assimilation capacities on the side of DMs and the design of a more complex retrieval algorithm.

The third set of columns within Table 1 describes five alternatives from the latter, increasingly complex, 'complete evaluation process'. In this case, DMs aim at finding six alternatives satisfying a predetermined criterion out of a total of ten. The code of the corresponding algorithm can be found in Figure A3 within Appendix A. Note how the alternatives satisfying the threshold requirement have also been grouped within the initial entries composing the lower section of the vector. The complete evaluation process requires introducing an even more complex set of relations than those described in the grouped setting. However, both retrieval frameworks converge when considering the whole set of alternatives. The retrieval structure described in Fig. 7 is more elaborated than the binary decision tree presented in Fig. 6. That is, the grouped scenario is constrained by a predetermined number of evaluations that binds independently of the value of the realizations observed. The complete evaluation process requires defining a more complex retrieval setting since DMs may have to proceed through the whole ranking of alternatives. i.e., x_i with $i = 1, \ldots, 10$.

Fig. 8 complements these results by illustrating the retrieval patterns obtained from 2000 queries per evaluation process when DMs aim to observe ten satisfying alternatives. The figure follows from pairing the value of the initial evaluations with the threshold satisfying alternatives clicked per search query. That is, the data illustrated consist of two column vectors, one describing the evaluations performed by the DMs and the other representing the subsequent alternatives clicked. As was the case in Table 1, each evaluation process has been assigned a threshold value of 0.5 per alternative. It therefore follows that the ordered setting only assigns a value of zero to realizations located below 0.5.

On the other hand, the grouped and complete processes only display this feature for the first alternative composing the ranking. This is the case since, whenever the first alternative is clicked, the corresponding entry of the initial evaluation vector must be associated with a realization higher than 0.5. However, when the first alternative is not clicked, the first entry of the vector describing the threshold satisfying alternatives corresponds to a number different from one. This feature applies to the whole set of alternatives and explains the different patterns observed in the grouped and complete processes when compared to the ordered one. It also highlights how the order in which the alternatives are inputted may determine the capacity of any categorization technique to identify the corresponding retrieval patterns.

Two observations are due. First, when considering a scenario based on two satisfying alternatives, the value of the alternatives clicked suffices to identify those DMs implementing a complete evaluation processes. This is the case even if the information is placed within the initial entries of the vector describing the retrieval processes. That is, the two initial entries of the vectors defining ordered and grouped retrieval processes will be composed by zeros, ones, and twos. On the other hand, the two initial entries of the vectors describing complete

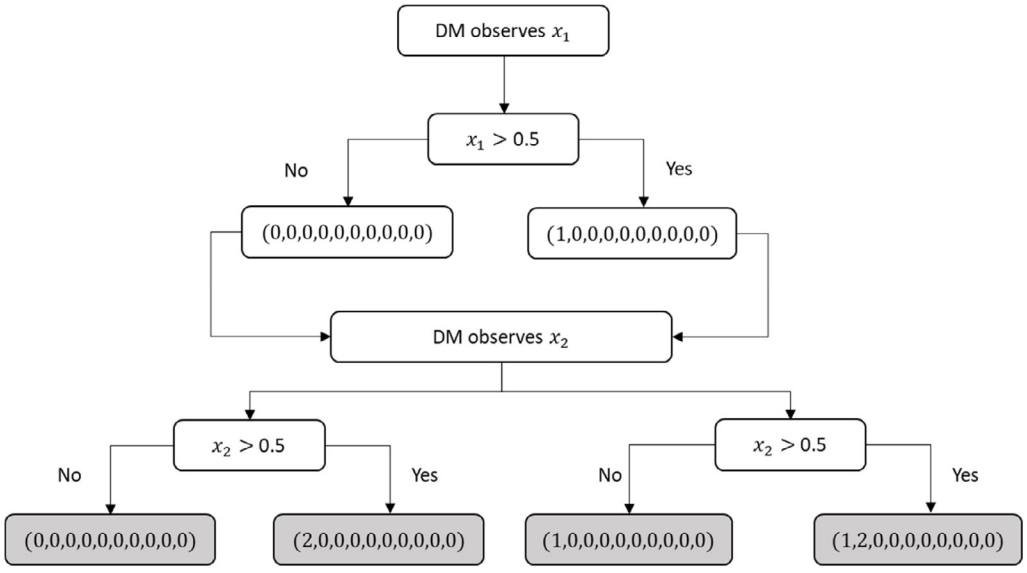

Fig. 6. Retrieval framework of the grouped evaluation process with two alternatives.

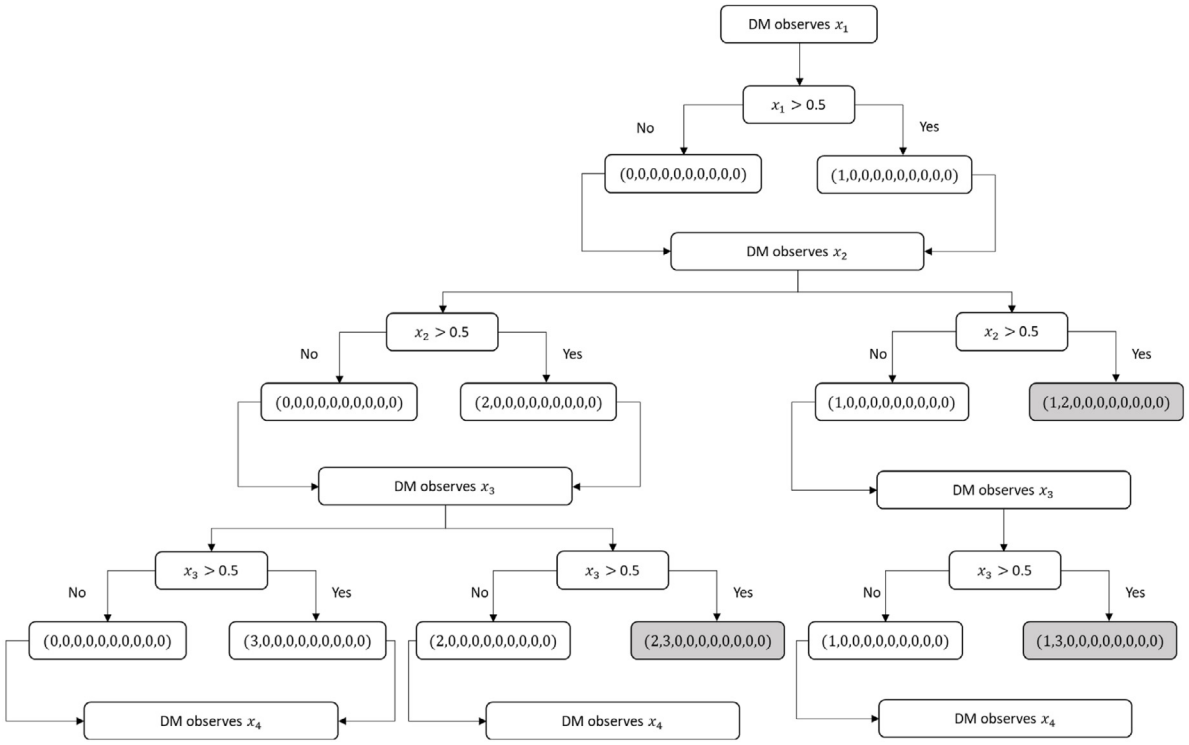

Fig. 7. Initial section of the complete evaluation process with two alternatives.

retrieval processes can display any number up to ten. This feature suffices on its own to differentiate between both process categories. Second, the ability of machine learning techniques to differentiate between processes vanishes as we increase the number of alternatives defining the search and approach the limit value of ten.

Given these numerical simulations, it can now be intuitively understood how

- the simplest ordered algorithm, where DMs perform ten independent evaluations, with a predetermined threshold value defined for each alternative, and
- the complex complete algorithm, where DMs aim at evaluating the ten alternatives delivered by a search engine within its first page of results

lead to the same stochastic retrieval structure and deliver identical CTRs.

3. Identification results

In computer science terminology, classification problems consist of a set of predictors, that is, features describing the different alternatives, defined via n-dimensional vectors and an outcome per alternative, namely, the class to which the alternative belongs. Machine learning

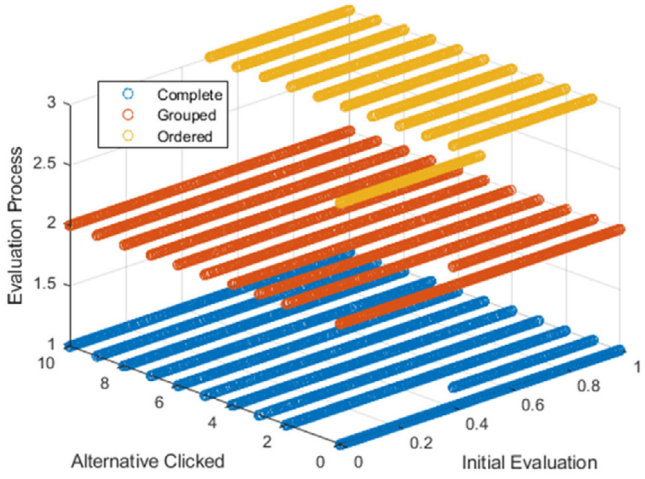

Fig. 8. Retrieval profiles generated by the different evaluation processes within the ten alternatives scenario.

Table 2
Seconds required by the algorithms to generate the corresponding evaluation processes.

Evaluation process	Number of alternatives		
	Two	Six	Ten
Ordered	0.112686	0.294349	0.339749
Grouped	0.129391	0.532126	63.807632
Complete	0.475651	41.757647	70.184606

techniques assess the features of each alternative together with its class and learn from them so that whenever an alternative is observed, the class to which it belongs can be predicted. Clearly, in the current setting, the alternatives refer to the retrieval processes defined by the different types of DMs – who determine the classes – when evaluating the information provided by a search engine.

Table 2 describes the number of seconds required for the code to run and generate the output describing the evaluation profiles defined by the different types of DMs. As illustrated in Table 1, the algorithms generate column vectors composed by 20 rows to describe the retrieval process triggered by a search query. The first half of the vector corresponds to the value of the realizations observed by DMs while the second half describes the pages being clicked. Each scenario simulated within Table 2 generates 1,000,000 queries, constituting the columns of the corresponding matrices. The time required to run the algorithms reflects the complexity of the retrieval structures being analyzed.

The differences arising across the scenarios simulated and within them are indeed substantial. When considering two satisfying alternatives, a noticeable difference in processing time arises between the ordered and grouped algorithms and the complete one, whose relative complexity becomes already evident. These differences are exacerbated when accounting for six alternatives and become even larger when ten alternatives are considered. Note how, in this latter scenario, the grouped and complete scenario converge in their processing times. As stated before, both algorithms share an identical binary retrieval structure when evaluating the whole set of alternatives available. The grouped evaluation process requires DMs to group the pages clicked within the initial entries of the lower half of the vector describing their retrieval behavior per search query. This feature imposes additional cognitive requirements on the DMs, whose complexity increases as the binary tree progressively accounts for all the potential alternatives composing the initial page of results.

Through this section, we illustrate how machine learning techniques can generally differentiate among the evaluation scenarios generated by the different types of DMs. At the same time, displaying the alternatives clicked according to their position within the upper section of the vector defining the retrieval process improves the identification capabilities of these techniques substantially. These results are presented in Table 3, where several machine learning techniques have been applied to identify the DMs implementing a complete evaluation scenario relative to the ordered and grouped ones. That is, the ORD entries of the table refer to categorization problems involving ordered and complete retrieval processes, while the GRP entries focus on grouped and complete processes. A total of 2000 queries have been simulated per retrieval processes within each evaluation scenario. The numerical results obtained remain unchanged when increasing or decreasing the number of queries simulated within a reasonable range.

The basic setting accounts for retrieval processes defined exclusively by the lower section of the evaluation vectors. Note how adding the upper section of the vectors, namely, the enhanced feature setting, improves slightly the categorization capacity of the techniques. However, it is the distribution of alternatives within the lower section of the vector according to the position reported within the upper one what increases their identification capacities. This quality is particularly evident in evaluation scenarios consisting of ten alternatives. As already stated, the algorithms presented in Figures A2 and A3 within Appendix A coincide when considering ten alternatives, distorting the identification capacity of the techniques considerably.

A technical note is due. The program applies a default 5-fold cross-validation procedure to protect against overfitting and estimate the predictive accuracy of the models. That is, the program partitions the data set in 5 folds. For each validation fold, the program trains a model using observations not contained in the validation fold and assesses its performance using the data from the validation-fold. Finally, it calculates the average validation error over all folds.

Table 4 displays the confusion matrices corresponding to the best performing techniques within each scenario analyzed in Table 3. Class 1 corresponds to the complete evaluation process while Class 2 refers to either the ordered or grouped one. Clearly, machine learning techniques can correctly categorize the evaluation processes in scenarios with two and six alternatives, while facing difficulties when considering ten alternatives. In this latter case, the improvement obtained when providing the whole vector of features, including the realizations composing its upper section, is marginal. On the other hand, when the lower section of the vectors describes the alternatives clicked according to the order displayed within the upper section, the categorization capacity of these techniques increases substantially.

These results are corroborated by the value of the area under the receiver operating characteristic (ROC) curve and through two additional accuracy tests described in Table 5. The F-measure is defined as the harmonic mean of precision and recall, that is, given the matrix entries described in Table 4(a)

$$\text{F-measure} = \frac{TP}{TP + 0.5\,(FP + FN)} \tag{2}$$

Its highest value is 1, indicating perfect precision and recall, and the lowest is 0, if either precision or recall equal zero.

The Kappa statistic was originally proposed as a chance-corrected version of accuracy, and is defined as follows

$$\text{Kappa statistic} = \frac{2\,(TP \times TN - FN \times FP)}{(TP + FP) \times (FP + TN) + (TP + FN) \times (FN + TN)} \tag{3}$$

Its worst value is given by −1, describing a perfectly wrong prediction, while the best one equals 1, corresponding to a perfect classification.

For completeness, we apply the retrieval scenario with ten alternatives to the average of the CTRs defined over the period described in Fig. 1. The average CTRs together with those generated by the ordered and complete algorithms for a total of 1,000,000 simulated queries are

Table 3
Pattern recognition capacities across machine learning techniques.

Evaluation Scenario	Features	Evaluation processes	Tree			Support Vector Machine										K-nearest neighbors						Ensemble				
			Fine	Medium	Coarse	Linear Discriminant	Quadratic Discriminant	Logistic Regression	Gaussian Naïve Bayes	Kernel Naïve Bayes	Linear	Quadratic	Cubic	Fine Gaussian	Medium Gaussian	Coarse Gaussian	Fine	Medium	Coarse	Cosine	Cubic	Weighted	Boosted Trees	Bagged Trees	Subspace Discriminant	Subspace KNN	RUSBoosted Trees
Ten	Basic	GRP	50.5	49.3	49.2	49.5	-	49.5	-	51.3	50.6	49.5	50.7	49.8	49.7	49.4	50.0	48.4	49.9	48.5	48.6	48.9	50.2	49.7	50.0	50.0	49.5
		ORD	99.6	99.4	98.4	97.5	94.1	98.2	94.3	96.4	98.3	99.1	99.1	99.3	99.2	98.3	99.3	98.8	97.4	98.9	98.7	99.3	99.4	95.7	97.3	96.5	99.4
	Enhcd	GRP	51.5	49.8	49.2	49.0	-	49.1	-	49.7	49.2	51.1	48.1	48.4	50.0	49.5	48.4	49.0	48.9	49.6	49.1	48.7	50.2	49.0	48.9	49.2	49.9
		ORD	98.8	99.2	98.3	97.7	98.8	98.6	94.4	95.8	98.5	99.2	99.4	98.1	99.2	98.4	99.1	99.1	98.4	99.2	98.9	99.2	99.2	98.7	97.6	99.5	99.1
Six	Basic	GRP	96.6	96.6	94.5	-	-	89.5	-	94.2	89.5	96.2	96.5	96.0	94.9	92.3	61.6	94.6	91.1	94.8	94.7	95.3	96.5	74.5	87.3	50.0	96.6
		ORD	98.9	99.0	98.1	-	-	98.3	-	93.9	98.2	98.7	98.8	98.9	98.5	98.2	95.0	98.4	97.4	98.5	98.4	98.5	98.8	90.5	96.0	73.9	99.0
	Enhcd	GRP	99.2	99.2	99.1	-	-	99.2	-	68.3	99.3	99.2	99.1	98.5	99.2	98.9	97.7	96.8	94.2	96.9	96.9	96.7	50.0	99.1	96.9	98.6	50.0
		ORD	99.1	99.1	99.2	-	-	99.1	-	96.7	99.3	99.3	99.3	98.9	99.3	99.1	98.9	98.5	97.6	98.7	98.6	98.5	89.3	98.9	97.8	99.2	89.3
Two	Basic	GRP	86.9	86.9	86.9	-	-	87.3	-	87.1	87.3	86.9	86.9	86.9	87.3	87.2	64.2	87.3	87.3	87.4	87.3	87.3	87.0	50.0	86.8	54.6	86.9
		ORD	87.6	87.6	87.6	-	-	87.6	-	87.1	87.6	87.6	87.6	87.6	87.6	87.6	77.5	87.6	87.6	87.6	87.6	87.6	87.2	50.0	87.1	55.2	87.6
	Enhcd	GRP	87.4	86.6	87.5	-	-	87.5	-	50.0	87.1	87.9	87.4	87.7	86.8	87.1	86.5	86.7	86.6	86.8	86.9	86.7	87.3	87.0	86.9	87.1	86.7
		ORD	87.6	87.8	87.3	-	-	87.8	-	50.0	87.4	87.8	88.0	88.3	87.5	87.1	88.0	87.8	88.1	88.0	87.6	87.9	87.9	87.1	87.1	87.8	87.6

Legend: The highest accuracy scores achieved by the different categorization methods within each evaluation scenario have been shaded.

Table 4
Confusion matrices across evaluation scenarios.

4(a). Interpretation of the matrix entries

		Predicted Class	
		1	2
True Class	1	True positive (TP)	False negative (FN)
	2	False positive (FP)	True negative (TN)

4(b). Evaluating ten alternatives

Basic				Enhanced			
GRP		ORD		GRP		ORD	
Kernel Naive Bayes		**Fine Tree**		**Fine Tree**		**Subspace KKN**	
1138	862	1985	15	642	1358	1983	17
1084	916	3	1997	583	1417	3	1997

4(c). Evaluating six alternatives

Basic				Enhanced			
GRP		ORD		GRP		ORD	
Fine Tree		**Medium Tree**		**Linear SVM**		**Cubic SVM**	
1863	137	1980	20	1970	30	1980	20
0	2000	20	1980	0	2000	9	1991

4(d). Evaluating two alternatives

Basic				Enhanced			
GRP		ORD		GRP		ORD	
Cosine KNN		**RUSBoosted Trees**		**Quadratic SVM**		**Fine Gaussian SVM**	
1997	3	1998	2	1844	156	1835	165
503	1497	492	1508	329	1671	302	1698

presented in Table 6. Clearly, both retrieval processes are able to mimic the CTR behavior of users observed empirically.

The accuracy scores achieved by the different machine learning techniques together with the corresponding confusion matrices and

Table 5
Pattern recognition capacities across machine learning techniques: enhanced performance analyses.

Alternatives	Two				Six				Ten			
Setting	Basic		Enhanced		Basic		Enhanced		Basic		Enhanced	
Scenario	GRP	ORD	GRP	ORD	GRP	ORD	GRP	ORD	GRP	ORD	GRP	ORD
Most accurate technique (Table 3)	Cosine KNN	RUSBoosted trees	Quadratic SVM	Fine Gaussian SVM	Fine tree	Medium tree	Linear SVM	Cubic SVM	Kernel Naive Bayes	Fine tree	Fine tree	Subspace KKN
Area under ROC curve	0.88	0.96	0.96	0.93	0.97	0.99	1.00	1.00	0.51	1.00	0.49	1.00
F-measure	0.888	0.890	0.884	0.887	0.965	0.990	0.992	0.993	0.539	0.995	0.398	0.995
Kappa statistic	0.747	0.753	0.758	0.767	0.932	0.980	0.985	0.986	0.027	0.991	0.030	0.990
Hyperparameter (HP)	Cheby-shev Metric	Bag Ensemble	Gaussian Kernel	Gaussian Kernel	Maximum Deviance Reduction	Gini's diversity index	Linear Kernel	Gaussian Kernel	Epanech-nikov Kernel	Gini's diversity index	Gini's diversity index	Gentle Boost Ensemble
HP accuracy	87.7%	87.6%	88.5%	88%	96.0%	98.9%	99.2%	99.2%	50.2%	99.4%	50.8%	99.5%
HP training time (s)	38.508	101.31	446.63	168.55	18.581	18.181	188.44	211.95	132.31	18.411	27.517	200.63

Table 6
Empirical CTRs and evaluation processes: The case with ten alternatives.

Ranking position	Average CTR 2016–2020	CTR ordered	CTR complete
1	34.58	34.51	34.57
2	17.21	17.25	17.26
3	10.51	10.51	10.52
4	7.14	7.12	7.17
5	5.11	5.10	5.13
6	3.79	3.78	3.83
7	2.89	2.88	2.86
8	2.27	2.26	2.25
9	1.82	1.83	1.83
10	1.53	1.54	1.54

Table 8
Confusion matrices and empirical CTRs.

Basic				Enhanced			
GRP		ORD		GRP		ORD	
Cubic KNN		**Boosted Trees**		**Bagged Trees**		**Boosted Trees**	
1968	32	781	1219	1046	954	868	1132
1957	43	11	1989	1077	923	123	1877

additional performance tests are summarized in Tables 7, 8 and 9, respectively. As was the case with the results presented in Table 3, a total of 2000 queries have been simulated per retrieval process within the current evaluation scenario. We can observe how feature positioning increases the capacity of these techniques to validate whether DMs

- perform a simple search, based on independent thresholds equating the values of the corresponding CTRs, or
- consider a complete sequential structure, where the realizations observed determine the subsequent retrieval paths,

when retrieving information from the alternatives delivered by a search engine within its first page of results.

We conclude by noting that machine learning techniques can be prevented from distinguishing among DMs by imposing an acceptance threshold sufficiently close to 100%, which generates indistinguishable patterns across the three evaluation processes. That is, the capacity of machine learning techniques to categorize DMs depends on the threshold values defining their retrieval behavior. We have selected a neutral value of $1/2$ within the uniform [0, 1] framework analyzed, though strong biases towards any end of the density domain would affect the categorization abilities of these techniques.

4. Hyperparameter optimization

The selection of the hyperparameters defining each machine learning technique constitutes one of the main determinants of their accuracy. Thus, those parameters that can be modified within each model can be tested to evaluate its performance. The parameters are determined by the type of model being considered within each family of classification techniques described in Tables 3 and 7, namely, decision trees, discriminant analysis, naïve Bayes classifiers, support vector machines (SVM), nearest neighbor and ensemble classifiers. For instance, given the accuracy displayed by the Fine Gaussian SVM in the enhanced ordered scenario with two alternatives, we optimize the hyperparameters within the SVM family of techniques. As illustrated in Table 5, a Gaussian Kernel displays the highest efficiency when optimizing the hyperparameters, leading to an almost identical accuracy as the original classifier.[2]

MATLAB tunes the hyperparameters though Bayesian optimization. The goal of the optimization problem is to find a set of hyperparameter values that minimize the classification error of the model. The program maximizes the expected improvement of the objective function to

[2] A complete description of all the optimizable hyperparameters, their potential values and ranges can be found at https://it.mathworks.com/help/stats/hyperparameter-optimization-in-classification-learner-app.html.

Table 7
Pattern recognition capacities and empirical CTRs.

Evaluation Scenario	Features	Evaluation processes	Tree								Support Vector Machine						K-nearest neighbors						Ensemble				
			Fine	Medium	Coarse	Linear Discriminant	Quadratic Discriminant	Logistic Regression	Gaussian Naïve Bayes	Kernel Naïve Bayes	Linear	Quadratic	Cubic	Fine Gaussian	Medium Gaussian	Coarse Gaussian	Fine	Medium	Coarse	Cosine	Cubic	Weighted	Boosted Trees	Bagged Trees	Subspace Discriminant	Subspace KNN	RUSBoosted Trees
Ten	Basic	GRP	49	49	49.8	-	-	48.8	-	50	49.5	49.7	50.3	48.5	49.9	49.5	50	50.2	49.7	49.9	50.3	50	49	49.3	48.8	50	49.2
		ORD	68.6	69.2	68.8	64.3	-	67.6	-	59.1	67	68.7	65	68.4	68.3	68	67.9	67.8	64.3	67.8	67.8	67.9	69.3	58.2	64.5	54.4	69.3
	Enhcd	GRP	49.6	49.6	50.5	-	-	49.9	-	49.8	50.1	50.1	50.1	49.9	49.3	50.5	49	50.7	50.4	50	50.7	49.9	50.2	50.9	50.1	50	49.6
		ORD	68.8	68.3	68.6	66	-	66.6	-	59.4	67.3	67.7	67.4	63.5	67.5	68.6	65.9	67.6	67.5	68.8	66.7	68.0	68.8	67.8	65.8	68.8	68.3

Legend: The highest accuracy scores achieved by the different categorization methods within each evaluation scenario have been shaded

Table 9
Pattern recognition capacities and empirical CTRs: enhanced performance analyses.

Alternatives	Ten (empirical CTRs)			
Setting	Basic		Enhanced	
Scenario	GRP	ORD	GRP	ORD
Most accurate technique (Table 7)	Cubic KNN	Boosted trees	Bagged trees	Boosted trees
Area under ROC curve	0.50	0.80	0.49	0.79
F-measure	0.664	0.559	0.507	0.580
Kappa statistic	0.006	0.385	−0.016	0.373
Hyperparameter (HP)	Spearman distance	Bag ensemble	Logit boost ensemble	Bag ensemble
HP accuracy	50.0%	69.0%	50.8%	68.7%
HP training time (s)	43.512	122.94	149.58	97.976

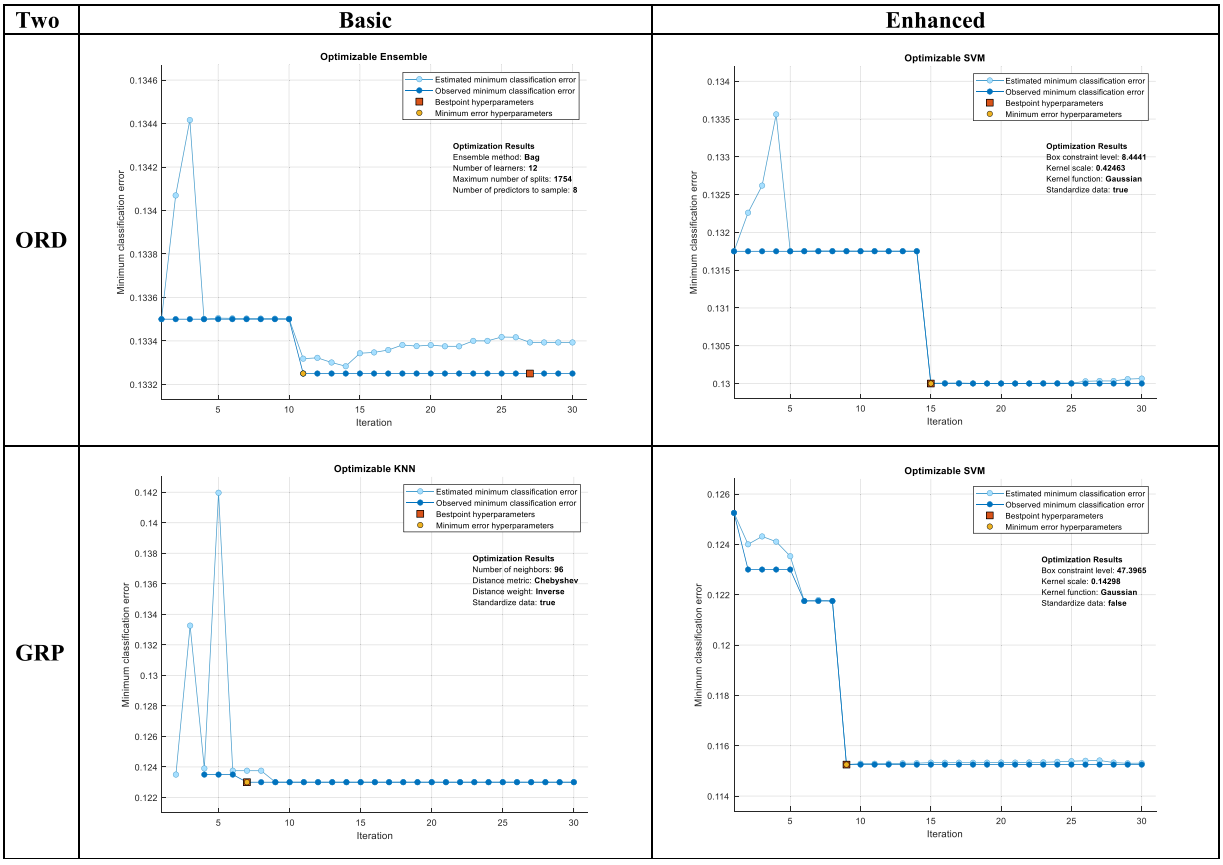

Fig. 9. Hyperparameter optimization and classification errors in evaluation scenarios with two alternatives.

determine the next set of hyperparameter values that will be tried.[3] Each iteration corresponds to a combination of hyperparameter values. The default number of iterations is 30. The optimized hyperparameters for each of the evaluation scenarios analyzed are described in Tables 5 and 9, together with their corresponding accuracies and the training time required to perform the 30 iterations.

As can be observed through Figs. 9 to 12, the tuning process delivers two different sets of hyperparameters, namely, best point and minimum error. This difference follows from the fact that the best points selected do not necessarily deliver the minimum classification error within the 30 iterations performed. Indeed, the program selects the hyperparameter values that minimize an upper confidence interval of the classification error objective model.[4]

We consider the optimization of hyperparameters as a validation scheme and focus on analyzing the error minimization processes defined through the different iterations displayed in Figs. 9 to 12. The analysis allows us to validate the main results described in the previous sections and emphasize the difficulties faced by the different techniques when dealing with the scenarios composed by ten alternatives. Note, in particular, the substantially larger classification errors exhibited by the optimization processes within the GRP scenarios depicted in Figs. 11 and 12 relative to those presented in Figs. 9 and 10. Further technical details describing the optimized hyperparameters are provided within each figure.

[3] The default acquisition function is denoted expected improvement per second plus. Further details describing the optimization problem implemented are given at https://it.mathworks.com/help/stats/bayesian-optimization-algorithm.html#bvbjtxi.

[4] Additional details regarding the definition of the best points selected by the program can be found at https://it.mathworks.com/help/stats/bayesianoptimization.bestpoint.html.

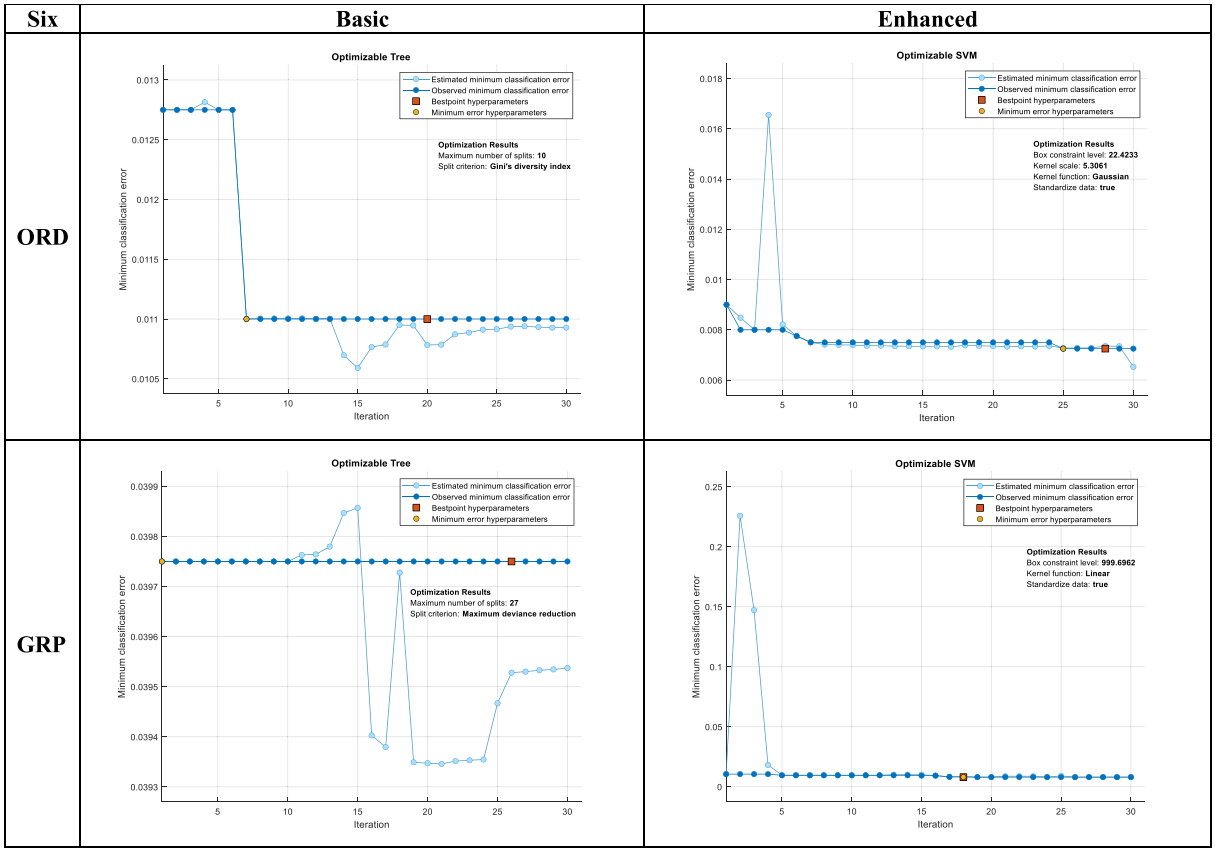

Fig. 10. Hyperparameter optimization and classification errors in evaluation scenarios with six alternatives.

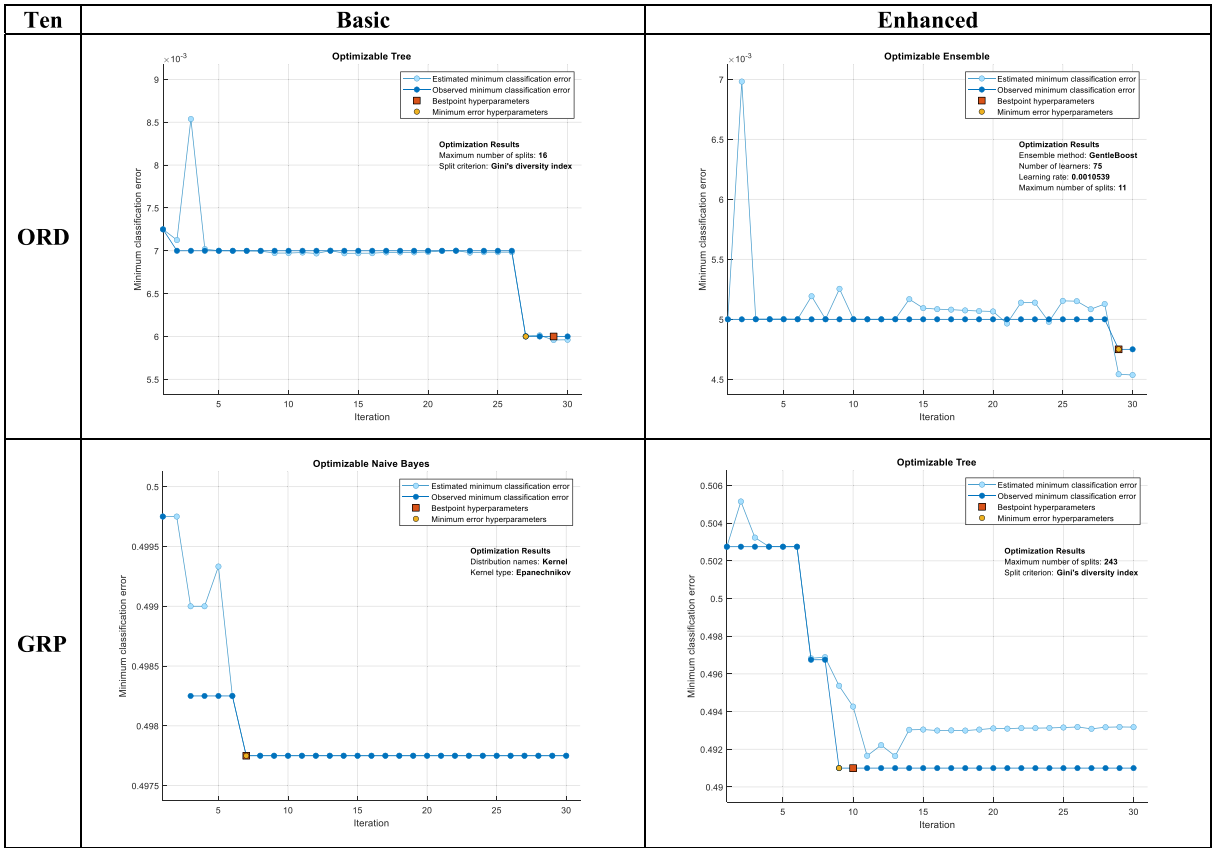

Fig. 11. Hyperparameter optimization and classification errors in evaluation scenarios with ten alternatives.

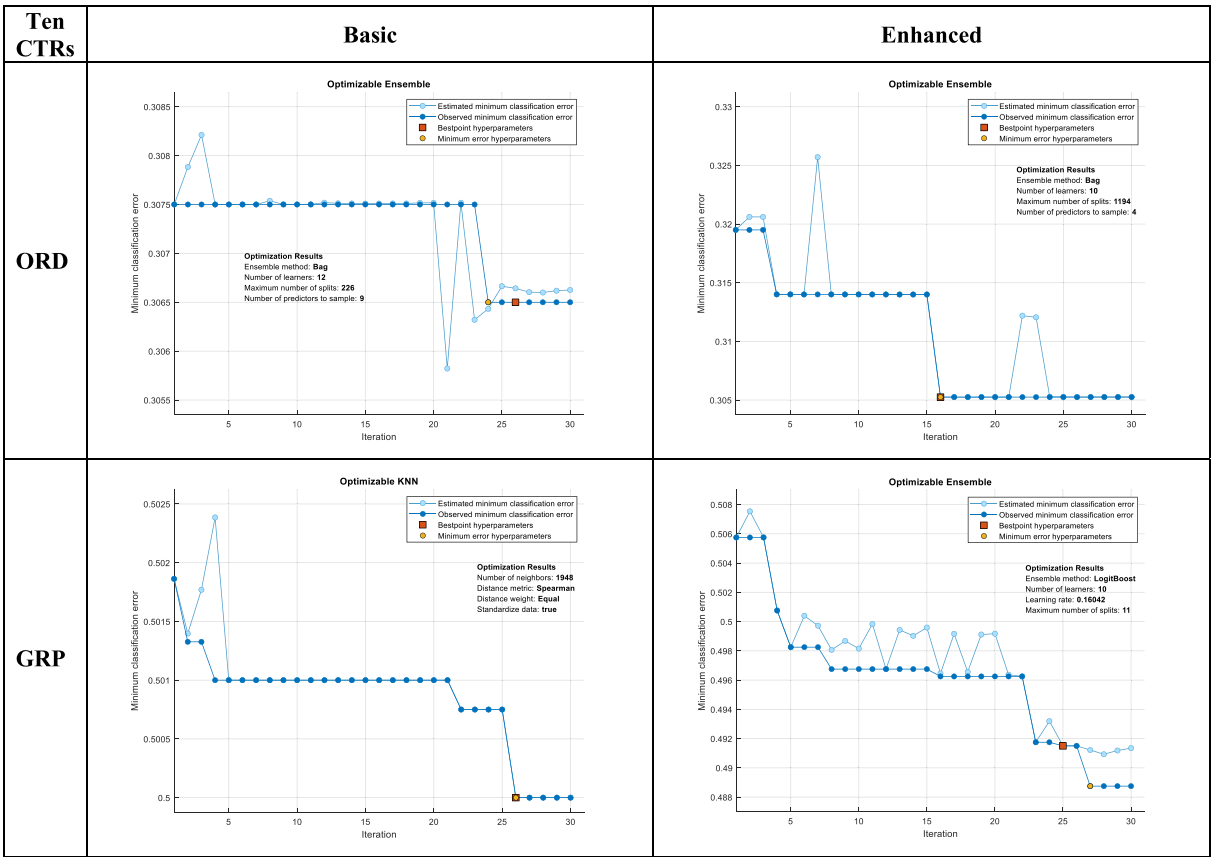

Fig. 12. Hyperparameter optimization and classification errors in evaluation scenarios with ten empirical CTRs alternatives.

Table 10
Statistical analysis of the relationship between evaluation processes.

Scenario	Two		Six		Ten		Ten (empirical CTRs)	
Setting	ORD	GRP	ORD	GRP	ORD	GRP	ORD	GRP
h	1	1	1	1	0	0	0	0
p-value	0	0	0	0	0.8433	0.8771	0.3469	0.8843
t-statistic	49.473	46.713	65.405	64.176	−0.198	0.155	0.941	−0.146
Standard deviation	1.2456	1.3327	3.1723	3.2201	5.2582	3.7499	1.4433	1.3116

Degrees of freedom: 19,999 in all settings.

5. Machine learning versus statistical analyses

We analyze the results obtained when performing a standard statistical t-test to differentiate among evaluation processes. We focus on the relationship existing between the strings of values describing the threshold satisfying alternatives clicked. This is done by merging the lower section of the evaluation vectors into column vectors describing the pages clicked per evaluation process and computing the t-value of the corresponding paired series.

The vectors describing each evaluation process are composed by a total of 20,000 rows. As illustrated in Table 1, each search query is represented by a matrix column composed by ten evaluations, with the resulting pages clicked determining the values inputted in the ten lower rows defining the query. That is, each query is composed by 10 potential clicks. Given the 2000 queries simulated per evaluation process within each scenario, we generate column vectors of 20,000 rows and compute the differences in means using a paired-sample t-test statistic. For instance, the null hypothesis states that the pairwise difference between the ordered and complete evaluation processes follows a normal distribution with zero mean and unknown variance. A value of h = 1 indicates that the t-test rejects the null hypothesis at the 5% significance level.

Table 10 summarizes the main results obtained from these pairwise comparisons. The statistic does not identify a relationship between evaluation processes when considering less than ten alternatives. That is, the statistical test suffices to validate the fact that the strings of clicks follow from different evaluation processes. The variability displayed within the lower section of the evaluation vectors suffices to identify the differences between processes. However, the statistic is unable to validate the fact that the retrieval processes are indeed different when considering ten alternatives. This is the case for all the evaluation scenarios accounting for ten alternatives.

Thus, standard statistics correctly identify the different processes when analyzing relatively simple retrieval scenarios. However, these techniques underperform as the complexity of the evaluation processes increases. On the other hand, machine learning techniques correctly differentiate between the ordered and complete evaluation processes in the scenario with ten alternatives. The capacity of these latter techniques to consider the whole vector of features when categorizing the alternatives confers them with an important advantage over standard statistical analyses, which are highly dependent on the quality of the data defining the corresponding independent variables.

Regarding the immediate applicability of these results, we must highlight the fact that, despite the reticence of physicians, machine

learning techniques remain a consistent reference tool in medical environments (Massie et al., 2020; Siga et al., 2020). The capacity of machine learning techniques to deliver consistent results when dealing with potential data misprints is particularly relevant when considering the coordination problems faced by hospitals in emergency situations or when comparing and merging databases across different sections (Arora, Banerjee, & Narasu, 2020; Rasheed et al., 2020; Vaishya, Javaid, Khan, & Haleem, 2020). This is an important problem among hospitals located in developing countries, where the quality of the data retrieved is generally lower, limiting their identification and extrapolation capacities. Thus, the implementation of machine learning techniques combined, for instance, with mathematical optimization methods can provide a consistent solution to an endemic suboptimal situation affecting institutions facing data quality constraints (Revuelta et al., 2021).

6. Conclusion

Machine learning techniques consider each observation as a set of predictor values ordered within a vector together with the class to which the observation belongs. We have illustrated how these techniques are able to categorize DMs correctly even when failing to convey important information regarding the sequential retrieval behavior characterizing the evaluation processes. In this regard, inputting the pages clicked so as to represent the order of evaluation enhances substantially the categorization capacity of these techniques. This quality constitutes an important advantage over standard statistical methods, particularly when the data is organized incorrectly or sparsely.

We have indeed concluded by performing a statistical analysis of the relationship between the strings of data describing the pages clicked by DMs within each evaluation scenario. We have run t-tests to validate the capacity of standard statistical techniques to recognize whether the clicks are derived from the same type of evaluation process. The results obtained confirmed the intuition developed through the paper. That is, the processes compared within the evaluation scenarios consisting of two and six alternatives lack any relationship at the 5% significance level. On the other hand, when considering the scenario with ten alternatives, the test validated the existence of a relationship between the retrieval processes compared. In other words, the t-test concluded that the ordered evaluation process was equivalent to the complete one, a drawback that machine learning techniques manage to overcome.

Note that these results should not be interpreted as a call to prioritize the implementation of machine learning techniques over standard statistical analyses, but as an illustration of the complementarities existing between both types of methods, particularly in scenarios constrained by the low quality of the data being analyzed.

Declaration of competing interest

The authors declare that they have no known competing financial interests or personal relationships that could have appeared to influence the work reported in this paper.

Appendix A. Supplementary data

Supplementary material related to this article can be found online at https://doi.org/10.1016/j.mlwa.2021.100196.

References

Advanced web ranking: Google organic CTR history. (2021). https://www.advancedwebranking.com/ctrstudy/ (Last accessed September 23rd, 2021).

Arora, N., Banerjee, A. K., & Narasu, M. L. (2020). The role of artificial intelligence in tackling COVID-19. *Future Virology*, *15*, 717–724. http://dx.doi.org/10.2217/fvl-2020-0130.

Bae, S., et al. (2020). Machine learning to predict transplant outcomes: Helpful or hype? A national cohort study. *Transplant International*, *33*, 1472–1480. http://dx.doi.org/10.1111/tri.13695.

Basu, S. (2018). Information search in the internet markets: Experience versus search goods. *Electronic Commerce Research and Applications*, *30*, 25–37. http://dx.doi.org/10.1016/j.elerap.2018.05.004.

(2013). Chitika: the value of google result positioning. In *Chitika Insights June 7, 2013*. Westborough: Chitika, Available at perma.cc/7AGC-HTDH.

Dean, B. (2019). We analyzed 5 million google search results. Here's what we learned about organic click through rate. Available at https://backlinko.com/google-ctr-stats.

Epstein, R., & Robertson, R. E. (2015). The search engine manipulation effect (SEME) and its possible impact on the outcomes of elections. *Proceedings of the National Academy of Sciences of the United States of America*, *112*, E4512–E4521. http://dx.doi.org/10.1073/pnas.1419828112.

Lancet (2021). Editorial. Artificial intelligence for COVID-19: Saviour or saboteur? *Lancet Digital Health*, http://dx.doi.org/10.1016/S2589-7500(20)30295-8.

Lewandowski, D., & Kammerer, Y. (2020). Factors influencing viewing behaviour on search engine results pages: A review of eye-tracking research. *Behaviour & Information Technology*, http://dx.doi.org/10.1080/0144929X.2020.1761450.

Li, H., Duan, H., Zheng, Y., Wang, Q., & Wang, Y. (2020). A CTR prediction model based on user interest via attention mechanism. *Applied Intelligence: The International Journal of Artificial Intelligence, Neural Networks, and Complex Problem-Solving Technologies*, *50*, 1192–1203. http://dx.doi.org/10.1007/s10489-019-01571-9.

Massie, A. B., et al. (2020). Identifying scenarios of benefit or harm from kidney transplantation during the COVID-19 pandemic: A stochastic simulation and machine learning study. *American Journal of Transplantation*, *20*, 2997–3007. http://dx.doi.org/10.1111/ajt.16117.

Pan, B., Hembrooke, H., Joachims, T., Lorigo, L., Gay, G., & Granka, L. (2007). In google we trust: Users' decisions on rank, position, and relevance. *Journal of Computer-Mediated Communication*, *12*, 801–823. http://dx.doi.org/10.1111/j.1083-6101.2007.00351.x.

Rasheed, J., et al. (2020). A survey on artificial intelligence approaches in supporting frontline workers and decision makers for the COVID-19 pandemic. *Chaos, Solitons & Fractals*, *141*, Article 110337. http://dx.doi.org/10.1016/j.chaos.2020.110337.

Revuelta, I., et al. (2021). A hybrid data envelopment analysis–artificial neural network prediction model for COVID-19 severity in transplant recipients. *Artificial Intelligence Review*, http://dx.doi.org/10.1007/s10462-021-10008-0.

Schwartz, B. (2015). What does it mean to be a rational decision maker? *Journal of Marketing Behavior*, *1*, 113–145. http://dx.doi.org/10.1561/107.00000007.

Siga, M. M., et al. (2020). Prediction of all-cause mortality in haemodialysis patients using a Bayesian network. *Nephrology Dialysis Transplantation*, *35*, 1420–1425. http://dx.doi.org/10.1093/ndt/gfz295.

Tavana, M., Caprio, D. Di., & Santos-Arteaga, F. J. (2017). A multi-criteria perception-based strict-ordering algorithm for identifying the most-preferred choice among equally-evaluated alternatives. *Information Sciences*, *381*, 322–340. http://dx.doi.org/10.1016/j.ins.2016.11.021.

Vaishya, R., Javaid, M., Khan, I. H., & Haleem, A. (2020). Artificial intelligence (AI) applications for COVID-19 pandemic. *Diabetes and Metabolic Syndrome: Clinical Research and Reviews*, *14*, 337–339. http://dx.doi.org/10.1016/j.dsx.2020.04.012.

Victorelli, E. Z., Dos Reis, J. C., Hornung, H., & Prado, A. B. (2020). Understanding human-data interaction: Literature review and recommendations for design. *International Journal of Human-Computer Studies*, *134*, 13–32. http://dx.doi.org/10.1016/j.ijhcs.2019.09.004.

Wynants, L., et al. (2020). Prediction models for diagnosis and prognosis of covid-19: Systematic review and critical appraisal. *BMJ*, *369*, m1328. http://dx.doi.org/10.1136/bmj.m1328.