Journal of Marketing Analytics
https://doi.org/10.1057/s41270-023-00217-7

ORIGINAL ARTICLE

Machine learning for product choice prediction

Josué Martínez-Garmendia[1,2]

Revised: 16 October 2022 / Accepted: 24 February 2023

Abstract
The goal of this paper is to provide a point of empirical evidence as to how machine-learning techniques stack-up in their ability to predict consumer choices relative to traditional statistical techniques. We compare a traditional (naïve) multinomial logit to six machine-learning alternatives: learning multinomial logit, random forests, neural networks, gradient boosting, support vector machines and an ensemble learning algorithm. The comparison is done by applying these methods to beer category stock keeping unit (SKU) level panel data. Results show that machine-learning techniques tend to perform better, but not always. Ensemble learning performs best while maintaining an overall high-performance level across all SKU classes, independently of their sample size. This result builds on existing evidence about the benefits of combining multiple prediction techniques over relying on a single best performing model, as conventional wisdom would intuitively make us believe. In general, the better performance of machine learning techniques at predicting product choice should not come as a surprise. At their core, machine learning techniques are designed to augment dimensionality of models and/or scan through orders of magnitude greater model alternatives, relative to the narrower focus of traditional approaches.

Keywords Choice prediction · Machine learning · Ensemble learning

Introduction

Machine-learning techniques have been gaining ground over traditional statistics in most fields, particularly among practitioners. Social sciences including marketing and economics are no exception. However, the quantitative foundations of these two fields have historically been rooted in interpretability and causal inference, while machine learning is generally understood to be stronger at predicting but relatively weaker at interpretability and hypothesis testing. This contrast has sparked some intellectually stimulating philosophical soul-searching in these two disciplines (Iskhakov et al. 2020; Dzyabura and Yoganarasimhan 2018). Academic marketing research based on machine learning techniques is nothing new (Hauser et al. 2010; Kazemia et al. 2013; Huang and Luo 2015; Liu and Dzyabura 2016; Yoganarasimhan 2020), but it is somewhat surprisingly limited in terms of assessments of this methodological disruption to the discipline. The contribution of this paper is to provide a point of comparison across a wide set of machine-learning techniques in their ability to predict consumer choices relative to more traditional statistical techniques and uncovering their performance strengths and weaknesses in the context of a typical consumer product choice problem.

The greatest conceptual difference between traditional statistical techniques and machine learning is that in the former a hypothesis is proposed in the form of a stylized mathematical model that is then tested; while the latter allows for greater mathematical construct flexibility, without explicit regard for theoretical foundations, as it tries to maximize predictive accuracy. One can also argue that from a procedural standpoint, there are five relatively common elements of machine learning that distinguish it from traditional statistics in their systematic application: (1) data preprocessing, (2) feature engineering, (3) data splitting between training and test sets, (4) cross-validation, and (5) use of tuning parameters (Kuhn and Johnson 2013). This is not to say that traditional statistics has not applied any of these steps, generally speaking; but perhaps, that they have been less central and widespread in their application.

✉ Josué Martínez-Garmendia
pm3100@columbia.edu

[1] School of Professional Studies, Columbia University, New York, USA

[2] School of Information, University of California Berkeley, Berkeley, CA, USA

Published online: 04 April 2023

Data preprocessing tends to focus on, among other transformations, the centering and scaling of predictor features by subtracting their mean and dividing them by their dispersion statistic, often the standard deviation. Removal of skewness is also often applied to the data by replacing raw input feature values with log, square root, inverse, or Box-Cox transformations. The treatment of missing values is also confronted head-on with imputation techniques that quite often lean on other predictive machine learning techniques, like k-nearest-neighbors. One of the main reasons behind these manipulations of the raw data is to contribute to the smoothness, speed, and numerical stability of the high computational demands associated with machine-learning parameter estimation.

Feature engineering is also not unique to machine learning, but ubiquitous relative to traditional statistics. In fact, techniques like principal components or factor analysis have been commonplace in applied statistics as ways to reduce dimensionality of models, resulting in among other benefits lower multicollinearity and overfitting. However, machine learning has taken this further with its foundational reliance on non-linear transformations like polynomials, hinge functions and kernels. These approaches often seek not to reduce dimensionality, but rather grow it (sometimes in ways that generate negative degrees of freedom) to capture nuances that improve a model's predictive performance. Often, multiple versions of engineered features are tested in the model iteratively, leveraging cross-validation (discussed below).

Splitting the data into training and test sets relies on random (and in some cases stratified, to maintain proportional distribution of feature values) sampling of the original data set. Its intent is primarily to evaluate a model on data it has not seen before, as a true gauge of a model's predictive ability. One can understand that this is important to machine learning since its application is heavily oriented towards actionable prediction, particularly amongst practitioners. In traditional statistics or econometrics, models are often evaluated on the full dataset, and less frequently on an out-of-sample set aside from the full dataset.

Cross-validation is perhaps one of the most salient departures of machine-learning techniques. The training dataset itself is split into multiple training subsets. This is, therefore, analogous to splitting the original full dataset into training and test sets. But a key difference is its reliance on resampling techniques rooted in bootstrapping (Efron and Tibshirani 1993). It involves randomly dividing the training set observations with resampling into two parts: a training subset and a validation set (more commonly, multiple validation sets). The model is fit on the training subset and the fitted model is then used to predict responses for the observations in the validation set. Resampling, refitting and evaluation of the model are done multiple times. Two common resampling techniques used for cross-validation are leave-one-out and k-fold. A key benefit of cross-validation is the estimation of how sensitive a model is to small changes in input data. Cross-validation results are often summarized with the mean or some other form of aggregation of the multiple model performance scores. Another application of cross-validation is to compare the performance of multiple somewhat different versions of a model to select the best performing one. This is related to the next key element of machine learning: tuning parameters.

Tuning parameters (also called hyperparameters) are fundamental to machine learning, and a significant departure relative to traditional statistics. Tuning parameters shape model and/or loss/objective functions to select model parameter values. They typically do not have analytical solutions, and instead rely on numerical search algorithms. Thus, tuning parameters require an iterative trial-and-error approach or numerical optimization algorithm across a preselected number or range of values to evaluate. As described above, this is done through cross-validation. Cross-validation to find optimal tuning parameter values is executed as follows:

- Step 1: Identify number of alternative models to run, based on tuning parameters to be tested.
- Step 2: For each model alternative

  o Step 2.1: Split the training data into a training subset and one or a number of validation-sets.
  p Step 2.2: Fit the model to the training subset data.
  q Step 2.3: Estimate performance of model against validation sets.

- Step 3: After all model alternatives have been run, aggregate, and contrast results of all model alternatives.
- Step 4: Select the best performing model, as shaped by a specific set of potential tuning parameters.
- Step 5: Re-fit the best performing model on the entire training dataset to update model parameters.

This paper's objective is to determine whether machine learning models are superior to traditional statistical models at predicting consumer choice. More specifically it compares beer SKU consumer choice predictive ability of a traditional (called naïve moving forward) multinomial logit model against six of its relatively more relevant machine-learning classification alternatives, in a realistic setting. The six alternatives are a machine-learning-based multinomial logit model (called learning multinomial logit moving forward), random forests, gradient boosting, neural networks, support vector machines and a learning ensemble method. The paper does not go in-depth into each technique and rather limits itself on describing the functional forms relative to the problem at hand, the objective/loss function, and tuning parameters to maximize model predictive performance.

There are better sources for a comprehensive treatise of the approaches leveraged in this paper (Hastie et al. 2009). The data used for the analyses is grocery scanner panel data for beer category purchases from Information Resources, Inc.'s academic dataset (Bronnenberg et al. 2008). The metrics applied to determine performance of the different models are Sensitivity, Accuracy and Cohen's Kappa. These three-performance metrics are some of the most commonly used ones for classification problems, such as brand or product choice prediction.

Data

In order for models to capture product and individual heterogeneity, this study combines panel and store scanner data for beer category purchases, similar to Chintagunta and Dubé (2005), from the U.S. by Information Resources, Inc. (Bronnenberg et al. 2008). This dataset is offered and maintained by the marketing research firm Information Resources (IRI), in the U.S. It is intended to enable academic researchers with the ability to study important topics in marketing and economics that are of concern to practitioners, policy-makers, and scholars. Well-recognized in the consumer-packaged goods space amongst academicians, it has broadly been applied to marketing research. Twelve years of weekly store-level data (2001 to 2012) for chain grocery and drug stores in 47 markets is provided. It also offers panel purchase and demographic data for two of IRI's BehaviorScan markets (11 years, for the same categories). While not used in this paper, TNS advertising data for two categories for some early years is also provided. For the purpose of this paper, three full years of data from 2001 to 2003 are used. However, the first year is used to provide historical values used as predictors for 2002. Thus, for the modeling exercise, a third of the observations are lost to create lagged variables for observations in years 2002 and 2003. As a result, the modeling relies on 35,391 observations, with no missing values. The dataset contains 149 beer category SKUs, ranging from 40 to 1,548 observations for each class value, and an average of 237.5. The dataset was constrained to only contain SKUs with at least 40 observations. SKUs are coded as SKU1-SKU149, rather than by product name. The models try to forecast a panelist's next purchase based on a number of predictor features, including prior purchase choices. Predictor features include the last choice made by a panelist in terms of brand, volume content, number of units in the package, and package type (Table 1). These are all treated as class features, and their descriptive statistics are shown in Table 1. Last brand purchased has a total of 62 levels, with at least 51 observations each. Last volume content purchased has three levels (96, 144 and 196 oz), with a minimum of 263 observations each. Last package units purchased has also three levels (6, 20 and 35 units), with a minimum of 4,934 observations. Last package type purchased has only two levels (can or glass). Table 2 contains two additional continuous predictor features: number of category purchases made in the prior year and number of days since last purchase. The minimum value for number of category purchases made in the prior year is zero days, the maximum is 119 days, and the average is 18.9 days. For number of days since last purchase, the minimum observed value is zero days, the maximum 151 and the average 7.0.

A third set of predictor features are demographic characteristics of panelists (Table 3). In the dataset, there are 3,404 unique panelists. The panelist observational frequencies range from one to 243 observations. The demographic features included are income, family size, residence type, age and education. All of them are treated as class features. For income, we have three levels: less than $35,000, $35,000

Table 2 Consumption habits: continuous features

Feature	Description	Observations	Minimum value	Maximum value	average value
PYCAT	Number of purchases by panelist in prior year	35,391	0	119	18.9
LP	Number of days since last purchase	35,391	0	151	7.0

Table 1 Consumption habits: class features

Feature	Description	Observations	Number of classes	Minimum class observations	Maximum class observations
SKU	SKU classes	35,391	149	40	1,548
LBR	Brand last consumed	35,391	62	51	3,894
LOZ	Total SKU volume in ounces	35,391	3	263	30,191
LCT	Number of units in SKU	35,391	3	4,934	15,248
LPK	Package type (can or glass)	35,391	2	10,529	24,863

Table 3 Panelist demographic features

Feature	Description	Observations	Number of classes	Minimum class frequencies (%)	Maximum class frequencies
IN	Income level	35,391	3	27.5	39.6
FS	Family size	35,391	3	12.1	44.9
RE	Residence type	35,391	2	17.0	83.0
AGE	Age group	35,391	3	15.5	60.5
EDU	Education level	35,391	3	28.4	40.8

to $65,000 and more than $65,000. The level with the smallest sample representation is more than $65,000 with a total of 27.5% of the observations. Family size has also three levels: one, two and greater than three. Of these three levels, the weakest representation is one, with 12.1% of the observations. Residence has two levels: owner and renter. The renters account for 17.0% of the observations in the dataset, while the rest are owners. Age has three levels: less than 45, between 45 and 54, and older than 54. The class with the least representation in the dataset is less than 45 years old with 15.5% of the observations. Finally, education has also three levels: basic, high-school, and at least some college. All are evenly distributed, with at least some college having the fewest observations (28.4%).

The fourth and final set of features that the models will leverage for predicting next SKU purchase are marketing variables (Table 4). Specifically, whether the SKU is only featured on the retailer circular, only on a prominent display in the store, both and whether the SKU is on price promotion. These four features are treated as dummies, and they all have zero or one values. In the case of feature only, there are 1,054 observations with value equal to one, while display only, feature-and-display and price reduction take the value one in 5,224, 2,280 and 4,089 occasions, respectively.

For all but the naïve multinomial logit model, data is preprocessed by centering and scaling each continuous feature. This is done as a way to free this benchmark model from one of the key systematic data treatments so commonly applied in machine learning, but less so in traditional modeling. As mentioned before, there are many occasions when researchers and practitioners apply preprocessing to the data before running traditional multinomial models. Centering is achieved by subtracting the mean from each observation, while scaling relies on dividing each observation by the standard deviation of the feature values. The data is then randomly split into training (80%) and test (20%) datasets, while preserving the proportion of the product categories in the full dataset (thus, this is a stratified random split). The 80/20 split while somewhat arbitrary is considered a good rule of thumb by Kuhn and Johnson (2013). Predictions on the test dataset are used as the ultimate arbiter of model performance. With the exception of the naïve multinomial logit model (the baseline comparison or proxy for traditional statistics), training data is split again (this time with resampling) into a training and tenfold cross validation datasets, following best practices (Kuhn and Johnson 2013). Estimation of feature and tuning parameters is based on tenfold cross validation of the training dataset. The cross validation selects parameters that maximize the model with the greatest level of Accuracy (% of times the model predicts product choice observations accurately) against the training validation samples.

Models

The conceptual problem of predicting a consumer's brand (that can be extended to SKU) choice can be framed in the form of a classification model (Russell 2014). Approaches supporting this type of models first predict the probability of each of the categories of a qualitative response feature, Y_i, as the basis for making the classification. In its simplest form

Table 4 Store marketing features

Feature	Description	Observations	Number of classes	Observations with Dummy = 1
FO	Circular feature-ad only	35,391	2	1,054
DO	Display only	35,391	2	5,224
FAD	Feature-ad and display	35,391	2	2,280
PR	Price reduction	35,391	2	4,089

$$p\left(Y_i = k\right) = f\left(X_i\right), i = 1, \ldots, N \tag{1}$$

expresses the probability that the i-th choice falls in the k = 1,…,K k^{th} category, where X is a set of predictor features, and i = 1,…,N is an index of all N observations in the dataset. In our case, Y ∈ {SKU1, …, SKU149}. X is composed of consumption habit, demographic, and marketing predictor features. Consumption habits include last choices made for four class features: brand choice made, volume content, number of units in the package, and package type. Consumption habits in X also include number of category purchases made in the last year and number of days since the last category purchase. Panelist demographic characteristics included in X are class features income, family size, residence, age and education. The last set of features included in X are store marketing conditions: whether the SKU is only featured in the retailer circular, only on a prominent display in the store, both and whether the SKU is on price reduction.

The function f(X) can take on a number of forms depending on the approach selected. In addition, for the machine-learning approaches, the final model can include the full set, just a subset, or an augmented set of the initial X predictor elements through feature engineering. Further, X can be transformed linearly or non-linearly in a number of ways. What follows is seven different ways of turning the conceptual abstraction of Eq. (1) into concrete model specifications and parameter estimations. The description of the approaches below focuses on defining the response model f(X) and the objective or loss function optimized to estimate model parameters, laying out the iterative processes to evaluate parameters and the tuning parameters involved.

Naïve multinomial logit

The model is expressed as a log-linear functional form between the response and the predictor features, for ease of use (Venables and Ripley 2002). To make sure that the probabilities add up to one, *f(X)* is transformed by a softmax function (Bridle 1990):

$$g_k\left(f\left(X_i\right)\right) = \frac{e^{f\left(X_i\right)_k}}{\sum_{l=1}^{K} e^{f\left(X_i\right)_l}} \tag{2}$$

The parameters of this model are estimated by minimizing the deviance function below:

$$-\sum_{n=1}^{N}\sum_{k=1}^{K} Y_{ik} log\left(f_k\left(X_i\right)\right) \tag{3}$$

The corresponding classifier (that turns probabilities into classes) is $G(x) = argmax_k\, f_k(X_i)$ (Hastie et al. 2009). This is a loss function without regularization, and therefore, without tuning parameters, of the type that is more commonplace in machine-learning rather than in traditional statistics (as this multinomial model is trying to represent). Parameters are estimated using maximum likelihood. Throughout the paper, we treat the performance of this model and its estimation procedure as the benchmark the rest of the models and their estimation approaches are compared against.

Learning multinomial logit

In this version of the model, the basic multinomial logit model leverages data preprocessing by scaling and centering it to improve estimation process smoothness. It also relies on regularization to prevent overfitting by adding a decay tuning parameter (λ) to the deviance function in Eq. (3):

$$\sum_{n=1}^{N}\sum_{k=1}^{K} Y_{ik} log\left(f_k\left(X_i\right)\right) + \lambda \sum_{K=1}^{K} \beta_k^2 \tag{4}$$

λ is the only tuning parameter in this algorithm that is estimated using standard tenfold cross-validation (Kuhn and Johnson 2013), thus incorporating an additional machine-learning element to the estimation. The parameters are estimated using maximum likelihood.

Random forests

Random forests (Breiman 2001) are a derivation of bagging, which in turn is a bootstrap version of the classification and regression tree (CART). Random forests' advantage over bagging is that they rely on less correlated individual classification trees. Each time a split is considered, a random sample of *mtry < rank(X)* predictor features is selected as split candidates from the full set of predictors (James et al. 2013). We can express the random forest model as

$$f(X) = \sum_{b=1}^{B} f_b\left(X;\beta_b\right) \tag{5}$$

where and *b = 1, …, B* is the index for number of random trees to estimate, *B*. In practice, each tree produces a class prediction for each observation $\hat{D}_b(X)$. Then the final classification is solved by majority vote across all *b = 1, …, B* trees

$$\hat{D}^B(x) = majorityvote\left\{\hat{D}_b(x)\right\}_1^B \tag{6}$$

Each individual tree leverages a traditional classification tree approach, using the Gini criterion to guide the splitting by Breiman et al. (1984), as in

$$\hat{p}_{nk} = \frac{1}{N}\sum_{x_i \epsilon R_n} I\left(y_i = k\right) \tag{7}$$

where $\widehat{p}_{nk}$ is the proportion of class k observations in node n, and where $I(y_i = k)$ is an indicator conditional function that takes a unit value if the classifier, k, holds and zero if it does not. Observations in each node n are assigned to class $k(n) = \text{arg}max_k \widehat{p}_{nk}$, the majority class in node n, using a greedy algorithm (i.e., making the locally optimal choice at each stage). The distribution of observations minimizing the Gini Index, GI, is:

$$GI = \sum_{k=1}^{K} \widehat{p}_{nk}\left(1 - \widehat{p}_{\text{nk}}\right) \tag{8}$$

Cross validation is leveraged again to estimate tuning parameters $mtry$ and B.

Gradient boosting

Gradient boosting relies on weak classifiers that build on each other to produce an ensemble classifier with a lower predictive error (Kuhn and Johnson 2013). Based on Friedman (2001), the approach initializes a tree to $f_{ko}(X)=0,\ k=1,\ldots,\ K$. Then, it iterates $z=1,\ldots,Z$ times to generate trees, $f_{kz}(X)$, that build on each other at every z step, estimating the probability of an observation belonging to a class:

$$p_k(X) = \frac{e^{f_k(X)}}{\sum_{q=1}^{K} e^{f_q(X)}}, \quad k = 1, \ldots, K \tag{9}$$

This ratio is taken to ensure that probabilities for all classes add up to one. $f(X)$ defines a tree with a constant value for each terminal node. To estimate the parameters of $f_k(x)$ and $f_h(x)$, the algorithm uses the following deviance loss function:

$$L(y, p(x)) = -\sum_{k=1}^{K} I\left(y = G_k\right) log\left(p_k(x)\right) \tag{10}$$

where $I(y = G_k)$ is an indicator conditional function that takes a unit value if the classifier, G_k, holds and zero if it does not. The classifier assigns classes based on what class k gets the greatest $p_k(X)$ value. An interesting aspect of gradient boosting is that during each iteration, z, the error is estimated as $r_{ikz} = Y_{ik}\text{-}p_k(X_i),\ i = 1,..,N$. The value of Y_{ik} takes a unit value if it belongs to class k, and zero otherwise. Then, a regression tree using the X_i features is fit to the errors r_{ikz}, giving terminal regions R_{vkz}, with $v = 1,\ldots V_m$ terminal nodes. R_{vkz} is the set of X values that define each terminal node. From here, Friedman (2001) computes:

$$\gamma_{vkz} = \frac{K-1}{K} \frac{\sum_{x_i \in R_{vkz}} r_{ikz}}{\sum_{x_i \in R_{vkz}} \left|r_{ikz}\right|\left(1 - \left|r_{ikz}\right|\right)} \tag{11}$$

This parameter is used to update the $f_k(X)$ of the z iteration, such as

$$f_{kz}(X) = f_{k,z-1}(X) + \sum_{v=1}^{V_m} \gamma_{vkz} I\left(X \in R_{jkz}\right) \tag{12}$$

One can think of $f_{kZ}(X)$ as the final gradient boost tree solution, $\widehat{f}_k(X)$.

Four tuning parameters are used in this process: number of boosting iterations, Z, maximum weak learner tree depth, d, shrinkage, or speed of learning, γ and minimum terminal node sample size, u. The tuning parameters estimated via cross-validation are Z, d, s, and u.

Neural networks

Neural networks is a multi-stage classification (in our case) approach. The relationship between the response and predictor features in the data is intermediated by features engineered by the approach, called hidden units, H. A hidden unit in a single layer neural network can be described as (Hastie et al. 2009)

$$H_e = \sigma\left(\alpha_{0e} + \alpha_e^T X\right), e = 1, \ldots, E \tag{13}$$

where $e = 1,\ldots,E$ are the number of hidden units in the hidden layer, and $\sigma(X)$ is a function of the linear component $\alpha_{0e} + \alpha_e^T X$ that allows for non-linearities. Then, a linear combination of the engineered features, T_k, is generated for each response variable class:

$$T_k = \beta_{0k} + \beta_k^T H \tag{14}$$

A final transformation step estimates the response function. In a multinomial case like the one in this paper, the function is akin to the multinomial logit, here called a softmax function (Bridle 1990), identical to the one seen in the multinomial model above in Eq. (4):

$$\widehat{f}_k(X) = g_k(T) \tag{15}$$

In addition, following the same algorithm as in the one used in the multinomial logit model, the objective function to be minimized yielding estimates for all parameters (α_{0m}, α_{m}, β_{0k} and β_{k}) is a deviance loss function:

$$R(\theta) = -\sum_{i=1}^{N} \sum_{k=1}^{K} y_{ik} log\left(f_k\left(X_i\right)\right) \tag{16}$$

And the corresponding classifier is $G(X) = argmax_k f_k(X)$. Similar to the one used in the learning multinomial logit model, the loss function in this case allows for regularization, in the form of a weight decay, λ, that penalizes inclusion of additional parameters ($\Theta \in \{\alpha_{\text{0m}}, \alpha_{\text{m}}, \beta_{\text{0k}}$ and $\beta_{\text{k}}\}$) in the model, thus reducing the chance for overfitting

$$R^{*}(\theta) = -\sum_{i=1}^{N}\sum_{k=1}^{K} y_{ik} log\left(f_k\left(X_i\right)\right) + \lambda \sum_{i=1}^{N} \theta_i^2 \tag{17}$$

Cross-validation is used to zero into the two tuning parameters involved in the algorithm, H and λ.

Support vector machines

Support vector machines (SVM) are an extension of maximal margin classifiers and support vector classifiers to accommodate non-linear separation boundaries, also expanding the feature space beyond a two-class setting, using kernels. Kernels, in this context, are inner product functions of predictor features, often represented as $K(x,x') = \langle x_i, x_{i'} \rangle$, capturing the similarity of two observations. Kernels facilitate a computational approach that allow for inclusion of large numbers of features, most of them engineered.

This form of hinge functions, $h(X)$, can be characterized by nonlinearities often in the form of polynomials, radial and neural network functions. The model becomes:

$$\hat{f}(X) = \beta_0 + \beta h(X)^T \tag{18}$$

The goal is to find parameter values that maximize the margin, π, which in SVM parlance is the distance between the classification separation boundary and the closest data point. This can be captured as a constrained optimization of the form:

$$\max_{\beta_0,\beta} \pi s.t. y_i\left(h\left(x_i\right)^T \beta + \beta_0\right) \geq \pi - \xi_i, \xi_i \geq 0, \sum \xi_i \leq C, i = 1, \dots N \tag{19}$$

where ξ is a slack factor to allow some error in the classification preventing overfitting and C is an arbitrary constant. This optimization problem can be turned into a more tractable convex optimization expression (Hastie et al. 2009)

$$\begin{aligned} &\min_{\beta_0,\beta} \|\beta\| s.t. y_i\left(h\left(x_i\right)^T \beta + \beta_0\right) \geq 1 \\ &- \xi_i, \xi_i \geq 0, \sum \xi_i \leq C, \quad i = 1, \dots N \end{aligned} \tag{20}$$

by making $\pi = \frac{1}{\|\beta\|}$. It can be taken a step further and turn the expression into

$$\begin{aligned} &\min_{\beta_0,\beta} \frac{1}{2} \|\beta\|^2 + C \sum_{i=1}^{N} \xi_i s.t. y_i \\ &\left(h\left(x_i\right)^T \beta + \beta_0\right) \geq 1 - \xi_i, \quad i = 1, \dots N \end{aligned} \tag{21}$$

The arbitrary constant C is treated as a tuning parameter. The constrained optimization can be solved using a Lagrangian

$$\begin{aligned} \min_{\beta_0,\beta,\xi} L_P &= \frac{1}{2} \|\beta\|^2 + C \sum_{i=1}^{N} \xi_i - \sum_{i=1}^{N} \alpha_i \\ &\left[y_i\left(h\left(x_i\right)^T \beta + \beta_0\right) - \left(1 - \xi_i\right)\right] - \sum_{i=1}^{N} \mu_i \xi_i \end{aligned} \tag{22}$$

where α_i and μ are Lagrangian multipliers. By setting the first derivatives to zero, analytic expressions of the parameters are generated:

$$\beta = \sum_{i=1}^{N} \alpha_i y_i x_i \tag{23}$$

$$0 = \sum_{i=1}^{N} \alpha_i y_i \tag{24}$$

$$\alpha_i = C - \mu_i \tag{25}$$

The approach then substitutes these first-order solutions into the Lagrangian above, to get the optimization for the Lagrangian dual objective function:

$$\begin{aligned} &\max_{\beta} L_D = \sum_{i=1}^{N} \alpha_i - \frac{1}{2} \sum_{i=1}^{N} \sum_{i'=1}^{N} \alpha_i \alpha_{i'} y_i y_{i'} h\left(X_i\right)^T h\left(X_{i'}\right), \\ &s.t. 0 \leq \alpha_i \leq C and \sum_{i=1}^{N} \alpha_{i'} y_i = 0 \end{aligned} \tag{26}$$

leading to

$$\hat{\beta} = \sum_{i \in S} \hat{\alpha}_i y_i x_i \tag{27}$$

where S is the collection of indices for which α is equal to zero. α is different from zero for only X values in the training set that are closest to the boundary and are predicted with the least amount of certainty. This reduces the feature space for which to estimate parameters more manageable from a computational standpoint.

Ensemble learning

Ensemble learning is becoming a widely used approach in machine learning (Bajari et al. 2015; Sagi and Rokach 2018). It can take a range of forms but at its core it is built by combining a collection of underlying models. Some of the machine-learning models applied in this paper are themselves ensemble models, like random forests and gradient boosting. That is, their final estimate is built from weaker models developed during the estimation process. For the ensemble learning model, this paper uses an approach similar to random forests. It takes estimates the predictions of all machine-learning models on the test observations, and applies a majority vote rule approach:

$$\widehat{SKU}_{i,Ensemble} = majority\ vote\left\{\widehat{SKU}_{i,Naive\ Multinomial}, \widehat{SKU}_{i,Learning\ Multinomial}, \widehat{SKU}_{i,Random\ Forests}, \widehat{SKU}_{i,Neural\ Network}, \widehat{SKU}_{i,Gradient\ Boosting}, \widehat{SKU}_{i,Support\ Vector\ Machine}\right\}$$

When ties occur, the best performing model is used for the final prediction.

Results

The objective of this paper is to provide a comparison point as to how machine-learning techniques stack-up in their ability to predict consumer SKU choices relative to each other and a more traditional statistical technique. While classification models produce both continuous probability estimates and the associated predicted class, the focus is typically the latter. That makes sense since it tends to be the observable event. As mentioned earlier, the performance of the classification forecast is done out-of-sample, rather than the in-sample. There are several metrics used to the determine class forecast performance on test observations. We rely on three commonly used ones including Accuracy (percentage of observations correctly classified), Kappa (similar to Accuracy but it removes the probability that an accurate prediction occurs by chance), and Sensitivity (proportion of observations of class *k* predicted correctly). Their values are estimated against a test dataset, instead of the training dataset. Following machine learning's modeling emphasis on predictive ability rather than interpretation of model parameters, just tuning parameters for each model are reported. Tuning parameters for the seven models are captured in Table 5, while Table 6 contains sensitivities by approach and Table 7 summarizes Accuracy and Kappa values by approach.

Table 5 Tuning parameters by approach

Approach	Parameter	Value
Naïve multinomial	NA	NA
Learning multinomial	λ	0.0001
Random forests	*mtry*	68
	B	50
Neural networks	H	7
	λ	0.3
Gradient boosting	Z	50
	d	0
	s	0
	u	0.3
Support vector machine	σ	0
	C	1
Ensemble learning	NA	NA

Table 6 Sensitivity statistics by approach

Approach	Minimum	Average	Maximum	Variance
Naïve multinomial	0.00	0.22	0.89	0.06
Learning multinomial	0.00	0.37	0.89	0.05
Random forests	0.00	0.62	1.00	0.04
Neural networks	0.00	0.05	0.85	0.03
Gradient boosting	0.00	0.63	1.00	0.05
Support vector machine	0.00	0.10	0.85	0.04
Ensemble learning	0.17	0.68	1.00	0.03

The naïve multinomial logit model is used as the benchmark to evaluate performance of subsequent machine learning models. It does not have any tuning parameters by design and thus does not have any records in Table 5. Sensitivity ranges from 0.00 to 0.89, with an average value of 0.22, while Accuracy is 0.37. However, we can see in Fig. 1 that prediction Accuracy for this model improves significantly as SKU observations increase.

Kappa is equal to 0.36, coming close to the Accuracy estimate. The closeness between these two numbers is likely a result of the large number of classes being predicted. With as many classes as the ones included in this analysis (149), the likelihood of an accurate prediction being the result of chance becomes small. Thus, the closeness between Accuracy and Kappa. This is further validated by the fact that we see the same dynamic in results for all approaches presented below. These performance statistics portray our naïve multinomial model as relatively poor at predicting next SKU purchase by a panelist. However, this low performance should not be surprising, since the model attempts to estimate next purchase choice by each panelist (out of 3,404) at the SKU level (out of 142), which is a relatively demanding forecasting task.

The learning multinomial logit model's optimal decay tuning parameter, λ, equals 0.0001, which is a relatively low penalization. Sensitivity estimates for the SKUs being predicted have a similar range to that observed for the naïve multinomial logit model, with only a slightly greater mean. Accuracy and Kappa for this model also show moderate improvement relative to the benchmark. Accuracy's trend improves for SKUs with larger number of observations, as it was the case for the naïve multinomial logit model (Fig. 1). Based on these results, we can conclude that the learning multinomial model performs better than the naïve multinomial model, overall. The improvement in performance by the learning multinomial logit model is a result of just applying data preprocessing and regularization.

Table 7 Accuracy and Kappa estimates by approach

Performance metric	Naive multinomial	Learning multinomial	Random forests	Neural networks	Gradient boosting	Support vector machine	Ensemble learning
Accuracy	0.37	0.48	0.70	0.18	0.69	0.11	0.76
Kappa	0.36	0.45	0.70	0.16	0.69	0.10	0.76

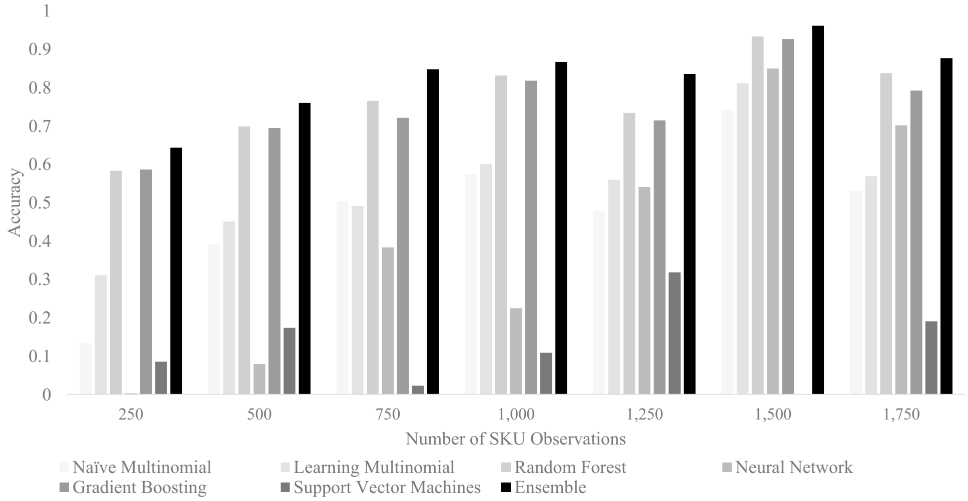

Fig. 1 Method accuracy by number of SKU observations

The random forest model selected the tunning parameter *mtry* to be 68 and B equal to 50. Sensitivity estimates for the SKUs being forecast show a sizable relative improvement vs. the naïve multinomial logit model, and even its learning counterpart. The same can be said about Accuracy and Kappa values. Accuracy is relatively strong for even the lower range of class observations (Fig. 1), but as shown for the prior approaches, it also benefits from a greater number of observations.

The neural network model's optimal tunning parameters are $H=7$ hidden layers and $\lambda=0.3$ decay (Table 6). Accuracy and Kappa are markedly worse than the performance of the naïve multinomial logit model. Also, the range of sensitivities is only slightly narrower, but lower on average. This approach shows the weakest performance at the lower range of class observations, with sizable improvements as observations grow (Fig. 1), however. The neural network model thus shows the greatest need for observations to perform well, as far as this dataset is concerned.

The gradient boosting model has four tuning parameters with optimal levels estimated at $Z=50$, $d=0$, $s=0$ and $u=0.3$ (Table 5). Accuracy and Kappa resulting from the application of gradient boosting are relatively high, just shy of the values reached by the random forest model, and better than for the benchmark naïve multinomial logit model. Sensitivity is on average also greater than for even the random forest. Accuracy shows relatively robust levels even with few observations (Fig. 1), while still benefiting from greater number of observations.

The support vector machine has two tuning parameters σ and C. Their optimal values are estimated to be $\sigma=0$ and $C=1$ (Table 5). The Accuracy and Kappa levels for the SVM model are the lowest of all tested approaches, including the naïve multinomial logit benchmark model, while sensitivity is only slightly better than for the worst performer, the neural network model. Performance is volatile across the entire range of SKU observations (Fig. 1), without consistent improvements as the number of observations grow. SVM proves to be weakest and least stable approach for the dataset used in this paper. It is worth noting that at 1,500 observations per SKU (Fig. 1), the SVM approach failed to converge, stressing further its relatively unstable behavior for this dataset.

The ensemble model has no tuning parameters since it is a simple majority vote algorithm. Its range of sensitivities across predicted SKUs is the narrowest and highest on average of all models evaluated. Its superior performance is also reflected by the highest Accuracy and Kappa levels: Accuracy is six percentage points better than the next best model's (the random forest model). Furthermore, Fig. 1 shows that the ensemble model's SKU choice prediction outperforms consistently every other model at any number of SKU observations predictions.

Discussion

Machine learning is bringing a great deal of due diligence to our analyses in marketing research. Its focus is largely directed towards improved model predictive performance. Particularly, it is doing so by turning researchers to evaluate multiple forms of a conceptual model, and different alternative techniques to estimating parameters in one single swoop. That systematization of alternative approach testing is done with computationally intensive automated algorithms based on tuning parameters and cross-validation. This is a move away from parametric and towards non-parametric methods that do not make explicit assumptions about the functional form of *f(X)*. Non-parametric methods, however, often have higher observational requirements and come at also higher computational costs. Machine learning also increases the likelihood that a model will perform well in a new environment, by splitting observations between training and test datasets. This emphasis is of particular importance to applied research in the commercial fields of marketing, reducing risk for decision-makers.

In this paper, we compare a naïve multinomial logit product purchase predictive model against six other alternatives: learning multinomial logit, random forests, neural networks, gradient boosting, support vector machines and an ensemble learning algorithm. The results show that random forest and gradient boosting, in that order, perform best among the six single model approaches (excluding the ensemble model). However, some machine-learning approaches, namely neural networks and support vector machines, resulted in weaker predictive performance. In fact, both neural network and support vector machine approaches perform worse than even the naïve multinomial logit benchmark model, on average. However, neural network's relative performance improves significantly with greater observations, surpassing naïve multinomial logit model's performance at the higher end of observations (Fig. 1). On the other hand, support vector machine's performance remains low and somewhat volatile across the spectrum of observations (Fig. 1). That is likely a result of the fact that support vector machines is perhaps the most data demanding of all methods tested in this paper, given its reliance on large number of kernels.

An important finding is that the ensemble learning algorithm, that combines predictions from all the other six models, displays the best predictive performance, six percentage points above the next best approach. This is a result of ensemble learning's ability to increase predictive accuracy for SKU classes that prove to be difficult to predict by single models. Conventional wisdom would make us believe that the approach of choice should be to rely on the best-performing single model. Thus, marketing researchers may need to consider not only testing a wide range of machine-learning models to answer specific questions, but also taking one step further to generate ensemble solutions as a potential best-in-class approach to generate robust answers in any observational conditions.

The results of this paper resonate with findings in similar studies. For example, Bajari et al. (2015) demonstrated that machine learning methods led to superior predictive accuracy relative to the more traditional linear regression or logit models when applied to aggregate demand data. Further, they also found that an ensemble model in the form of a linear combination of the underlying models can improve fit even further, as it is also shown in this paper. The focus of their work was explaining longitudinal continuous aggregated salty-snacks demand at the store level, rather than predicting individual consumer choice events like in this paper. This is evidence of the robustness of machine-learning across categories and aggregation levels. There are also studies that point to traditional choice models, like the multinomial logit model used as a benchmark in this paper, as superior to machine learning. Feldman et al. (2022) find that gradient boosting underperforms relative to a traditional multinomial logit model challenger. They make the side-by-side comparison between the two approaches for two of Alibaba's marketplaces (Tmall and Taobao), using a randomized field experiment. Specifically, they expose two randomized sets of consumers coming to their websites to product recommendations generated by one of the two models: one set of consumers was exposed to product recommendations from the gradient boosting model and another set of consumers to recommendations from the multinomial logit model. They then compare the revenue generated per customer visit between the two cells. The results showed a 28% lift in revenue per customer when the recommendations were driven by the multinomial logit model, relative to when gradient boosting was making the recommendations. This finding is a lesson that suggests that we should not always take it as a given that machine learning techniques are superior. Instead, multiple comparisons should be made against out-of-sample or test datasets. Even better, the comparisons should be made by applying randomized field experiments such as Feldman et al. (2022) do. The benefit of this approach is threefold: (1) we can feel confident of the causality behind the results, (2) the model performance comparison is done in a realistic setting that considers operational elements, marketplace dynamics and consumer behavior, and (3) it is measured against business outcomes rather than statistical measures like accuracy.

Both researchers and practitioners will find value in the flexibility, ease-of-use, and scalability of machine learning methods for a wide variety of marketing research problems, not just product purchase prediction. Machine learning may have already advanced more amongst practitioners in the marketing research space. This could in part be because it focuses on prediction vs. explanation and validation of theoretical constructs of consumer behavior. Marketer incentives may skew more towards getting it right, than in understanding the underlying dynamics of consumer choice processes. As a result, the implications of continued improvements in consumer product choice predictions with machine-learning loom large for the industry. Both consumers and suppliers, in the form of either manufacturers and/or retailers, can benefit from improvements to consumer product choice prediction. By applying machine-learning techniques uncertainty of future demand can be reduced, according to the results presented in this paper. This can improve supply chain efficiencies on both the manufacturer and retailer sides significantly—which is an area that continuously tries to find ways to smooth operations and reduce waste. Better forecasts of consumer choice can help manufacturing investment allocations and assortment optimizations. Inventory of idle product costs and penalties incurred in by manufacturers for not meeting supply needs by the retailer will be reduced. Another area improved predictions will impact is a decline in opportunity costs from not having product at the ready to meet consumer choices. Consumers that are unable to find their preferred choice, will result in lower levels of satisfaction. Thus, a direct effect of better planning resulting from improved consumer choice predictions will then prevent loyalty erosion on brand and retailer. Academia's mission, on the other hand, tends to be more focused on validation of consumer behavior theories. It is often argued that traditional methods are better at uncovering the whys behind observed consumer behavior, while machine learning approaches are better at predicting it. Traditional models are relatively easier to interpret, while machine learning models tend to be more convoluted and difficult to wrap one's head around. Conventional wisdom may be worth being challenged in future research, though. There is room to investigate whether machine learning models can yield even richer understanding and new hypotheses about consumer behavior than traditional models. They could do so by leveraging these models in simulation runs under alternative scenarios. The findings could then be confirmed with the application of designs of experiments to truly establish causality. Comparing the learnings of this two-step approach (machine learning based simulations first, followed by design of experiments) relative to those from traditional statistical modeling may be worth considering.

Conclusion

It is concluded based on the analysis presented in this paper that machine-learning techniques in general are superior to a traditional approach in predicting consumer product choices, but their performance can vary depending on the approach, and presumably the dataset. In addition to their intrinsic inherent strengths, machine learning techniques consistently follow a series of steps that make them formidable relative to traditional statistical approaches. These steps include data preprocessing, feature engineering, data splitting between training and test sets, cross-validation, and use of tuning parameters (Kuhn and Johnson 2013). The result of such painstaking process is more flexible relationships between predictor and predicted features, and less over-fitting. The net benefit of such due-diligence is better out-of-sample predictions.

A salient aspect of the findings in this paper is that ensemble models that rely on both machine learning and traditional modeling approaches are particularly robust. Best practices in both academic and applied marketing analytics should involve implementation of multiple modeling approaches, including traditional and machine learning techniques, to analyze every individual study. This set of multiple diverse models should then be leveraged to generate an ensemble model. This meta-modeling technique is likely to be the most robust, as shown by this and similar papers (Bajari et al. 2015). There is room for further research when it comes to the development and performance of ensemble models in the marketing analytics space. There are a variety of ways to build this type of models-of-models: combination methods (like majority vote used in this paper), diversity, ensemble pruning, boosting, clustering, and bagging are just a few of them. Thus, identifying which methods work best for marketing and consumer analytics is a fertile ground for future research.

The chain reaction unleashed by enhanced predictions has two large implications: one to consumers and another to supply chain stakeholders including manufacturers, distributors, and retailers. Better predictions of future consumer behavior will anticipate supply needs. With that foresight, meeting those supply needs will then result in fewer out-of-stocks encountered by consumers looking for their preferred products. Fewer out-of-stocks will likely translate into a build-up of consumer satisfaction levels. This buttressing of satisfaction on the part of consumers will likely result in improved loyalty for products, manufacturers, and retailers. A final consequence of this sequential domino reaction started with improved predictions of consumer preferences is that the loyalty increase will help supply chain stakeholders, including manufacturers and retailers, generate better business results.

Although outside of the scope of this paper, an area that may be worth challenging is the maxim that machine learning models are catered for prediction, while traditional statistical models are better at testing hypotheses. If it is established that machine learning models can predict consumer behavior better, it stands to reason that they should be able to test hypotheses better also. The way to test hypotheses may require simulations of conditions we are interested in, based on more complex single or ensemble models that may not have theoretical underpinnings. This is a departure from traditional parametric hypothesis testing, but necessary in a world where complex dynamics are recognized (often beyond our ability to understand) and captured. In any case, one can also argue that for true causal hypothesis testing, executions of experimental designs may be needed rather than stopping at just observational data driven traditional or machine learning model generated results.

References

Bajari, P., D. Nekipelov, S.P. Ryan, and M. Yang. 2015. Machine Learning Methods for Demand Estimation. *American Economic Review: Papers & Proceedings* 105 (5): 481–485.

Breiman, L. 2001. Random Forests. *Machine Learning* 45: 5–32.

Breiman, L., J. Friedman, R. Olshen, and C. Stone. 1984. *Classification and Regression Trees*. New York: Chapman and Hall.

Bridle, John S. 1990. Probabilistic Interpretation of Feedforward Classification Network Outputs, with Relationships to Statistical Pattern Recognition. In *Neurocomputing: Algorithms, Architectures and Applications*, ed. Françoise Fogelman and Soulié J. Hérault, 227–236. Berlin: Springer-Verlag.

Bronnenberg, B., M. Kruger, and C. Mela. 2008. Database paper: The IRI marketing data set. *Marketing Science* 27 (4): 745–748.

Chintagunta, P., and J. Dubé. 2005. Estimating a Stockkeeping-Unit-Level Brand Choice Model That Combines Household Panel Data and Store Data. *Journal of Marketing Research* 42 (3): 368–379.

Dzyabura, D., and H. Yoganarasimhan. 2018. Machine Learning and Marketing. In *Handbook of Marketing Analytics: Methods and Applications in Marketing Management, Public Policy, and Litigation Support*, ed. Natalie Mizik and Dominique M. Hanssens, 255–279. Northamptom, MA: Edward Elgar Publishing.

Efron, B., and R. Tibshirani. 1993. *An Introduction to the Bootstrap*. London: Chapman & Hall.

Feldman, J., D. Zhang, X. Liu, and N. Zhang. 2022. Customer Choice Models vs. Machine Learning: Finding Optimal Product Displays on Alibaba. *Operations Research* 70 (1): 309–328.

Friedman, J. 1991. Multivariate Adaptive Regression Splines. *The Annals of Statistics* 19 (1): 1–67.

Friedman, J. 2001. Greedy Function Approximation: A Gradient Boosting Machine. *The Annals of Statistics* 29 (5): 1189–1232.

Hastie, T., R. Tibshirani, and J. Friedman. 2009. *The Elements of Statistical Learning: Data Mining, Inference and Prediction*. New York: Springer.

Hauser, J., O. Toubia, T. Evgeniou, and R. Befurt. 2010. Disjunctions of Conjunctions, Cognitive Simplicity, and Consideration Sets. *Journal of Marketing Research* 47 (3): 485–496.

Huang, D., and L. Luo. 2015. Consumer Preference Elicitation of Complex Products using Fuzzy Support Vector Machine Active Learning. *Marketing Science* 35 (3): 445–464.

Iskhakov, F., J. Rust, and B. Schjerning. 2020. Machine Learning and Structural Econometrics: Contrasts and Synergies. *Econometrics Journal* 23 (3): 81–124.

James, G., D. Witten, T. Hastie, and R. Tibshirani. 2013. *An Introduction to Statistical Learning: With Applications in R*. New York: Springer.

Kazemia, S., E. Hadavandib, F. Mehmanpazirc, and M. Nakhostinc. 2013. A hybrid intelligent approach for modeling brand choice and constructing a market response simulator. *Knowledge-Based Systems* 40: 101–110.

Kuhn, M., and K. Johnson. 2013. *Applied Predictive Modeling*. New York: Springer.

Liu, L., and D. Dzyabura. 2016. Capturing Multi-taste Preferences: A Machine Learning Approach. *SSRN Electronic Journal*. https://doi.org/10.2139/ssrn.2729468.

Russell, G. 2014. Brand Choice Models. In *The History of Marketing Science*, ed. Russell Winer and A Neslin Scott, 19–46. Hanover, MA: Now Publishers.

Sagi, O., and L. Rokach. 2018. Ensemble learning: A survey. *Wiley Interdisciplinary Reviews: Data Mining and Knowledge Discovery* 8 (4): e1249.

Venables, W., and B. Ripley. 2002. *Modern Applied Statistics with S*. New York: Springer.

Yoganarasimhan, H. 2020. Search Personalization using Machine Learning. *Management Science* 66 (3): 1–52.

Publisher's Note Springer Nature remains neutral with regard to jurisdictional claims in published maps and institutional affiliations.

Josué Martínez-Garmendia professional focus is Marketing and Business Analytics. Specifically, he applies econometric, machine learning and design of experiments techniques to understand consumer behavior and optimize business decisioning. He has extensive experience in consumer-packaged-goods, media, and financial services (www.linkedin.com/in/josué-martínez-6771536). He is also a part-time Lecturer at Columbia University's School of Professional Studies, University of California Berkeley's School of Information, and University of California Davis' Graduate School of Management. The analyses and views expressed in this paper, including any errors, are solely attributable to the author, and not to the institutions he is affiliated with currently or in the past.