ISSN: 2831 - 6789

Research Article

Journal of Robotics and Automation Research

Increasing Generalizability: Naïve Bayes Vs K-Nearest Neighbors

Fahad Mansoor Pasha

Fahad Mansoor Pasha is an Assistant Professor (Marketing) at the School of Management, FAST-NUCES-LAHORE, Pakistan

***Corresponding author**
Fahad Mansoor Pasha is an Assistant Professor (Marketing) at the School of Management, FAST-NUCES-LAHORE, Pakistan.

Submitted: 03 Jun 2022; **Accepted:** 15 Jun 2022; **Published:** 01 Jul 2022

Citation: *Fahad Mansoor Pasha. (2022). Increasing Generalizability: Naïve Bayes Vs K-Nearest Neighbors. J Robot Auto Res 3(2), 178-188.*

Abstract

Marketing research is often criticized for lacking generalizability and inability to reproduce results. The problem lies in using models to fit data, rather than determining the predictive power of models in conditions of uncertainty. For instance, how does the predictive power of a model change when customer dynamics change? The current study suggests that marketing researchers can supplement existing research methods with non-probabilistic prediction methods, such as the kNN algorithm-based model. Unlike probabilistic models that rely on past outcomes to predict future events – and lose predictive power when newer events are observed - non-probabilistic models better capture uncertainty. In the current study, the predictive power of the kNN algorithm-based model and the Naïve Bayes model is compared using data from two real markets. The kNN algorithm-based model provides more accurate predictions, showing the utility of combining the kNN algorithm-based model with existing marketing research to improve the predictability and generalizability of models. Implications for research and future research are discussed.

Keywords: kNN; Naïve Bayes; generalizability; methods.

Statement of Intended Contribution

Marketing research is often criticized for producing studies that provide non-reproducible and non-generalizable results. Although many articles are dedicated to improving the generalizability of results, no study suggests an empirical solution. In the current study, the kNN algorithm-based model is highlighted as a solution to increase the reproducibility and generalizability of results.

The current study is the first of its kind. The current study suggests that marketing studies lack generalizability and reproducibility due to a disproportionate focus on fitting models to data. However, in real-life settings, customer habits and market dynamics change, bringing an element of uncertainty. Since Marketing studies rely on probabilistic methods to derive relationships within data, newer outcomes (i.e. outcomes that do not have a probability of occurrence) are predicted poorly as market dynamics change. Hence, the generalizability and reproducibility of experimental results decrease.

The current study suggests that the generalizability and reproducibility of experimental results can increase by using a non-probabilistic method, such as the kNN algorithm-based model. The kNN algorithm-based model is easy to use and calculates the distance between features (and identifies patterns). Since customer habits and behavioral patterns are less likely to change, the kNN algorithm-based models can accurately predict outcomes when market dynamics change and uncertainty increases.

The current study compares results from two markets using the kNN algorithm-based model (non-probabilistic model) and the Naïve Bayes model (probabilistic model), showing that the kNN algorithm-based model better predicts outcomes. The results of the study have important implications. The results suggest that marketing researchers can increase the generalizability and reproducibility of experimental results by supplementing experimental results with the kNN algorithm-based models.

Increasing Generalizability: Naïve Bayes Vs K-Nearest Neighbors

Introduction

Many empirical methods are studied within the Marketing discipline to make accurate predictions [1]. Such methods include linear, non-linear, and non-parametric models [2-4]. However, the utility of a predictive model relies on processing old information to make accurate predictions with newer data. Models that only create predictions using one set of data can overfit [5,6]. For instance, regression models make predictions using the Ordinary Least Squares method of minimizing deviations from the mean using a single dataset. The utility of a predictive model using the Ordinary Least Squares method is difficult to estimate unless the predictive model generated from the Ordinary Least Squares method is tested on another set of data.

Studies that check the accuracy of predictive models using a test dataset are uncommon within the Marketing discipline, although such studies are present in other disciplines [7,8]. Over the past few decades, the diversity of research methods within the Marketing discipline has observed a downward trend, with most studies focusing on experiments and modeling [9]. The absence of studies within the Marketing literature that verify the accuracy of predictive models can create problems with generalizability [10]. For instance, if a predictive model is exposed to newer data, but the predictive model produces inaccurate results, what is the utility of such a predictive model with low levels of accuracy? Numerous studies have cited the difficulty in reproducing results within the Marketing discipline [10,9,11]. However, the Marketing literature does not propose an approach that can help Marketers assess the accuracy of predictive models.

The current study proposes an approach to help Marketers assess the accuracy of predictive models, helping increase the generalizability of results. The current study proposes the use of the kNN (K-nearest neighbors) algorithms to assess the accuracy of predictive models. The kNN algorithms use distance-based calculations, are easy to understand and use, and provide remarkable predictive accuracy compared to probabilistic models, such as the probabilistic models based on the Bayes Theorem. The current study compares the predictive accuracy of the kNN algorithm-based model and the Bayes Theorem-based model across two real markets. To determine how both the methods predict under uncertain conditions, data from one market is used to train the algorithms, while data from another market is used to test the accuracy of the predictive models.

Results show that the kNN algorithm-based model has better predictive power, potentially because the Bayes Theorem-based model cannot accurately predict outcomes that have not occurred. On the other hand, since the kNN algorithm-based model focuses on calculating distances between features, rather than considering the probability of events, the kNN algorithm-based model better predicts newer outcomes. Since customer behavior is ever-evolving, using probability-based methods to predict customer behavior decreases predictive accuracy. However, the kNN algorithm-based model better predicts newer outcomes, since customers have similar habits that change relatively slowly. By focusing on features associated with customer habits, such as the combined use of cola and chocolate, the kNN algorithm-based model can create better associations and increase predictive power.

The current study contributes to the Marketing literature on generalizability and empirical method selection. The current study contributes to the generalizability literature by showing how supplementing existing research techniques with the kNN algorithm-based model can increase predictive power and generalizability when uncertainty increases, such as when consumer habits change. Secondly, the current study contributes to the empirical method selection debate by showing how marketing researchers can easily include the kNN algorithm-based model with existing experiments and models, improving the robustness of research results.

Theoretical Background

Many predictive methods are discussed within the Marketing literature [9]. Some of these derive from the regression methods, while others focus on analyzing between-group variance [12,13]. However, all the methods discussed within the Marketing literature are probability-based. For instance, regression-based models rely on the law of large numbers and the central limit theorem to approximate beta values based on an asymptotic normal distribution [14]. The models use the F test to determine the joint significance of coefficients. Similarly, ANOVA methods tabulate the Chi-square statistics and calculate the probability of events [15]. Hence, marketing research relies on probability-based calculations to create predictive models.

However, given the nature of ever-changing consumer behavior and habits probability models that rely on past events to predict future outcomes will have little generalizability [16,17]. For instance, if consumers purchase coke and candy instead of coke and muffins, predictive models based on past instances of coke and candy consumption will have decreased predictive power. Predictive models base predictions using past probabilities of events to determine future likelihood.

A question that arises is, what happens if we use a simple non-probabilistic measure to make predictions, like using a distance measure? In probabilistic models, joint probabilities are calculated [18]. However, another way to use mathematical data is to plot points on a vector space and find the distance between points: a method commonly known as the Euclidean distance [19,20]. Is it possible that the more simple Euclidean distance method makes better predictions compared to the more complex probabilistic methods? No studies in the Marketing literature compare the predictive accuracy between the probabilistic methods and the Euclidean distance method.

The current study addresses the shortcomings. The current study compares the utility of a probabilistic method and a non-probabilistic method: the Bayes Theorem and the kNN algorithm. The Bayes Theorem is the basis for many statistical models, while the kNN algorithm is a simplistic distance-based calculation algorithm. The current study compares the predictive accuracy of the kNN algorithm and the Bayes Theorem. The underlying assumptions and structures of the kNN algorithm and the Bayes Theorem are discussed first.

kNN (K-nearest neighbors) Algorithm

The kNN (where k is an integer) algorithm offers distinct advantages. Firstly, it makes no assumptions about the underlying data distribution [21]. The algorithm simply plots observations as vectors on a multi-dimensional space and calculates the distance between the observations. For instance, in classifying customers into loyal/disloyal groups using customer features - such as customer purchases, customer income, and the number of items purchased - locating the nearest neighbor (loyal/disloyal customer) for a customer requires a distance function that measures similarity between two features (e.g. customer income and customer purchases) [22]. There are many methods to measure the distance between two features. The most basic method is the Euclidean distance.

The Euclidean distance formula involves comparing the values of each feature, such as comparing customer incomes (e.g., *0=below 1,000$, 1=above 1000$ and less than 4000$, 2= more than 4000$ and less than 6000$, 3= more than 6000$) and customer purchases (e.g., 0= less than 3 products purchased each week, 1= more than 3 products purchased each week and less than 6 products purchased each week, 3= more than 6 products purchased each week*) using equation 1.

$$d(p,q) = \sqrt{\sum_{i=1}^{n}(q_i - p_i)}^2 \qquad (1)$$

In equation 1, p and q are two points plotted on the n-dimensional space, pi and qi are initial points of the space, and the space has n dimensions. For instance, if customer A has income below 1000$ (point 0) and purchases more than 6 products a week (point 3), whereas customer B has an income of more than 6000$ (point 3) and purchases more than 3 but less than 6 products each week (point 3), the distance between the two customers is given in equation 2.

$$\sqrt{}(6-3)^2 + (3-0)^2 = \sqrt{18} = 4.24 \qquad (2)$$

The kNN algorithm tabulates combinations of all distances for customer A with all customers in a dataset (e.g., 10,000 other customers in a department store's dataset). To classify customer A as loyal/disloyal, customer A is assigned the class (i.e. loyal/disloyal) of the customer (i.e. nearest neighbor) that has the least distance among all other customers (e.g. class of customer with a distance of 1.4 is assigned to customer A compared to the class of a customer with a distance of 2.9).

The classification depends on the number of nearest neighbors chosen. For instance, if K=1, only the single nearest neighbor (i.e., the customer with the shortest distance) is chosen. If K=3, a vote among 3 nearest neighbors (i.e., 3 customers with the shortest distance) is done and the majority class among the customers with the shortest distance is selected. For example, if 2 loyal customers are present among the three customers compared with customer A, customer A will be classified as loyal [23]. The choice of K influences the generalizability of the results. If K=1, the single nearest neighbor will be chosen. However, if data is biased or mistaken, the kNN algorithm will wrongly classify future data based on the wrong training data. Using K=1 allows noisy data or outliers to influence classification, such as when a training example is mislabeled [21].

If the mislabeled example is near to an unlabeled example in the test data, errors can occur. On the other hand, if a large number of K is chosen, the model will overfit by finding too many features that are not generalizable. If k is equal to the number of observations, the majority class always wins and the model will always predict the majority class regardless of which neighbors are nearest (e.g., if more than 50% of customers in a dataset are loyal, the kNN algorithm always classifies future customers as loyal).The best K value lies between K=1 and K= n, where n is the number of observations. A rule of thumb to select K is to set K equal to the square root of the number of observations in the training data [21]. However, testing several values of K on a variety of test data provides the best classification estimates. If datasets have little measurement error, selection of K loses importance, since even subtle concepts have a sufficiently large pool of examples to vote as nearest neighbors.

Advantages of kNN algorithms

Due to the simplicity of the kNN algorithm, the model training phase is quick. However, a shortcoming of the kNN algorithm is that the kNN algorithm does not produce a parametric model [24]. The kNN algorithm is a non-parametric learning method; no parameters are learned from the data. The kNN finds natural patterns rather than trying to fit data in a preconceived form or using a theoretical basis. The kNN algorithm is first trained on a dataset. The training dataset consists of examples pre-classified into several nominal categories (e.g., loyal/disloyal customers). After the kNN algorithm is trained on a dataset, the kNN algorithm is provided with an unlabeled dataset to test the utility of the kNN algorithm in predicting an outcome of interest (e.g., predicting loyal/disloyal customers based on new customers).

For instance, if marketers want to classify customers into loyal/disloyal categories, a training dataset is provided to the kNN algorithm. The training dataset has customers classified into loyal/disloyal categories and contains several customer features that the kNN algorithm uses to make predictions, such as customer purchases, customer incomes, and the number of items purchased. The kNN algorithm plots customer features on a multidimensional space as vectors and calculates the distance between various features and categorizes the features based on the distance between the features. The kNN algorithm treats the features as coordinates in a multidimensional feature space. For instance, if three features are used to predict customer loyalty/disloyalty - customer purchases, customer incomes, and the number of items purchased – the three features are plotted on a 3-dimensional space [25]. The kNN algorithm identifies patterns until loyal/ disloyal customers are grouped. For example, loyal customers purchase more and buy more items, but may have both high and low incomes. The kNN algorithm uses the least distance between features (hence the name nearest neighbor) to determine which class (loyal/disloyal) is a better fit for a new customer.

The test dataset must contain the same features that the kNN algorithm is provided in the training dataset. However, in the test dataset, customers are not labeled into loyal/disloyal categories. For each customer in the test dataset, kNN identifies K records in the training data that are the "nearest" in similarity, where K is an integer specified in advance. The unlabeled customer in the test data is assigned the class of the majority of the K nearest customers (i.e. K nearest neighbors). Hence, the kNN algorithm classifies customers in the test dataset based on relationships identified in the training dataset. Based on the predictive accuracy of the kNN algorithm on the testing dataset, marketers can provide newer data to the kNN algorithm and help determine loyal/disloyal customers. The kNN algorithm can simplify the decision-making process for marketers. Marketers can focus on disloyal customers by increasing loyalty or focusing scarce resources on loyal customers.

Bayes Theorem

Probabilistic methods describe uncertainty. Probabilistic methods use past events to predict future events. For instance, the chance of winning a soccer match describes the proportion of prior matches with similar game conditions in which a team won soccer matches. A popular probabilistic technique is the Bayes Theorem. The Bayes Theorem is used in many probability methods [26]. For instance, word frequency in emails can identify junk mails and create junk filters [33]. The Bayes Theorem describes the probability of events and how probabilities are revised in light of more recent information. In Machine learning, Bayes Theorem is used to classify data by using a training dataset to calculate the observed probabilities of each class based on feature values. When a predictive model trained using Bayes Theorem is exposed to a new dataset, the predictive model classifies new unlabeled data based on the observed probabilities of train data.

Bayes Theorem is especially useful when situations require considering several attributes simultaneously [21]. Bayes Theorem can combine the impact of many features with minor effects into a combined larger impact. The Bayes Theorem leverages the fact that many events occur jointly, and the occurrence of one event is used to predict another. For instance, the presence of high educational attainment can be used to predict high future incomes. Using information about one event, such as high education, the probability of high education and high income occurring together are calculated. Similarly, the likelihood of selling products at higher than mean prices can be calculated based on the presence of certain product features.

The relationship between independent events is described using the Bayes theorem in Equation #3 (Joyce 2003). In Equation #3, P(A|B) is the probability of event A given that event B occurs. P(A|B) is the conditional probability since the probability of occurrence of A (e.g. selling at higher than mean price) depends on event B (e.g. type of product sold).

$$P(A|B) = \frac{P(B|A)\,P(A)}{P(B)} = \frac{P(A\cap B)}{P(B)} \qquad (3)$$

As an example, to calculate the probability of higher than mean prices given a product with yellow packaging sold, Equation #4 is shown

$$\frac{\text{P(yellow product package |price above mean) P(price above mean)}}{\text{P(yellow product package)}} \qquad (4)$$

Equation #4 can be written as Equation #5 or as Equation #6

$$\frac{\text{P(yellow product package |price above mean) P(price above mean)}}{\text{P(yellow product package)}} \qquad (5)$$

$$\frac{P\text{(price above mean}\cap\text{ yellow product package)}}{P\text{(yellow product package)}} \qquad (6)$$

Where P(price above mean| yellow product package) is known as the posterior probability and P(yellow product package |price above mean) is known as the likelihood and P(price above mean) is known as the prior possibility.

Method. To compute the Bayes theorem, a frequency table recording the number of times features occur in combination with a class is required. For instance, to see how frequently loyal/ disloyal customers appear with income greater than 4,000$, a two-way cross table (Table 1) shows one dimension level of class variable (loyal/ disloyal customer) and a second dimension indicating a feature (incomes below 4,000 or above). Table 1 shows a likelihood table used to calculate frequencies and probabilities for the Bayes Theorem.

Table 1: Probability Estimation Using Bayes Theorem.

Frequency	**Income levels (A)**		**Total**
	Below 4000$	**Above 4000$**	
Loyal	75	244	319
Disloyal	182	47	229
Total	257	291	

A commonly used algorithm that employs the Bayes Theorem for probability calculations is the Naïve Bayes (NB) algorithm [27]. The NB algorithm creates classifications. The NB algorithm is simple to use and requires relatively few examples for training. The NB algorithm makes some naïve assumptions about the data (hence the name Naïve Bayes). For instance, all features are given equal importance and are treated as independent events. A distinct advantage of the Naïve Bayes algorithm is that it can combine information from a large number of features. Suppose that in addition to income levels (i.e. income below or above 4,000$), we have information about product types purchased (e.g. durable/non-durable products) and whether the company email newsletter is subscribed or not. The Naïve Bayes can combine information across several features and calculate the probability of a customer being loyal/disloyal. As more features become available to the Naïve Bayes algorithm, information regarding all possible intersecting events is used to create probabilities of customer loyalty/disloyalty [28]. For example, if a customer's income is below 4000$, the customer buys durable goods, and the customer subscribes to the company email newsletter, the customer is likely loyal. As features increase, modifications are made to Table 1 to calculate probabilities using the Bayes Theorem, as shown in Table 2.

Table 2: Probability Estimation Using Bayes Theorem When Several Features Are Available.

Frequency	Income levels (A)		Company newsletter subscribed (B)		Total
	Below 4000$	Above 4000$	Yes	No	
Loyal	75	244	22	45	386
Disloyal	182	47	0	24	253
Total	257	291	22	69	

However, as features increase, calculations to tabulate probabilities using Bayes Theorem become complex. As the number of features increases, Equation #3 is modified, as shown in Equation #7 (where A=income levels and B=company newsletter subscribed).

$$P(Loyalty|A \cap B) = \frac{P(A \cap B|Loyalty)\ P(Loyalty)}{P(A{\cap}B)} \qquad (7)$$

Adding more features increases the complexity of calculations. To simplify calculations, the Naïve Bayes assumes conditional class independence, implying that events are independent when they are conditioned on the same class value. Tabulations of intersecting events become easier since the Naïve Bayes assumes independence among events, as shown in Equation #8 [21].

$$P(Loyalty|A \cap B) = \frac{P(A|Loyalty)\ P(B|Loyalty)P(Loyalty)}{P(A)P(B))} \quad (8)$$

The Laplace estimator. A problem that arises in Equation #8 is that an event might not occur for a certain class. For instance, disloyal customers may not subscribe to the company newsletter (see Table 2). Hence, probabilities are multiplied by P(Disloyal|company newsletter subscribed)=0%. The 0% value causes the posterior probability of disloyalty to be 0. Hence, zero probability of newsletter subscription nullifies and overrules evidence provided by other features, such as income levels. To prevent a non-occurring event from overruling evidence provided by occurring events, a Laplace estimator is used [29]. The Laplace estimator places a small number to each count in the frequency table to ensure a non-zero probability of occurrence in each class. Typically, the Laplace estimator is set to 1.

Difference between kNN algorithm and Naïve Bayes algorithm

A key difference between the kNN algorithm and the Naïve Bayes algorithm is that the kNN algorithm is trained and tested together (stage 1). In stage 2, the kNN algorithm-based model's predictive output is compared with the actual output from test data. In the Naïve Bayes model, the training and testing occur in separate stages, while the last stage involves comparing predicted output with actual output (similar to the case of the kNN algorithm). Data format. For the Naive Bayes model, each feature is categorical to help create likelihood combinations between classes and features. Any numeric data is converted to categories. For the kNN algorithm-based model, features are numerical.

Overview of Study

To test the utility of the kNN algorithm-based model and the Naïve Bayes model, data from two real markets is selected. Data from one market trains and creates predictive models, while data from the second market test the predictions. The data is gathered from two cattle markets in a highly populous developing country: the L Cattle Market and the S Cattle Market. The cattle markets are cash-rich and attract thousands of buyers and sellers. In both the cattle markets, expensive cows and buffalos are traded that provide high-quality milk [30]. When animals become old, the animals are slaughtered for meat consumption. Thus, animals in both markets are traded for milk consumption and meat consumption. Although expensive animals are sold in both cattle markets, buyers and sellers in each cattle market create their market dynamics. Hence, even though the two cattle markets are similar in terms of products exchanged, market dynamics differ. The two cattle markets provide an ideal opportunity to assess the predictive power of the kNN algorithm-based model and the Naïve Bayes model.

In the current study, data from one cattle market is used to train the kNN algorithm-based model and the Naïve Bayes model, and the data from the second cattle market is used to test the predictive power of both models. Cattle markets are dynamic and have many dimensions. A predictive model must consider all factors and predict events in another cattle market with approximately similar dynamics. The greater the predictive power of a model, the better the model is at predicting unforeseen events. Once a model is identified that can accurately predict outcomes across both cattle markets, the model can be used for predicting marketing-relevant outcomes. For instance, we can predict loyal/disloyal customers, high/low sale prices, satisfied/dissatisfied customers, etc.

The two cattle markets are administered by a single organization. The administering organization does not interfere in buying or selling in the cattle markets. Rather, the administering organization only fulfills administrative responsibilities, such as maintaining cleanliness, providing security, supplying amenities for buyers and sellers, and managing parking spaces. Since similar policies are implemented in both cattle markets, the impact of administrative policies on the study is controlled.

The administering organization has introduced an electronic animal e-tagging system. Using the animal electronic e-tagging system, detailed information regarding animal characteristics and sales price is gathered at the entrance of the cattle markets. For instance, the animals are weighed and inspected when the animals enter and exit the cattle market. The use of the electronic animal e-tagging system ensures that measurement errors or researcher bias are minimized. Classification outcome. In the current study, we are going to predict if the sales price for an animal is below the mean price of that animal category (i.e., mean price of cow, buffalo, sheep, or goat) or not. Variables used in the study are reproduced in Table 3. To ease calculations, all data are dummy coded [21].

Table 3: Variables Used in Study

Outcome: Sales price classification (in Local currency) • ***Price below mean for the animal category*** • ***Price above mean for the animal category***	
Animal breed • ***Camel*** • ***Buffalo*** • ***Cow*** • ***Goat*** • ***Sheep***	*Animal weight (average in kg)* • *0-100 kg* • *101-200 kg* • *201-300 kg* • *301-400 kg* • *400 kg and above*
Animal gender • ***Male*** • ***Female***	*Sale purpose* • *Breeding* • *Meat* • *Milk*
Spots on the animal? • ***Animal has spots*** • ***The animal is without spots***	*Is animal large?* • *No* • *Yes*
Animal age (in years) • ***0-2 years*** • ***3-4 years*** • ***5-6 years*** • ***7-8 years*** • ***9-10 years***	*Distance between owner home and cattle market* • *0-100 km* • *101-200 km* • *201-300 km* • *301-400 km* • *401-500 km* • *501-600 km*
Animal color • ***Black*** • ***Brown*** • ***Other*** • ***White***	

While Table 3 identifies the variables used in the current study, Table 4 identifies the mean sale prices for all animal categories across the two cattle markets. Table 4 shows that in the S Cattle Market, Buffalos had the highest sales price, with an average of 94,902 Local currency. However, Buffalos in the L Cattle Market had a higher average price, with an average price of 100,537 Local currency. Cows and Goats sell at slightly higher average prices in the S Cattle Market, while Sheep are sold at twice the average price in the L Cattle Market. Table 4 shows that around 65% of animals are sold below the mean price for that animal type in the S Cattle Market, while around 40% of the animals are sold at higher than the average price for the animal type in the L Cattle Market. Small differences in the average prices between the two markets will reveal the ability of the kNN algorithm-based model and the Naïve Bayes model in predicting uncertainty in outcomes since no two markets are completely similar.

Table 4: Summary for Sales Price

Outcome: Sales price classification (in Local currency)		
	S Cattle Market	L Cattle Market
Mean sales price for animal category (in Local currency)		
Buffalo	94,902.72	100,537.1
Cow	49,920.53	44,256.52
Goat	9,570	6,029.412
Sheep	9,910.448	18,500
Outcome: Sales price classification (in Local currency)		
Price below mean for the animal category	65.01%	394
	59.5%	200
Price above mean for the animal category	34.9%	212
	40.4%	136

Model 1

kNN algorithm. To train the kNN algorithm-based model, features are transformed within a standard range, since different feature values and different scales can cause some features to strongly dominate distance calculations (Deng et al 2016). For instance, the maximum distance that the owner travels to the cattle market is 600 km, while the maximum age of an animal is 10 years. In distance calculations, distance traveled by owners to the cattle market will dominate the calculations. Hence, the data is scaled using the minimum-maximum normalization, when the minimum of X is deducted from X and divided by the range of X (see Equation #9) [21]. As a result, all data now falls within a range of 0 and 1 (normalized values show how far the feature values are from the original value on a scale of 0% to 100%). Each feature now contributes relatively equally to distance calculations.

$$X(new) = \frac{X - \min(X)}{\max(X) - \min(X)} \tag{9}$$

The Euclidean distance formula requires feature data in numeric format. As a result, all nominal data are dummy coded, with 0 and 1 indicating two categories. For instance, 0-100 km distance traveled by the animal owner to the cattle market is coded 0, while 101-200 km distance traveled by the animal owner to the cattle market is coded 1. When multiple categories are present (e.g., animal age or sale purpose), simple nominal coding assumes that the distance between categories is the same. For instance, values of 0 (breeding), 1 (milk), and 2 (meat) assume that distance between the three categories is the same – a relationship that is likely to hold only for ordinal data. Hence, data with multiple categories are dummy coded.

ID variables. All variables that identify the owners or the animals are removed since ID variables cause the algorithms to uniquely predict each example and overfit the training data. Since the unique IDs are not repeated in the future (i.e. different customers enter the market), prediction accuracy decreases, and the predictive models do not generalize well to new data. Hence, all ID variables are removed.

kNN algorithm-based model results. The current study predicts whether the sales price of an animal is greater than or less than the mean sale price for animals in that category. The two cattle markets are similar (i.e. animals exchanged are similar) but have slight differences owing to customer differences. It is interesting to assess how the predictive models work on data that is similar but represents different dynamics. By predicting differences between actual markets, the kNN algorithm-based model is being tested.

Since the S Cattle Market is better organized and planned compared to the L Cattle Market, buyers and sellers likely feel more relaxed and empowered in the S Cattle Market. Since buyers can better inspect animals and easily compare with animals offered by other sellers, slightly different buyer behavior dynamics are possible in the S Cattle Market, such as intense bargaining [30,31]. Hence, the kNN algorithm-based model is trained using the L Cattle Market data and tested on the S Cattle Market data. Testing the predictive accuracy of the kNN algorithm-based model on the S Cattle Market data will show how robust the kNN algorithm-based model's predictions are to changing market conditions.

As mentioned before, a rule of thumb to select K (i.e. nearest neighbors who are compared to observation) is to set K equal to the square root of the observations in the training data [21]. Since the L Cattle Market data contains 336 observations, K is set at 18. The kNN algorithm-based model is tabulated using the "knn" function in R studio (package: "class"). The L Cattle Market dataset contains all dummy coded features as well as the class variable (i.e. Price below mean for the animal category/ Price above mean for the animal category). The "knn" function is provided with the L Cattle Market data for training purposes and the S Cattle Market data for testing purposes. The "knn" function will apply learning from the L Cattle Market dataset to the S Cattle Market dataset. However, the S Cattle Market dataset does not contain the class labels (i.e. Price below mean for the animal category/ Price above mean for the animal category), since the kNN algorithm-based model will predict class labels for the S Cattle Market dataset based on learning from the L Cattle Market dataset.

In the final step, the predictive accuracy of the kNN algorithm-based model is assessed. The predicted class values of the kNN algorithm-based model (i.e. Price below mean for the animal category/ Price above mean for the animal category) are compared with the actual class values from the S Cattle Market dataset. A cross-table (package: "gmodel") compares the output between the predicted class values and the actual class values.

Table 5: Predicted vs Actual Class values: kNN algorithm-Based Model

Outcome: Sales price classification (in Local currency) **(0) Price below mean for the animal category** **(1) Price above mean for the animal category** **Total Observations: 606 (S Cattle Market)**			
kNN algorithm-based model predictions			
Actual class values	0	1	Row Total
0	True Negative 350 (88.8 % of row values) (80.6 % of column values) (57.8 % of total values)	False Positive 44 (11.2 % of row values) (25.6 % of column values) (7.3 v% of total values)	394 (65 % of total values)
1	False Negative 84 (39.6 % of row values) (19.4 % of column values) (13.9 % of total values)	True positive 128 (60.4 % of row values) (74.4 % of column values) (21.1 % of total values)	212 (35% of total values)
Column Total	434 (71.6 % of total values)	172 (28.4 % of total values)	606

Table 5 shows the proportions of values that fall into four categories. True negative values represent values (80.6%) for which the "Price below mean for the animal category" is correctly identified by kNN algorithm-based model. Conversely, the true positive represent values (74.1%) for which the "Price above mean for the animal category" is correctly identified by kNN algorithm-based model. However, 19% of the "Price above mean for the animal category" are wrongly classified as "Price below mean for the animal category" (false negative), while 25.6% of "Price below mean for the animal category" are classified as "Price above mean for the animal category" (false positive). In sum, the kNN algorithm-based model is better at predicting "Price below mean for animal category" (80.6%) compared to "Price above mean for the animal category" (74.1%). The total number of prediction mistakes is 21% (i.e. (84+44)/606).

As a whole, the model is quite predictive. If the kNN algorithm-based model has higher predictive accuracy than the Naïve Bayes model, results show that distance-based calculations better capture and predict complex market developments when compared to probability-based models. Hence, complex developments are better captured with simpler models. In the next step, the predictive power of the probability-based model (i.e. the Naïve Bayes model) is assessed.

Model 2

Naïve Bayes. The Bayes Theorem is calculated using the "naiveBayes" function in R studio. In stage 1, the Naïve Bayes model is trained using the L Cattle Market dataset (similar to the kNN algorithm-based model). However, as mentioned previously, to rule out the possibility of non-occurring events in nullifying evidence from occurring events, the Laplace estimator value is set to 1. In the second stage, similar to the kNN algorithm-based model, the S Cattle Market dataset is provided for testing the accuracy of the Naïve Bayes model (without class labels). In the final stage, the predictive accuracy of the Naïve Bayes model is assessed. The predicted class values of the Naïve Bayes model (i.e. Price below mean for the animal category/ Price above mean for the animal category) are compared with the actual class values from the S Cattle Market dataset. A cross-table (Table 6), similar to the one for the kNN algorithm-based model, compares the output between the predicted class values and the actual class values.

Table 6: Predicted vs Actual Class values: Naïve Bayes Model

Outcome: Sales price classification (in Local currency) **(2) Price below mean for the animal category** **(3) Price above mean for the animal category** **Total Observations: 606 (S Cattle Market)**			
Naive Bayes predictions			
Actual class values	0	1	Row Total
0	True Negative 373 (80 % of row values) (94.7 % of column values) (61.6 % of total values)	False Positive 93 (20 % of row values) (43.9 % of column values) (15.3 % of total values)	466 (76.9 % of total values)
1	False Negative 21 15 % of row values) (5.3 % of column values) (3.5 % of total values)	True positive 119 85 % of row values) (56.1 % of column values) (19.6 % of total values)	140 (23.1 % of total values)
Column Total	394(65 % of total values)	212 (35 % of total values)	606

Table 6 shows the proportions of values that fall into four categories. True negative values represent values (94.7 %) for which the "Price below mean for the animal category" is correctly identified by the Naive Bayes model. Conversely, the true positive represent values (56.1 %) for which the "Price above mean for the animal category" is correctly identified by the Naive Bayes model. However, only 5.3 % of the "Price above mean for the animal category" are wrongly classified as "Price below mean for the animal category" (false negative), while 43.9 % of "Price below mean for the animal category" are classified as "Price above mean for the animal category" (false positive). In sum, the Naive Bayes model is better at predicting "Price below mean for the animal category" (94.7 %) compared to "Price above mean for the animal category" (56.1 %). The total number of prediction mistakes is 18.8 % (i.e.21+93/606).

Compared to the kNN algorithm-based model's prediction error rate (21%), the Naïve Bayes model has a smaller error rate (18.8 %). However, wide differences are observed in the error rates. For instance, the kNN algorithm incorrectly classified 19% of the "Price above mean for the animal category" as "Price below mean for the animal category" (false negative) and 25.6% of "Price below mean for the animal category" as "Price above mean for the animal category" (false positive). The proportion of total errors across both categories is almost similar. In contrast, the Naïve Bayes method has a very high error rate (43.9 %) in predicting "Price above mean for the animal category", and very few errors (5.3 %) in predicting "Price below mean for the animal category". Hence, errors are largely prevalent in a single category using the Naïve Bayes method, whereas the errors are similarly distributed across both categories using the kNN algorithm-based model.

Discussion

Results from Model 1 and Model 2 show that both the kNN algorithm-based model and the Naïve Bayes model make approximately 80% correct predictions. However, wide differences are observed in prediction accuracy across the two outcome classes: "Price below mean for the animal category" and "Price above mean for the animal category". The kNN algorithm-based model incorrectly predicts around 20% of values for both the outcome classes. However, the Naïve Bayes model has disproportionately high prediction errors for "Price above mean for the animal category". The question that arises is, is the kNN algorithm-based model more accurate, or is the Naïve Bayes model more accurate? Considering total errors alone shows that the Naïve Bayes model has a smaller error rate (18.8%) compared to the kNN algorithm-based model (21%). However, given the high error rate of the Naïve Bayes in predicting "Price above mean for the animal category", the Naïve Bayes is less dependable in making predictions when market dynamics change.

A simple explanation can unravel the Naïve Bayes model's lack of predicting "Price above mean for the animal category" accurately. The Naïve Bayes model is trained on the L Cattle Market data and tested on the S Cattle Market data. Although both markets trade similar animals (e.g. cows, buffalos, goats, sheep), market dynamics differ due to variations in buyer/seller behaviors. For instance, in one market, buyers aggressively bargain and reduce seller profits, whereas in another market buyers bargain relatively less and allow sellers to charge relatively higher prices. The Naïve Bayes model cannot incorporate information when market dynamics change, even to a small extent. The Naïve Bayes model relies on past data to predict future events. However, if past data does not contain information about certain events, such as buyers bargaining relatively less, the Naïve Bayes Theorem cannot incorporate such information in predictions. Relying on past events to predict future events is a disadvantage when market dynamics change.

On the other hand, the kNN algorithm-based model is better at predicting outcomes. Unlike the Naïve Bayes model, the kNN algorithm-based model does not rely on past events to predict future outcomes. Rather, the kNN algorithm-based model treats features as points on a multi-dimensional space and calculates the distance between the features. The kNN algorithm-based model charts distance between values, finding patterns, and then classifies based on observations that have minimum distances. Since no past event data is used, the kNN algorithm-based method is more flexible in incorporating newer information. Since consumer habits, even though changing, will follow a certain pattern, the kNN algorithm-based method will use patterns to classify observations with higher accuracy compared to the Naïve Bayes model. The ability of the kNN algorithm-based model to accurately predict class values reflects the robustness of the kNN algorithm-based model.

The ability of the kNN algorithm-based model to accurately predict has implications. Firstly, marketing researchers can supplement experimental data results with the kNN algorithm-based models to determine the predictive accuracy of experimental variables across different situations.

For instance, if several experiments are conducted, data from several experiments can be divided into test and train datasets to determine how well variables explain outcomes across different situations. Using the kNN algorithm-based models to validate experimental data findings will help generalize study findings and add robustness to the results. Currently, most marketing research focuses on explaining a single dataset without generalizing the study results. The kNN algorithm-based models can increase the generalizability of results.

Secondly, the kNN algorithm-based models are simple to use and require little effort to tabulate. Marketing researchers will expend minimum effort and benefit from accurate and robust results. Thirdly, adding the kNN algorithm-based models in research results will add diversity in estimation methods. Currently, marketing research heavily relies on the use of probabilistic methods, such as ANOVA and regression [9]. Supplementing marketing research with a non-probabilistic method, such as the kNN algorithm-based model, will increase diversity and compare the results of probabilistic and non-probabilistic methods. Deviations in results reveal interesting insights and point out areas that require greater attention (e.g., prediction accuracy might decrease for certain variables, increasing the need for greater investigation).

In sum, marketing research will greatly benefit by supplementing existing models with the kNN algorithm-based models and producing generalizable results.

Limitation and Future Research

The present study suffers from several limitations. Firstly, the present study considers a Laplace estimator of 1 for the Naïve Bayes Model and K value of 18 for the kNN algorithm-based model. Prediction accuracy can change if the Laplace estimator value or the K value is changed. Hence, future studies can manipulate the Laplace estimator values and the K values to determine which K and Laplace values produce the most accurate predictions. Secondly, the data available for both cattle markets does not contain information about buyer or seller habits or behaviors. Information on buyer or seller habits or behaviors can help establish criteria for different dynamics across the two cattle markets. Although the current data from both the cattle markets are adequate for the current study, future research can include detailed data that measures differences in market dynamics.

Thirdly, the current study compares the predictive accuracy of two commonly used methods, the kNN algorithms, and the Bayes Theorem. Other methods to compare predictive accuracy can be included. For instance, complex machine learning techniques, such as Neural Networks or Support Vector Machines, can be used. Complex machine learning techniques can increase predictive accuracy by dividing algorithms into groups (or nodes) that make independent decisions. However, if the groups (or nodes) make wrong predictions, the algorithm penalizes such groups (or nodes), increasing predictive accuracy. Future research can consider complex machine learning techniques [32,33].

References

1. Reiss, P. C. (2011). Structural workshop paper—descriptive, structural, and experimental empirical methods in marketing research. Marketing Science, 30(6), 950-964.
2. Sunder, S., Kim, K. H., & Yorkston, E. A. (2019). What drives herding behavior in online ratings? The role of rater experience, product portfolio, and diverging opinions. Journal of Marketing, 83(6), 93-112.
3. Tellis, G. J., MacInnis, D. J., Tirunillai, S., & Zhang, Y. (2019). What drives virality (sharing) of online digital content? The critical role of information, emotion, and brand prominence. Journal of Marketing, 83(4), 1-20.
4. Wang, H. S., Noble, C. H., Dahl, D. W., & Park, S. (2019). Successfully communicating a cocreated innovation. Journal of Marketing, 83(4), 38-57.
5. Lever, J., Krzywinski, M., & Altman, N. (2016). Points of significance: model selection and overfitting. Nature methods, 13(9), 703-705.
6. Salkind, N. J. (Ed.). (2010). Encyclopedia of research design (Vol. 1). sage.
7. Hair Jr, J. F., & Sarstedt, M. (2021). Data, measurement, and causal inferences in machine learning: opportunities and challenges for marketing. Journal of Marketing Theory and Practice, 29(1), 65-77.
8. Ma, L., & Sun, B. (2020). Machine learning and AI in marketing–Connecting computing power to human insights. International Journal of Research in Marketing, 37(3), 481-504.
9. Davis, D. F., Golicic, S. L., Boerstler, C. N., Choi, S., & Oh, H. (2013). Does marketing research suffer from methods myopia?. Journal of Business Research, 66(9), 1245-1250.
10. Blair, E., & Zinkhan, G. M. (2006). From the editor: nonresponse and generalizability in academic research. Journal of the Academy of marketing Science, 34(1), 4-7.

11. Deutskens, E., de Jong, A., de Ruyter, K., & Wetzels, M. (2006). Comparing the generalizability of online and mail surveys in cross-national service quality research. Marketing letters, 17(2), 119-136.
12. Srinivasan, R., & Sarial-Abi, G. (2021). When algorithms fail: Consumers' responses to brand harm crises caused by algorithm errors. Journal of Marketing, 85(5), 74-91.
13. Zhang, Z., & Patrick, V. M. (2021). Mickey D's Has More Street Cred Than McDonald's: Consumer Brand Nickname Use Signals Information Authenticity. Journal of Marketing, 85(5), 58-73.
14. Stock, J. H., & Watson, M. W. (2003). Introduction to econometrics. Boston: Addison-Wesley.
15. Malhotra, N. K., & Dash, S. (2010). An applied orientation. Marketing Research, 2.
16. Douglas, S. P., & Craig, C. S. (1997). The changing dynamic of consumer behavior: implications for cross-cultural research. International journal of research in marketing, 14(4), 379-395.
17. Kohli, S., Timelin, B., Fabius, V., & Veranen, S. M. (2020). How COVID-19 is changing consumer behavior–now and forever. McKinsey & Company, 1-2.
18. Chater, N., Tenenbaum, J. B., & Yuille, A. (2006). Probabilistic models of cognition: Conceptual foundations. Trends in cognitive sciences, 10(7), 287-291.
19. Danielsson, P. E. (1980). Euclidean distance mapping. Computer Graphics and image processing, 14(3), 227-248.
20. Liberti, L., Lavor, C., Maculan, N., & Mucherino, A. (2014). Euclidean distance geometry and applications. SIAM review, 56(1), 3-69.
21. Lantz, B. (2019). Machine learning with R: expert techniques for predictive modeling. Packt publishing ltd.
22. Zhang, M. L., & Zhou, Z. H. (2007). ML-KNN: A lazy learning approach to multi-label learning. Pattern recognition, 40(7), 2038-2048.
23. Zhang, S., Li, X., Zong, M., Zhu, X., & Cheng, D. (2017). Learning k for knn classification. ACM Transactions on Intelligent Systems and Technology (TIST), 8(3), 1-19.
24. Imandoust, S. B., & Bolandraftar, M. (2013). Application of k-nearest neighbor (knn) approach for predicting economic events: Theoretical background. International journal of engineering research and applications, 3(5), 605-610.
25. Soofi, A. A., & Awan, A. (2017). Classification techniques in machine learning: applications and issues. Journal of Basic & Applied Sciences, 13, 459-465.
26. Efron, B. (2013). Bayes' theorem in the 21st century. Science, 340(6137), 1177-1178.
27. Ning, B., Junwei, W., & Feng, H. (2019). Spam message classification based on the Naïve Bayes classification algorithm. IAENG International Journal of Computer Science, 46(1), 46-53.
28. Taheri, S., & Mammadov, M. (2013). Learning the naive Bayes classifier with optimization models. International Journal of Applied Mathematics and Computer Science, 23(4), 787-795.
29. Lewis, S. M., & Raftery, A. E. (1997). Estimating Bayes factors via posterior simulation with the Laplace—Metropolis estimator. Journal of the American Statistical Association, 92(438), 648-655.
30. Local Government And Community Development (2015), "Model Cattle Market S", Available At: Model Cattle Market S | Local Government And Community Development (Punjab.Gov.Pk) (Accessed: 6 Sep 2021)
31. Deng, Z., Zhu, X., Cheng, D., Zong, M., & Zhang, S. (2016). Efficient kNN classification algorithm for big data. Neurocomputing, 195, 143-148.
32. Joyce, J. (2003). Bayes' theorem.
33. Metsis, V., Androutsopoulos, I., & Paliouras, G. (2006, July). Spam filtering with naive bayes-which naive bayes?. In CEAS (Vol. 17, pp. 28-69).