Combining Parametric and Non-parametric Methods for Predicting Consumer Choice Using Supermarket Scanner Data

Elena Eneva ELENA.ENEVA@CS.CMU.EDU
Center for Automated Learning and Discovery, Carnegie Mellon University, Pittsburgh, PA 15213 USA

Abstract

It is important for national chain stores to be able to customize their prices in individual stores to adapt to the neighborhood demand. Marketing research indicates that micro-marketing pricing strategies can increase operating profit margins by 33% to 83%. While a lot of stores collect scanner data which can be used to determine the price distribution over products which optimizes profits, often this data resource is under-utilized. In previous research both parametric (such as linear regression) and non-parametric (such as artificial neural networks) models have been successfully used for prediction. We use both parametric and non-parametric methods to investigate ways of combining the lower-level models into a high-level one which takes advantage of prior domain-specific knowledge and is a better predictor than either of the simpler methods by itself. We model the data on Chilled Orange Juices provided by a supermarket chain and provide a valuable tool for marketers concerned with predicting consumer choice. We believe that this work will also generalize to other Machine Learning applications where both parametric and non-parametric methods are currently used.

1. Introduction

In the past years there has been a trend to consolidate independent stores into nation-wide chains, thus taking advantage of the economies of scale, established store name, centralized administration, unified advertising and marketing, among other reasons. However, this trend means that in most cases the prices are determined globally for all stores in the chain, without taking into account the neighborhood differences in demand. Micro-marketing refers to bringing back the adaptability to individual stores by profitably customizing prices at individual store-level. A basis for these customized price strategies are the differences in interbrand competition in different stores. These fluctuations in interbrand competition are measured using weekly store-level scanner data at the product level. Marketing research indicates that micro-marketing pricing strategies can increase gross profit margins by 4% to 10%, which after administrative and operating costs are subtracted would translate into an increase of operating profit margins by 33% to 83% (Montgomery, 1997). Note that these gains come from encouraging consumers, through everyday price changes, to switch to bundles of products which are more profitable for the store, and not through overall price changes at the chain-level (Montgomery, 1997). The methods developed in this paper will provide a valuable tool for marketers concerned with predicting consumer choice and optimizing its profits.

We use weekly store-level scanner data at the product level provided by a local supermarket chain, to construct a predictive model for each store which learns the price elasticity of substitute products within a given category (e.g. different brands and types of chilled orange juice) based on consumer preferences. This model will predict the amounts which will be purchased at any set of prices and thus adjust the prices of the products so as to maximize profits in that category. In previous research, both parametric and non-parametric methods have been used to predict consumer choice. We explore different ways of combining parametric and non-parametric methods which have the effect of biasing the neural network to the model of the other learner (linear regression), which we believe is generally correct. Our results show that the combination methods outperform both of the individual methods and provide models that are more understandable for the marketing community.

2. Prior Work

Montgomery used a hierarchical Bayesian model to investigate pricing strategies from scanner data (Montgomery, 1997). He showed that store differences in demand can be measured and translated into micro-marketing pricing strategies that result in significant expected profit gains for the retailer.

Guadagni and Little use a Logit model of brand choice calibrated on scanner data to estimate price elasticity. They found that consumer response varies across brand sizes in a systematic way (Guadagni and Little, 1983).

West et al. have performed a comparative analysis of neural networks and statistical methods (such as linear regression) for predicting consumer choice. They carried out several experiments and found that in most cases the neural network outperformed the linear regression, and for a small fraction of the tasks the linear regression gave more accurate predictions (West et al., 1997). We will go further and experiment to combine two successful simpler techniques by supplementing the weaknesses of each by the strengths of the other one and create another even more successful higher-level technique.

Rossi and Allenby have performed analysis of scanner data by incorporating prior information through a Bayesian method that yields different parameter estimates for each household. Their research confirmed that household estimates could be used to tailor marketing strategies to specific households. They recommend direct mail coupon drops targeted to price sensitive households to be used instead of traditional blanket mailing (Rossi and Allenby, 1993). This is a similar concept to micro-marketing – if we think of an individual family as all the people who shop at a particular store, it is easy to see that different stores would have customers with different price sensitivity and, therefore, should offer customized prices for each store to optimize profit.

Stacked generalization is a general method of using a high-level model to combine lower-level models to achieve greater predictive accuracy (Ting and Witten, 1999). We will build on some ideas from stacked generalization on improving the predictive accuracy of model combination methods. This will have the effect of biasing the neural network to the model of the other learner (linear regression), which we believe is generally correct

3. Approach

To simplify and focus the problem, we limit our attention to everyday price changes (i.e. the prices of products that are not advertised). This is justified by the importance of everyday pricing in the marketing mix, because most profits are earned on products sold at their everyday price. The supermarket chain providing the data expressed particular interest in being able to price more strategically the various products in the Chilled Orange Juice category, so this is the category that our experiments are conducted on. In previous research, both parametric and non-parametric methods have been used to predict consumer choice. We explore different ways of combining parametric and non-parametric methods to achieve a better predictor than either method by itself. We use a Linear Regression model as a representative of the parametric methods and an Artificial Neural Network for the non-parametric method.

3.1 Price elasticity

Price elasticity measures the proportional change in quantity with respect to a proportional change in price. In our case we are interested in the price elasticity of demand because we want to measure how consumers' demand will respond to a change in price of the goods we are surveying. Price elasticity of demand (E_d) measures the change in quantity demanded (Q_d) with respect to the change in price (P).

$$E_d = \frac{\text{percent change in } Q_d}{\text{percent change in } P}$$

If a decrease of 10% in the price of Tropicana "Lots of Pulp" Orange Juice leads people to buy 15% more of that juice, this means this brand of juice is highly price-elastic and the people are willing to switch from other juices to Tropicana "Lots of Pulp" if it becomes cheaper. On the other hand, maybe it is the case that if the price for Minute Maid Orange Juice decreases by 10%, people will buy only 5% more of that juice. Then this juice would be quite price-inelastic and people would not be as willing to switch to it from other brands even when it gets cheaper. It's important for the stores to know what responses in consumption their price changes would evoke so that they can optimize their profit. However, in the case when close substitutes are present (like Tropicana and Minute Maid) there are other factors which influence the price elasticity of demand – not only the price of a particular product, but also the prices of all the products regarded as its substitutes. It is also important to know exactly how the consumers are changing their preferences in order to estimate the individual product price elasticities, as well as the price elasticity of the whole product category.

Therefore, if we could learn a function which takes in the prices of all the products in a category and outputs the quantities demanded for each product, it can be used to modify prices and encourage people to switch to those products which are most profitable for the store and thus optimize profit. We were provided 2 years' worth of data for the category Chilled Juices, in weekly reports, for 100 stores of a supermarket chain, and are asked to learn such a function for the 14 products in the category.

3.2 Technical Approach – Combining Classifiers

Our task is to learn a function that maps prices to consumption quantities. In previous research both Neural Networks (NN) and Linear Regression (LR) have been used to estimate a similar function. In most cases it was found that NNs outperform the LR, but in a few cases LR was able to predict the results just as well and even slightly better. Typically, different learning algorithms learn different models for the task at hand and to learn an even more accurate function which predicts quantities from prices we would like to take advantage of models generated both by parametric and non-parametric methods. The purpose of this research is to combine the two lower-level models into a higher-level model which achieves greater predictive accuracy than either one by itself.

We combine the two methods in a way that uses the strengths of each method to compensate for the weaknesses of the other and has the effect of biasing the neural network to the linear regression model, which we believe is generally correct The parametric models, such as LR, are based on many assumptions, such as the inherent linear nature of the problem, the normal distribution of errors, etc. They are generally robust and often easier to interpret. The non-parametric methods, such as NNs, on the other hand, operate on quite fewer assumptions and are more flexible, but are often a “black box”. We want the best of both worlds. Knowing that we can model the price elasticity quite closely by the loglinear model, we want our ideal learner to recognize this fact (and in a way, use it as a convenient prior), but we do not want to be absolutely restricted by the linearity assumption. We like the flexibility the NNs offer, but we would also like to have the clarity the LR provides. Therefore, we look for ways of having a combination model which will have the behavior of both the NN and the LR. We look for ways to bias the NN to the linear model and to have it simulate the LR in more flexible ways.

We present different ways of combining the strengths of the NNs and LR. Here we provide description of the methods for constructing a higher-level model which is explored in this research. In the next section we discuss results obtained by using each of the methods and evaluate them.

3.2.1 TRAIN THE CLASSIFIERS SEPARATELY, THEN COMBINE

The first and simplest method is to train a NN and a LR separately and then combine the outputs. The crucial part will be the ways we choose to do the combination of outputs. It is expected that, consistent with the results of previous work, NNs will generally outperform the LR.

Therefore, a simple linear combination with equal weights for the predictions of the two methods will not be justified. We use two other ways of combining the outputs:

<u>Static Weights</u> (proportional weights of the two predictions given their overall success) – the combination of predictions is obtained by a weighted vote from the two methods, deciding how much we should trust each method by taking into account the prediction accuracy of each.

<u>Dynamic Weights</u> (local weights for the regions of the space where each learner does better) – since it is expected that each method performs differently over different parts of the input space, instead of fixing the weighted vote proportion to be the same over the whole space, we let it vary according to the strength of the

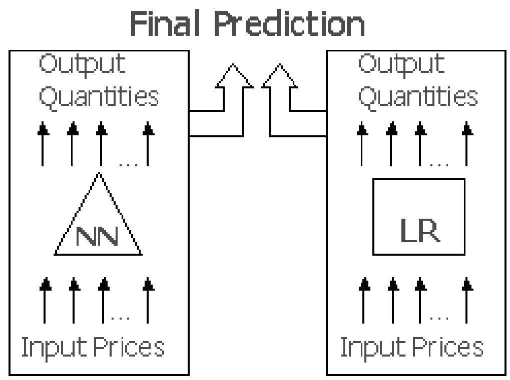

Figure 1. Training the classifiers separately and then combining the predictions.

predictor on that area. The weight proportion is learned by another NN which gets the prices of the products as inputs and outputs the weight that should be given to the prediction of each method.

3.2.2 USE THE PREDICTION OF ONE CLASSIFIER AS AN EXTRA INPUT FEATURE FOR THE OTHER

Another method is to train the NN as above, by giving the product prices of the products as inputs, but this time adding an extra input which is the prediction of the LR over the same prices. Similarly, we train a LR model by giving it the product prices and adding an extra input which is the prediction of the NN over the same prices. In this way, each predictor will have the extra information of what the other one would have predicted and can choose to either incorporate the additional information, “learn from the mistakes” of the other one, or ignore it if it does not prove to be helpful, and thus do at least as well as the better one of the simple models (see Figure 2).

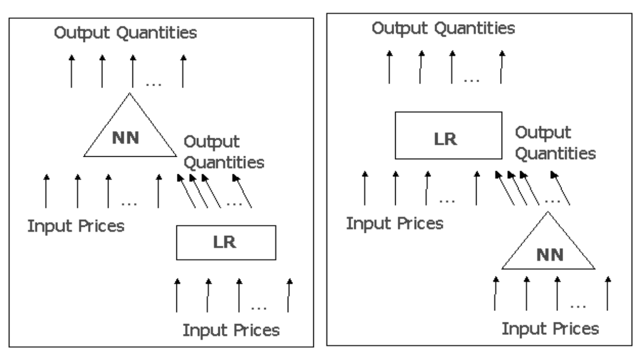

Figure 2. Using the prediction of one classifier as an extra input feature for the other

3.2.3 JUMP CONNECTIONS

Another method is to change the architecture of a standard Neural Net so that the input layer is connected to both the hidden layer and the output layer. Jump connection backpropagation networks are known to work with very complex patterns that are not easily noticeable. We view this as a way to combine LR and a simple NN into a new complex NN which aims at modeling both simpler methods simultaneously. Figure 3 is an illustration of the jump connections architecture, where the lighter lines represent the connections in a standard NN and the darker ones represent the connections corresponding to the regression model.

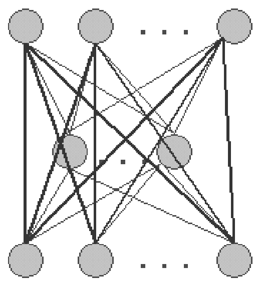

Figure 3. Jump Connections architecture.

4. Evaluation Measure

The purpose of this project is to construct a high-level model using both linear regression and neural networks so as to achieve a better predictor than either of the simpler methods by itself. Therefore, when we evaluate the success of the project we test the models on a holdout set of approximately 20% of the data and determine whether a high-level model with accuracy superior to both of the lower-level models was achieved. The new models are compared among themselves, as well as to the two original simpler models. The performance of each model is assessed using root mean squared error (RMS) technique. The smaller RMS error of a model, the more successful it will be considered.

Root Mean Squared Error

$$RMSerror = \sqrt{\frac{1}{N}\sum_{i-1}^{N}\left\| p_i - \hat{p}_i \right\|^2}$$

RMS error is a common measure of the overall error of an estimator. It is the average deviation between the predicted quantity and the true quantity.

The RMS measures the average error of an estimator. That is, we expect the value of estimator to differ from the value of the parameter by about the square root of the RMS. A smaller RMS would indicate a model which is more accurate.

5. Results and Discussion

Table 1 shows the RMS errors for each of the classifiers on our data set. All of our combination methods perform better than the two original simple methods (NN and LR). From this we can infer that LR and NN are focusing on different aspects of the learning problem and thus result in a better model when combined together. Among the three combination methods, the “Outputs as Inputs” method with the LR output being fed as input to the NN had the least RMS error.

Since the difference in errors among the different models is very small, it is difficult to judge how much improvement we obtain. Therefore, we use another evaluation measure which gives us a better idea about the relative performance of our methods. This measure is calculated by looking at the average percent error (APE) in predicted output quantity for each classifier(absolute difference between predicted quantity and actual quantity) per product and then dividing the APE for every method by the APE for the best method. This way the best method gets the value of 1 and the other values indicate the relative error of the other methods with respect to the best one, averaged over time and over all the products.

Figure 4 shows the results in terms of the new evaluation measure. Now we can see the meaning of the improvement we have attained with our best classifier, “Outputs as Inputs”. For every unit of error which the best classifier makes, the NN error is more than double and the LR error is more than 5 times greater. The other two combination methods, “Jump Connections” and “Weighted Average”, are slightly better than the NN and significantly better than LR.

One way to improve the results even further would be to examine the assumptions that are implicit in our modeling techniques. Some of the assumptions we make are those of independence and stationarity. While there might be some dependence between juices and other products (such as fresh fruit, for example), for simplicity we choose to assume that they are independent. Also, we choose to regard the process as stationary, meaning that we will not account for consumer behavior changing from year to year due to crop factors, different economic conditions, etc. These assumptions are necessary so that we can better

Table 1. Average RMS error

	LINEAR REGRESSION	NEURAL NETWORK	WEIGHTED AVERAGE	JUMP CONNECTIONS	OUTPUTS AS INPUTS
RMS ERROR	0.192	0.075	0.074	0.072	0.070

concentrate on the combination of the parametric and non-parametric models. We believe that they do not affect our results significantly. The challenging problem of removal of these assumptions and tailoring the model to accommodate the above concerns will remain for future work.

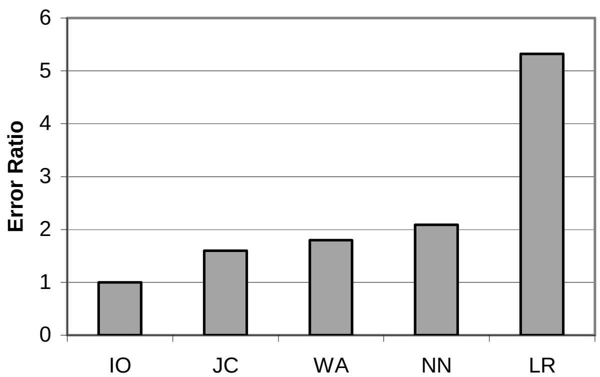

Figure 4. Percent Error Ratio for the different models

6. Future Work

There are several other methods which can be used to combine NN and LR in such a way as to bias the NN towards the regression model. For example, Multitask learning has been explored in depth by Caruana and some very good results have been reported over various domains (Caruana, 1996, 1997). We plan to use multitask learning in a way that the extra task which we will predict in parallel with the output of the neural network will be the output of another learner over the same data, instead of predicting a related problem. This will have the effect of biasing the neural network to the model of the other learner (linear regression), which we believe is generally correct.

If we add the output of the LR model as an extra output to the NN, we will have a model which will attempt to do two tasks simultaneously – predict the quantities purchased from the input prices and also predict the output of the LR model over the same inputs. Learners often learn to use large patterns while ignoring small or less common inputs that are useful. MLT can be used to coerce the learner to attend to a pattern in the input it would otherwise ignore. In a way, the NN will simulate the LR model, as well as perform its usual predictions, and it will attempt to choose the model which both predicts accurately and agrees with the LR model, thus taking advantage of some strengths of the LR model (assuming that LR at least sometimes outperforms the NN) (see Figure 2). MLT has been used successfully for NNs and our expectation is that in our problem it will produce better results than either of the simpler models.

Another way to combine the two types of models would be to train a NN with the residuals of the LR. Linear regression attempts to explain the relationship between two random variables. In our case, one variable is the set of purchase prices of each product and the other one is the set of purchase quantities corresponding to these prices. Regression attempts to explain the relationship with a curve to fit to the data. Each product is modeled as an equation in which its log of quantity demanded is a function of both its own price and the prices of the other products in the category. The regression model states that:

$$\ln(q_i) = a_i + \sum_{i=1}^{N} b_i p_i + e_i,$$

where the subscript i represents the product (in our case, 14 products in the Chilled Juice category), q_i is the quantity demanded for a that product, p_i is the product price, and the "residual" e_i is a random variable with a Normal distribution and mean zero. The coefficients a_i and b_i are determined by the condition that the sum of the square residuals is as small as possible.

We can train a LR on our data and then use the residuals from the regression model as additional inputs to the NN. The goal behind this approach is to let the NN benefit from the model that the LR has constructed. We do this not by giving the NN the final prediction of the LR model, but by providing the residuals as a way for the NN to “gain insight” into the other model and use it if it’s helpful.

Another way of biasing the NN to inherit some of the assumptions and behaviors of the LR is to train it on synthetic data generated by extrapolating from the LR model. Then we could take advantage of the knowledge represented in the LR model and incorporate it in the NN model by training the NN on the synthetic data in addition to the actual data. One way to think of this is to compare it to using unlabeled data for learning. While we are not sure what the correct classification for the unlabeled data instances should be, we can make our best guess (through the LR model, in our case) and treat the prediction as an actual data point. Using unlabeled data has been repeatedly shown to improve the accuracy of the classifiers when they could benefit from additional data. In our case this is not done to address the issue of insufficient data, but to start off the NN in a direction pointed by the LR model, in a way of giving it a prior, and allow it benefit from it, if it chooses to. Some preliminary results we obtained with this method show that this approach is promising and is likely to result in a model which is more accurate than either NN or LR.

7. Conclusions

We approached the problem of estimating consumer choice from a different angle than what had been done before and investigated ways of combining parametric and non-parametric methods into a high-level model so as to achieve a better predictor than either of the simpler methods by itself. We applied our proposed models to the available scanner data and determined that the combined models we developed were more accurate. The best performer was the method which provides the Neural Network with the prediction of the LR, in addition to the usual input.

The new methods we developed are more useful for modeling consumer preference based on prices than the standard simple models currently used in marketing. They are able to more accurately predict the amounts which will be purchased at any set of prices and thus help adjust the prices of the products as to maximize profits.

The supermarket has expressed readiness to apply the results of this study in an experimental effort and implement the proposed pricing strategy in a subset of their stores. It will be very valuable to see our efforts applied directly in practice, since it will give us real-world feedback on the success of our proposed methods and provide us with a chance to assess and improve them.

References

Alpaydin, E. and Gurgen, F. (1998) Comparison of Statistical and Neural Classifier and their Applications to Optical Character Recognition and Speech Classification. Neural Network Systems Techniques and Applications. Academic Press

Bates, J. and Granger, C (1969) The Combination of Forecasts. Operations Research Quarterly, 20,1-68

Caruana, R. (1997) Multitask Learning. Machine Learning, 28, 41-75

Caruana, R. (1996) Algorithms and Applications for Multitask Learning, 13th International Conference on Machine Learning, Bari, Italy

Guadagni, P. and Little, J. (1983) A Logit Model of Brand Choice Calibrated on Scanner data. Marketing Science, 2, 203-238

Gama, J.(1998) Combining Classifiers by Constructive Induction. In Proceedings of the 10th European Conference on Machine Learning (ECML'98), 178-189

Granger, C. (1989) Invited review; Combining Forecasts - Twenty Years Later. Journal of Forecasting, 8

Hashem, S. (1993) Optimal Linear Combinations of Neural Networks. PhD thesis, Purdue University

Ho, T. (1997) Adaptive coordination of multiple classifiers. Document Analysis Systems II, World Scientific Publishing, 371-384

Ho, T., Hull, J. and Srihari, S. (1994) Decision combination in multiple classifier systems. IEEE Transactions on Pattern Analysis and Machine Intelligence, 16, 66-75

Jacobs, R. (1995) Methods for combining experts' probability assessments. Neural Computation, 7, 867-888

Kittler, J., Hatef, M., Duin, R. and Matas, J. (1998) On combining classifiers. IEEE Transactions on Pattern Analysis and Machine Intelligence, 20, 226-239

Krogh, A. and Sollich, P. (1997) Statistical mechanics of ensemble learning. Physical Review, 55

Merz, C. (1998) Classification and Regression by combining Models. PhD thesis, University of California, Irvine

Merz, C. and Pazzani, M. (1997) Combining neural network regression estimates with regularized linear weights. Advances in Neural Information Processing Systems 9, 564-570. MIT Press

Mojirsheibani, M. (1999). Combining classifiers via discretization. Journal of the American Statistical Association, 94, 600-609

Montgomery, A. (1997). Creating Micro-Marketing Pricing Strategies Using Supermarket Scanner Data. Marketing Science, 16, 315-337

Perrone, M. (1993) Improving Regression Estimation: Averaging Methods for Variance Reduction with Extensions to General Convex Measure Optimization. PhD thesis, Brown University

Rossi, P. and Allenby, G. (1993) A Bayesian Approach to Estimating Household Parameters. Journal of Marketing Research, 30, 171-182

Ting, K. and Witten, I. (1997) Stacked generalization: when does it work. In Proceedings of the Fifteenth International Conference on Artificial Intelligence (ICAI '97), 866-871

Ting, K. and Witten, I. (1999). Issues in Stacked Generalization. Journal of Artificial Intelligence Research, 10, 271-289

Tresp, V. and Taniguchi, M. (1995) Combining estimators using non-constant weighting functions. Proceedings of the Neural Information Processing Systems 7. MIT Press

West, P., Brockett, P. and Golden, L (1997) A Comparative Analysis of Neural Networks and Statistical Methods for Predicting Consumer Choice. Marketing Science, 16, 370-391

Xu, L., Krzyzak, A. and Suen, C. (1992). Methods of combining multiple classifiers and their applications to handwriting recognition. IEEE Transaction on Systems, Man, and Cybernetics, Vol. 22, 3, 418-435