Enhancing a Financial Service organization's cross-sell strategy using Artificial Neural Networks

Vishal Krishna, Vaishak Salin, Dhawal Joharapurkar
Department of Computer Science and Engineering, Manipal University, Manipal, Karnataka
Email:vishal.krishna@learner.manipal.edu, vaishak.salin@learner.manipal.edu, dhawal.manoj@learner.manipal.edu

***Abstract*—The cost of acquiring a new customer is upwards of four times the cost of serving an additional product to an existing customer, so it makes sense for a financial service organization to have a sound cross-sell strategy. We have built a model to predict the likelihood of an existing customer accepting an offer of a term deposit. The data is related with direct marketing campaigns of a Portuguese banking institution which were based on phone calls. Often, more than one contact to the same client was required, in order to assess if the product (bank term deposit) would be (or not) subscribed. We pre-processed the data by converting categorical data into numeric data and performed data binning on few features to reduce the effect of minor observation errors. We then trained a feedforward neural network, which is an artificial neural network where connections between the units do not form a directed cycle, using backpropagation. Finally, we used cross-validation to adjust the number of hidden layers and hidden units and to resolve the bias-variance trade-off to fine tune our model.**

I. Introduction

Cross-selling is the practice of selling an additional product or service to an existing customer. It ranks as a top strategic priority for many industries including financial services, insurance, health care, accounting, telecommunications, airlines, and retailing. Despite the increasing investment in cross-selling programs, firms find that these million-dollar marketing campaigns are not profitable. The average response rate as measured by a customer purchase within three months after a cross-selling campaign is about 2 per cent. A managerial challenge is to improve the response rates of a cross-selling campaign while avoiding the targeting of customers with irrelevant messages. Cross-selling offers several advantages. Except for the obvious from the extra products sold, it also increases the dependence of the customer on the vendor and therefore reduces churn.

Most current cross-selling campaigns are designed with this orientation: lets find the customers who are most likely to respond. Firms begin cross-selling campaigns by setting a time schedule (e.g., mail the promotional material in one month) and then select a communication channel (e.g., phone, email, or mail) for this campaign. Analysts then develop a customer-response model with the purchase decision as a dependent variable and product ownership and customer demographics as explanatory variables. Finally, upon estimation of the customer-response model, the expected profit is computed, and firms schedule all customers with positive expected profits to receive the promotion. If the firm has to heed a budget constraint, it will only solicit the most profitable customers. We refer to this process as campaign-oriented cross-selling.

We argue that an improved customer-centric orientation for cross-selling is: how do we introduce the right product to the right customer at the right time using the right communication channel to ensure long-term success. Conceptually, customer demand for financial services depends upon the customers evolving financial maturity. Thus, each individual customers preferences and responsiveness to cross-selling solicitations may change over time and the marketer has to track and anticipate these changes. In addition, cross-selling solicitations may provide more than just a promotional incentive that immediately stimulates purchase. Cross-selling can create enduring relationships between a customer and the firm by serving as a general advertisement for the brand, a signal of quality, and to educate consumers about the scope of product offerings and how various products meet their long-term financial needs. Ultimately this requires the marketer to have a long term view and generate dynamic solicitations in accordance with the customers evolving financial status and preferences in order to maximize the long-term financial payoff. Integral to the above is a sound cross-sell analytics model. While getting access to global customer master may not be difficult, the challenge is to determine whom to contact, with what kind of offers, and at what intervals. Also we will examine the impact of false positives and true negatives - getting a customer irritated about being contacted, or leaving out a potentially valuable customer. Our ambition in mind would be to find out which is worse and what the model should prefer

II. Data

The data pertains to direct marketing - tele-marketing campaigns of a Portuguese banking institution. It is conceivable that more than one contact to the same client is required in order to close the Term Deposit conversation. The data is from a public data source. We randomly divide the data (42113 rows) into 3 parts, 60% training data, 20% cross-validation and 20% test data. We have created a data dictionary using which we transformed our categorical data into numeric data. Data dictionary is as shown in table I.

This data set is converted to .arff file for quick prototyping and visualization in weka by attaching attribute descriptions to each of the data columns. Following visualization will help us better in understanding the problem. Attributes pertaining to those customers who subscribed a term deposit is shown in 'red', and while those who didn't are shown in 'blue'

TABLE I. VARIABLE DESCRIPTION

Variable Name	Description	Code	Meaning
Person id	Unique Customer id	NA	
Age	Age of customer	NA	
Job_Type	The nature of job of the customer	NA	
		1	management
		2	technician
		3	entrepreneur
		4	blue-collar
		5	unknown
		6	retired
		7	admin
		8	services
		9	self-employed
		10	unemployed
		11	housemaid
		12	student
Marital_Status	Marital status of the customer	1	married
		2	single
		3	divorced
Education	Level of education of the customer	1	tertiary
		2	secondary
		3	unknown
		4	primary
Default	This variable defines if the customer has defaulted anytime earlier	1	Yes
		2	No
Balance	Average yearly balance	NA	
Housing_loan	This variable defines if the customer has already taken a housing loan	1	Yes
		2	No
Personal_loan	This variable defines if the customer has already taken a personal loan	1	Yes
		2	No
Day	Last contact day of Month	NA	
Month	Last contact Month of Month	1	jan
		2	feb
		3	mar
		4	apr
		5	may
		6	jun
		7	jul
		8	aug
		9	sep
		10	oct
		11	nov
		12	dec
Duration	Last contact duration in seconds	NA	
Campaign	Number of contacts performed during this campaign and for this client	NA	
Pdays	Number of days that passed by after the client was last contacted		
Previous	Number of contacts performed before this campaign and for this client	NA	
Poutcome	Outcome of the previous marketing campaign made to the customer	1	unknown
		2	failure
		3	other
		4	success
Converted	Has the customer subscribed a term deposit	1	Yes
		0	No

III. METHODOLOGY

1) Data Pre-processing
   a) Compute data dictionary
   b) Perform data binning
2) ANN
   a) Perform feedforward
   b) Compute cost and error
   c) Perform backpropagation
3) Results
   a) Calculate accuracy on the training set
   b) Perform cross-validation
   c) Perform bias-variance calculation
4) Tuning
   a) Adjustment of number of hidden layers and hidden units
   b) Final output model

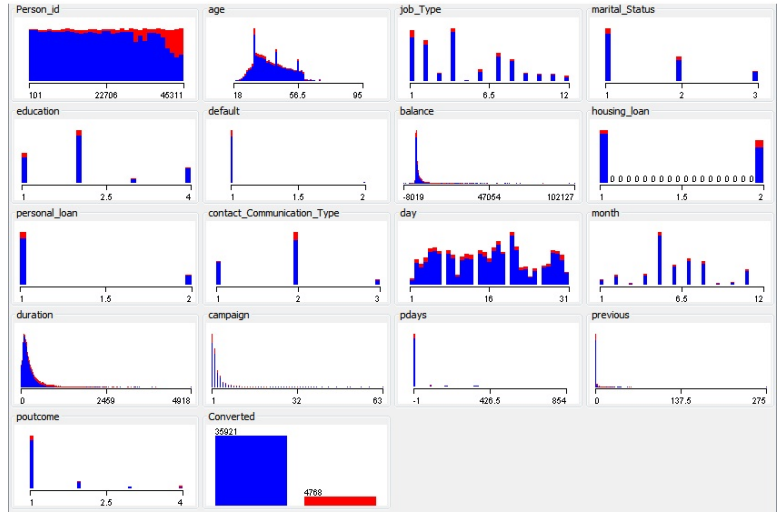

Fig. 1. Data Visualization

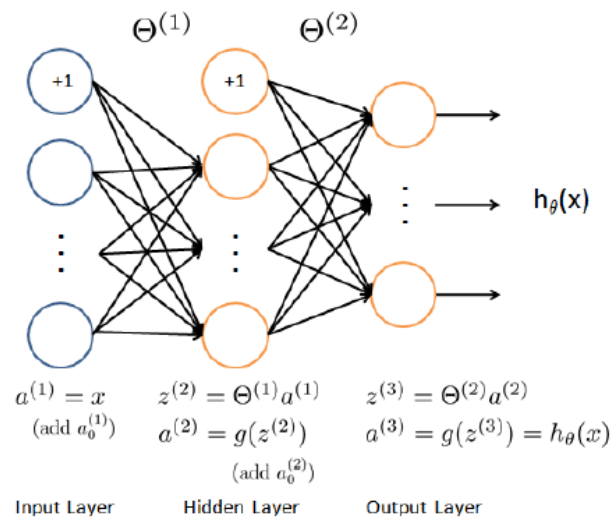

Fig. 2. Neural Network

A. Pre-processing

We pre-process the text by converting the categorical data from the data dictionary to numerical data. We continue pre-processing by performing data binning to group the data based on attributes like age where we replace the original data values which fall in a given small interval, known as a bin, and are replaced by a value representative of that interval. It is a form of quantization.

B. Model: Artificial Neural Network

If a multilayer perceptron has a linear activation function in all neurons, that is, a linear function that maps the weighted inputs to the output of each neuron, then it is easily proved with linear algebra that any number of layers can be reduced to the standard two-layer input-output model. What makes a multilayer perceptron different is that each neuron uses a nonlinear activation function which was developed to model the frequency of action potentials, or firing, of biological

neurons in the brain. This function is modeled in several ways, but must always be normalizable and differentiable. The two main activation functions used in current applications are both sigmoids, and are described by

$$\phi(v_i) = \tanh(v_i) and \phi(v_i) = (1 + e^{-v_i})^{-1},$$

in which the former function is a hyperbolic tangent which ranges from -1 to 1, and the latter, the logistic function, is similar in shape but ranges from 0 to 1. Here y_i is the output of the ith node (neuron) and v_i is the weighted sum of the input synapses. Alternative activation functions have been proposed, including the rectifier and softplus functions. To compute each element in the summation, we have to compute hypothesis for every example *i*, where hypothesis is equal to the sigmoid function. It turns out that we can compute this quickly for all our examples by using matrix multiplication. We then proceed to use feedforward propagation to a neural network, compute th to train the classifier on the data

Our multilayer perceptron consists of four layers (an input and an output layer with two hidden layers) of nonlinearly-activating nodes. Each node in one layer connects with a certain weight w_{ij} to every node in the following layer.

Learning occurs in the perceptron by changing connection weights after each piece of data is processed, based on the amount of error in the output compared to the expected result. This is an example of supervised learning, and is carried out through backpropagation, a generalization of the least mean squares algorithm in the linear perceptron. We represent the error in output node j in the nth data point by $e_j(n) = d_j(n) - y_j(n)$, where d is the target value and y is the value produced by the perceptron. We then make corrections to the weights of the nodes based on those corrections which minimize the error in the entire output, given by

$$\mathcal{E}(n) = \frac{1}{2}\sum_j e_j^2(n)$$

. Using gradient descent, we find our change in each weight to be

$$\Delta w_{ji}(n) = -\eta \frac{\partial \mathcal{E}(n)}{\partial v_j(n)} y_i(n)$$

where y_i is the output of the previous neuron and η is the learning rate, which is carefully selected to ensure that the weights converge to a response fast enough, without producing oscillations. The derivative to be calculated depends on the induced local field v_j, which itself varies. It is easy to prove that for an output node this derivative can be simplified to

$$-\frac{\partial \mathcal{E}(n)}{\partial v_j(n)} = e_j(n)\phi'(v_j(n))$$

where ϕ' is the derivative of the activation function described above, which itself does not vary. The analysis is more difficult for the change in weights to a hidden node, but it can be shown that the relevant derivative is $-\frac{\partial \mathcal{E}(n)}{\partial v_j(n)} = \phi'(v_j(n)) \sum_k -\frac{\partial \mathcal{E}(n)}{\partial v_k(n)} w_{kj}(n)$. This depends on the change in weights of the kth nodes, which represent the output layer. So to change the hidden layer weights, we must first change the output layer weights according to the derivative of the activation function, and so this algorithm represents a backpropagation of the activation function.

TABLE II. SUMMARY

Correctly Classified Instances	37076	91.1205 %
Incorrectly Classified Instances	3613	8.8795 %
Kappa statistic	0.5289	
Mean absolute error	0.1197	
Root mean squared error	0.248	
Relative absolute error	57.838 %	
Root relative squared error	77.1122 %	
Coverage of cases (0.95 level)	99.2627 %	
Mean rel. region size (0.95 level)	65.7438 %	
Total Number of Instances	40689	

C. Tuning based on analysis

In machine learning, the bias-variance dilemma or bias-variance tradeoff is the problem of simultaneously minimizing the bias (how accurate a model is across different training sets) and variance of the model error (how sensitive the model is to small changes in training set). This tradeoff applies to all forms of supervised learning: classification, function fitting, and structured output learning. The bias-variance tradeoff is a central problem in supervised learning. Intuitively, it means that a model must be chosen that at the same time captures the regularities in its training data, but also generalizes well to unseen data. Models can have high bias, meaning they impose restrictions on the kind of regularities that can be learned, or they can have high variance, meaning they can learn many kinds of complex regularitiesincluding noise in the training data. To achieve good performance on data outside the training set, a tradeoff must be made.

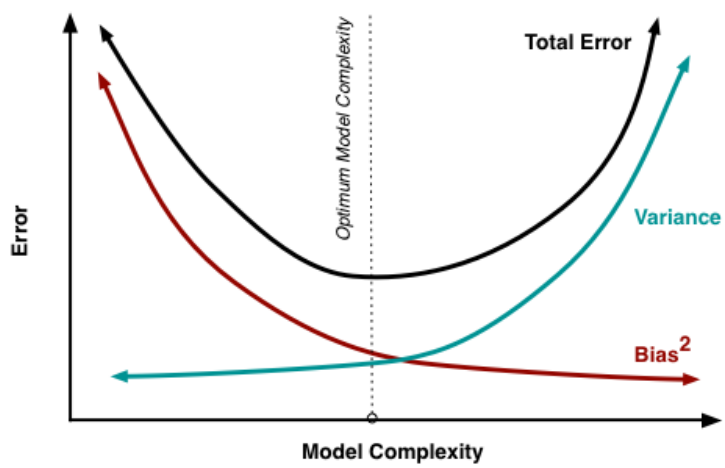

Fig. 3. Bias Variance Trade-off

IV. RESULTS

The initial logistic regression model performed with an error rate of 0.3345, which is slightly better than the bench mark classifier. The artificial neural network model however performed with an error rate of 0.2347 which is a considerable improvement over the bench mark. The final ANN model used 26 hidden units in each layer and 2 hidden layers which was computed by resolving the bias and variance over 400 iterations.

TABLE III. DETAILED ACCURACY BY CLASS

	TP	FP	Precision	Recall	F1	MCC	ROC	PRC	Class
	0.963	0.482	0.938	0.963	0.950	0.533	0.928	0.988	0
	0.518	0.037	0.652	0.518	0.578	0.533	0.928	0.627	1
Avg	0.911	0.429	0.904	0.911	0.907	0.533	0.928	0.946	

TABLE IV. CONFUSION MATRIX

	0	1
0	34604	1317
1	2296	2472

V. CONCLUSION

Low response rates are challenging managers to improve the effectiveness of cross-selling campaigns. We believe current cross-selling focuses too much on individual campaigns and not enough on the dynamic effects inherent in a customer-centric approach. We find that cross-selling campaigns can be improved by understanding how cross-selling solicitations change customer purchase behavior and tailoring these campaigns to each customer's evolving needs and preferences in order to enhance long-term customer relationships and optimize long-term profits

Using cross-selling campaigns and purchase history data provided by the bank, we propose and estimate a customer-response model that recognizes latent financial maturity and preference for communication channels. Our framework allows cross-selling solicitations to influence the customer's latent financial state so that they may become more receptive to particular products in the future.

It should be noted that our method does not provide model which are understandable to humans. This is the case even for decision trees, where large size of the tree and potential variable correlations make it difficult to understand the underlying causal structure. Classifier models can thus be useful for selecting customers and services which should be targeted, but not to explain why particular customers prefer particular services. Such knowledge could of course result in a better marketing campaign.

The Bayesian networks based method offers lower accuracy but gives full insight into dependencies between attributes in the data. While building the network we "learn" the data, and eventually get a model describing not just the correlations, but also causal relationships between all variables. We can thus understand how changing one of the parameters will influence probability distributions of other parameters.

The first direction of future research will be improving the Bayesian network implementation such that conditional probability distributions can be represented using classifiers. This should allow the Bayesian network method to achieve accuracy comparable with classifier based methods.

In a longer perspective it would be interesting to create a model which would describe general aspects of customer behaviour. It would thus become possible to predict the demand for a service before it was even rolled out to the market. A Bayesian network could form a basis of such model. It would also be useful to couple such a model with customer's lifetime value prediction module.

REFERENCES

[1] A. M. Wedel, F. de Rosa, and J. A. Mazzon. 2003. Cross-Selling through Database Marketing: A Mixed Data Factor Analyzer for Data Augmentation and Prediction. International Journal of Marketing Research. 20(1): 45-65

[2] R. Agrawal, T. Imielinski, and A. Swami, Mining association rules between sets of items in large databases, in ACM SIGMOD Conf. Manage. Data, Washington, USA, 1993, pp. 207216

[3] J. Pearl, Probabilistic Reasoning in Intelligent Systems. Los Altos: Morgan Kaufmann, 1998

[4] S. Jaroszewicz and T. Scheffer, Fast discovery of unexpected patterns in data, relative to a Bayesian network, in 11th ACM SIGKDD Int. Conf. Knowl. Discov. Data Min. KDD 2005, Chicago, USA, 2005, pp. 118127

[5] Fukushima, K. (1980). Neocognitron: A self-organizing neural network model for a mechanism of pattern recognition unaffected by shift in position. Biological Cybernetics 36 (4): 93202

[6] J.K. Anlauf and M. Biehl. The AdaTron: an Adaptive Perceptron algorithm. Europhysics Letters 10: 687-692 (1989)

[7] Small explanation of binning in image processing. Steve Cannistra. Retrieved 2011-01-18

[8] Kohavi, Ron (1995). A study of cross-validation and bootstrap for accuracy estimation and model selection. Proceedings of the Fourteenth International Joint Conference on Artificial Intelligence 2 (12): 11371143.(Morgan Kaufmann, San Mateo, CA)