High-Quantile Modeling for Customer Wallet Estimation and Other Applications

Claudia Perlich, Saharon Rosset,
Rick Lawrence,
IBM T.J. Watson Research Center
P. O. Box 218
Yorktown Heights, NY 10598
{srosset, perlich, ricklawr}@us.ibm.com

Bianca Zadrozny
Instituto de Computaçao
Universidade Federal Fluminense
Rua Passo da Pátria, 156
Niterói, RJ, Brazil, 24210-240
bianca@ic.uff.br

ABSTRACT

In this paper we discuss the important practical problem of *customer wallet estimation*, i.e., estimation of potential spending by customers (rather than their expected spending). For this purpose we utilize quantile modeling, whose goal is to estimate a quantile of the discriminative conditional distribution of the response, rather than the mean, which is the implicit goal of most standard regression approaches. We argue that a notion of wallet can be captured through high quantile modeling (e.g, estimating the 90th percentile), and describe a wallet estimation implementation within IBM's Market Alignment Program (MAP). We also discuss the wide range of domains where high-quantile modeling can be practically important: estimating opportunities in sales and marketing domains, defining 'surprising' patterns for outlier and fraud detection and more. We survey some existing approaches for quantile modeling, and propose adaptations of nearest-neighbor and regression-tree approaches to quantile modeling. We demonstrate the various models' performance in high quantile estimation in several domains, including our motivating problem of estimating the 'realistic' IT wallets of IBM customers.

Keywords

Quantile Estimation, Quantile Loss, Regression Trees, kNN

1. INTRODUCTION

In standard regression modeling, we are given n observations on a continuous numeric variable Y and a set of p explanatory variables, or features, $\mathbf{x} = (x_1, ..., x_p)^t$, and we try to estimate the 'dependence' of Y on $\mathbf{x}$, so that in the future we can observe $\mathbf{x}$ only and predict what Y may be. This typically leads us to build a model for a conditional central tendency of $Y|\mathbf{x}$, usually the mean $E(Y|\mathbf{x})$. For example, under appropriate model assumptions, modeling based on a least squares loss function (like linear least squares or most regression tree approaches), is as a maximum likelihood approach to estimating this conditional mean.

In this paper we address the situations when we are really not interested in estimating a conditional mean, but rather a different property of the conditional distribution $P(Y|\mathbf{x})$, in particular a high quantile of this distribution, such as the 0.9 quantile of $P(Y|\mathbf{x})$, which is the function $c(\mathbf{x})$ such that $P(Y \leq c(\mathbf{x})|\mathbf{x}) = 0.0$. As we discuss in Section 2 below, these problems (of estimating conditional mean vs. conditional high quantile) may be equivalent under certain simplistic assumptions about our models, but in practice they are usually not. We are typically interested in modeling high quantiles because they represent a desired 'prediction' in some business and scientific domains.

Our primary motivating application is the problem of customer wallet estimation, which is of great practical interest to us at IBM. A customer's wallet for a specific product category (for example, Information Technology) is the total amount this customer *can* spend in this product category. As an IT vendor, IBM observes what the companies that are our customers *actually* spend with us, but does not typically have access to the customers' budget allocation decisions, their spending with competitors, etc. Information about our customers' wallet, as an indicator of their potential for growth, is considered extremely valuable for marketing, resource planning and other tasks. For a detailed survey of the motivation, problem definition, and some alternative solution approaches, see [18]. In that paper we propose the definition of a customer's $REALISTIC$ wallet as the 0.9 or 0.95 quantile of their conditional spending — this can be interpreted as a highly optimistic (yet still attainable) estimate of what they could spend on buying IT from IBM. This task of modeling 'what we can hope for' rather than 'what we should expect' turns out to be of great interest in multiple other business domains, including:

- When modeling sales prices of houses, cars or any other product, the seller may be very interested in the price they may aspire to get for their asset if they are successful in negotiations. This is clearly different from the 'average' price for this asset and is more in line with a high quantile of the price distribution of equivalent assets. Similarly, the buyer may be interested in the symmetric problem of modeling a low quantile.
- In outlier and fraud detection applications we may

KDD'07, August 12 – 15, 2007, San Jose, California, USA.

have a specific variable (such as total amount spent on a credit card) whose degree of 'outlyingness' we want to examine for each one of a set of customers or observations. This degree can often be well approximated by the quantile of the conditional spending distribution given the customer's attributes. For identifying outliers we may just want to compare the actual spending to an appropriate high quantile, say 0.95.

In this paper, we address quantile estimation both as a generic problem and in the specific context of customer wallet estimation. In Section 2 we discuss the fundamental statistical and practical issues involved in the task of modeling high quantiles and evaluating performance of high-quantile prediction models. We then survey in Section 3 a variety of approaches that have been proposed in the literature for quantile modeling, and propose original approaches based on the adaptation of arguably the two most common regression approaches used in data mining — k-nearest neighbors and regression trees — to estimation of high quantiles instead of conditional means. Section 4 is devoted to a case study, where we describe the Market Alignment Program (MAP), aimed at refocusing IBM Sales resources using wallet estimates, and present the modeling and evaluation process we went through to determine which approach was most appropriate for supplying wallet estimates for MAP. Our main tool in that effort is wallet values obtained from consultation with IBM experts, which we use to analyze which of a large set of candidate models best captures the experts' notion of customer wallet. Finally, we conduct and present in Section 5 an extensive experimental study on a number of practical prediction problems where high quantile prediction is a well justified prediction task. This includes our motivating application of customer wallet estimation, as well as several publicly available datasets.

Our main conclusions are:

- The IBM experts' notion of customer wallet seems most consistent with a high-quantile model for the 0.8 quantile. Compared to our previously proposed definition of REALISTIC wallet as the 0.9 quantile [18], it seems that the experts (who are IBM sales executives) are a little more cautious.
- High quantile predictions are meaningfully different from mean predictions in all the domains we experiment with here.
- Most of the quantile modeling methods we consider turn out to be useful, in the sense that most of them do very well on at least one of our experimental domains, and all of them do consistently better than standard modeling approaches.

We discuss these conclusions in more detail in Section 6.

2. BUILDING AND EVALUATING QUANTILE PREDICTION MODELS

In this section we review some of the fundamental statistical and algorithmic concepts underlying the two main phases of predictive modeling — model building and model evaluation and selection — when our goal is ultimately to predict high quantiles well.

Let us start from the easier question of model evaluation and model selection: given several models for predicting high quantiles and an evaluation data set not used for modeling, how can we estimate their performance and choose among them? The first, obvious answer, would be to obtain observations about high quantiles of holdout data (for example, by asking experts). This is in general a difficult or very expensive endeavor, and is often simply impossible (because no experts are available who can estimate this).

A more sound approach to this problem is to find a loss function which describes well our success in predicting high quantile and evaluate the performance using this loss function. Clearly, the most important requirement from a loss function for evaluation is that the model which always predicts the conditional quantile correctly will have the best expected performance. Such a loss function indeed exists [11]. Define the quantile loss function for the pth quantile to be:

$$L_p(y,\hat{y}) = \begin{cases} p\cdot(y-\hat{y}) & \text{if } y \geq \hat{y} \\ (1-p)\cdot(\hat{y}-y) & \text{otherwise} \end{cases} \tag{1}$$

In Figure 1, we plot the quantile loss function for $p \in \{0.2, 0.5, 0.8\}$. With $p = 0.5$ this is just absolute error loss. Expected quantile loss is minimized by correctly predicting the (conditional) pth quantile of the conditional distribution. That is, if we fix a prediction point $\mathbf{x}$, and define $c_p(\mathbf{x})$ to be the pth quantile of the conditional distribution of Y given $\mathbf{x}$:

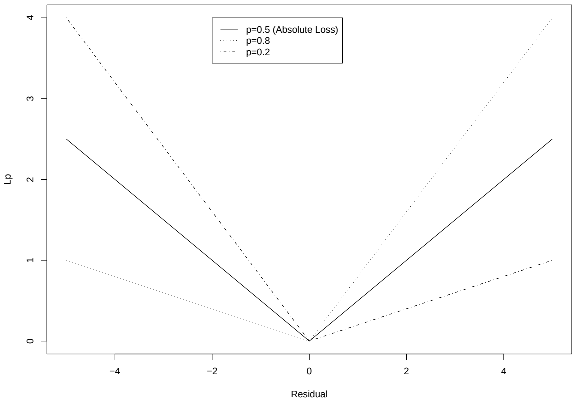

Figure 1: Quantile loss functions for some quantiles.

$$P(Y \leq c_p(\mathbf{x})|\mathbf{x}) = p\ ,\ \forall \mathbf{x}$$

then the loss function is optimized in expectation *at every point* by correctly predicting $c_p(\mathbf{x})$:

$$\arg\min_c E(L_p(Y,c)|\mathbf{x}) = c_p(\mathbf{x})$$

With $p = 0.5$, the expected absolute loss is minimized by predicting the median, while when $p = 0.9$ we are in fact evaluating a model's ability to correctly predict the 90th percentile of the distribution $P(Y|\mathbf{x})$.

Another approach to evaluation is to look at the proportion of positive and negative residuals on the holdout data. A perfect prediction model for the 0.9 quantile will predict a value that is higher than the actual observed holdout response 90% of the time, on average. Thus we can examine whether the actual percentage of the time that the predictions are higher than observed response is indeed close to that, as a way of 'evaluating' high-quantile models. This is dangerous, of course, because a model which predicts $+\infty$ 90% of the time and $-\infty$ the other 10% would be perfect according to this measure.

On the modeling side, our first observation is that any property of the conditional distribution $P(Y|\mathbf{x})$ can be estimated well if we estimate well the whole distribution. In particular, if we have a parametric model for $P(Y|\mathbf{x})$ which we believe is true and which we have enough data to estimate, then it is often the best policy to apply all our effort towards estimating this model's parameters well (e.g., using a maximum likelihood approach), regardless of what property of $P(Y|\mathbf{x})$ we are ultimately interested in. For example, if we believe that the distribution of $Y|\mathbf{x}$ is homoscedastic gaussian and $E(Y|\mathbf{x}) = \alpha^t\mathbf{x}$ is linear in $\mathbf{x}$, then a maximum likelihood approach would call for fitting a linear regression model of Y on $\mathbf{x}$. Furthermore, this would also trivially imply that the 0.9 quantile of $P(Y|\mathbf{x})$ is linear in $\mathbf{x}$, and is simply a fixed offset from the expectation: $E(Y|\mathbf{x}) + 1.28 \times \sigma$.

However, parametric and distributional assumptions are usually over simplifications of realistic modeling problems, especially those encountered in complex data-mining domains, and one should either dispose with them completely (and choose non parametric approaches), or at least treat with skepticism the notion of high quality estimation of complete conditional distributions. An alternative is to build the model by minimizing an 'empirical risk' over the training data, which represents well the prediction task. In the case of quantile modeling, the quantile estimation loss function L_p (1) certainly qualifies (a similar approach leads Friedman et al. [7] to advocate the logistic regression loss function for boosting, for example).

In practice, both of these approaches may have advantages and disadvantages. An additional consideration is one of variance, especially when modeling high quantiles — does the high-quantile loss function allow us to make efficient use of the data for modeling? See [11] for detailed discussion of the dependence of variance on quantile for simple quantile regression.

All of this leads us to adopting the following methodological guidelines in developing and testing high-quantile estimation modeling approaches:

1. Where available, we would like to use expert-supplied 'ground truth' for evaluation. Thus, in Section 4 we use the wallet estimates supplied by experts as a noisy target, and evaluate our model's success in terms of its success in predicting numbers that are close to this target. When such expert input is not available, our hold-out data evaluation is based primarily on the appropriate high-quantile loss function L_p (1).
2. Following the concept of 'empirical risk minimization' we expect that adaptation of existing learning methods to use the quantile loss function may lead to useful quantile modeling tools.
3. We believe these approaches should still be compared to 'standard' learning approaches in terms of the high-quantile performance of the resulting models, and that general purpose prediction models may still occasionally do better on the high-quantile prediction task than the models based on use of quantile loss function, due to the statistical reasons discussed above. In all of our experiments below, however, we observe that the standard approaches performance is significantly inferior to that of quantile estimation models.

3. ADJUSTING MODELING APPROACHES TO QUANTILE PREDICTION

Over the recent past, the modeling of quantiles has received increasing attention. The modeling objectives were either prediction or to gain insights how the statistical dependencies for quantiles differ from expected value models. We review some of these efforts in 3.1. Two of the best studied and also practically most common standard regression approaches in machine learning and data mining are k-nearest neighbors and regression trees. We discuss in some detail how these methods can be adjusted to modeling quantiles in 3.2 and 3.3.

3.1 Existing Approaches

We are aware of a number of such quantile estimation methods including Linear quantile regression [11], Kernel quantile regression [19], Quanting [12], Quantile regression forests [14] and polynomial regression trees [6]. Many of these methods suffer from intractable computational behavior for larger quantile estimation tasks as in the case of our IBM wallet. We next discuss the two most relevant approaches to our work in some detail.

Linear quantile regression
A standard technique for quantile regression that has been developed and extensively applied in the Econometrics community is linear quantile regression [11]. Linear quantile regression assumes that the conditional quantile function is a linear function of the explanatory variables of the form $\beta\mathbf{x}$ and we estimate the parameters $\hat{\beta}$ that minimize the quantile loss function (Equation 1). It can be shown that this minimization is a linear programming problem and that it can be efficiently solved using interior point techniques [11]. Implementations of linear quantile regression are available in standard statistical analysis packages such as R and SAS. The obvious limitation of linear quantile regression is that the assumption of a linear relationship between the explanatory variables and the conditional quantile function may not be true. To circumvent this problem, Koenker [11] suggests using nonlinear spline models of the explanatory variables.

Quanting
Recently, a reduction from quantile regression to classification has been proposed [12]. The *Quanting* reduction trans-

forms a quantile regression problem into a series of classification problems such that a small average error rate on the classification problems leads to a provably accurate estimate of the conditional quantile. This allows us to apply any existing classifier learning algorithm to solve quantile regression problems. The essential idea of quanting is that each classifier c_t attempts to answer the question 'is the q-quantile above or below t?' In the (idealized) scenario where A is perfect, one would have $c_t(\mathbf{x}) = 1$ if and only if $q(\mathbf{x}) > t$ for a q-quantile $q(\mathbf{x})$, hence the algorithm would output $\int_0^{q(x)} dt = q(x)$ exactly. The quanting analysis [12] shows that if the error of A is small on average over t, the quantile estimate is accurate.

3.2 Quantile k-Nearest Neighbor

The traditional k-nearest neighbor model is defined as

$$\hat{y}(\mathbf{x}) = \frac{1}{k} \sum_{\mathbf{x}_i \in N_k(\mathbf{x})} y_i, \tag{2}$$

where $N_k(x)$ is the neighborhood of $\mathbf{x}$ defined by the k closes points $\mathbf{x}_i$ in the training sample for a given distance measure (e.g., Eucledian). From a statistical perspective we can view the set $y_j \in N_k(\mathbf{x})$ as a sample from approximated conditional distribution of $P(Y|\mathbf{x})$. The standard kNN estimator of $\hat{y}$ is simply the expected value of this conditional distribution approximated by a local neighborhood. For quantile estimation we are not interested in the expected value (i.e., an estimate of $E(Y|\mathbf{x})$) but rather a particular quantile $c(\mathbf{x})$ of the conditional distribution $P(Y|\mathbf{x})$ such that $P(Y \leq c(\mathbf{x})|\mathbf{x}) = q$. Accordingly we can estimate $\hat{c}(\mathbf{x})$ in a k-nearest neighbor setting as the q'th quantile of the empirical distribution of $\{y_j : \mathbf{x}_j \in N_k(x)\}$. If we denote that empirical distribution by:

$$\hat{G}_{\mathbf{x}}(c) = 1/k \sum_{\mathbf{x}_j \in N_k(\mathbf{x})} \mathbf{I}\{y_j \leq c\} \tag{3}$$

then our kNN estimate of the q'th quantile of $P(Y|\mathbf{x})$ would be $\hat{G}_{\mathbf{x}}^{-1}(q)$.

The interpretation is similarly that the values of Y in the neighborhood $N_k(\mathbf{x})$ are a sample from the conditional distribution $P(Y|\mathbf{x})$ and we are empirically estimating its q'th quantile.

An important practical aspect of this estimate is that, in contrast to the standard kNN estimates, it imposes a constraint on k. While $k = 1$ produces an unbiased (while high variance) estimate of the expected value, the choice of k has to be at least $1/(1-q)$ to provide an upper bound for the estimate of the qth 'high' quantile (more generally we have $k \geq \max(1/q, 1/(1-q))$). The issue of how exactly to estimate the q'th quantile when q/k is not an integer (and hence the quantile of the empirical distribution falls in between observed y_j values) also has to be addressed. In our experiments below we simply select the integer closest to q/k as the index for the estimate, although interpolation approaches may be considered as well.

The definition of neighborhood is determined based on the set of variables, the distance function and implicit properties such as scaling of the variables. The performance of a kNN model is very much subject to the suitability of the neighborhood definition to provide a *good* approximation of the true conditional distribution — this is true for the standard problem of estimating the conditional mean and no less so for estimating conditional quantiles.

3.3 Quantile Regression Tree

Tree-induction algorithms are very popular in predictive modeling and are known for their simplicity and efficiency when dealing with domains with large number of variables and cases. Regression trees are obtained using a fast divide and conquer greedy algorithm that recursively partitions the training data into subsets. Therefore, the definition of the neighborhood that is used to approximate the conditional distribution is not predetermined as in the case of the kNN model but optimized locally by the choice of the subsets. Work on tree-based regression models traces back to Morgan and Sonquist [15] but the major reference is the book on classification and regression trees (CART) by Breiman et al. [5]. We will limit our discussion to this particular algorithm. Additional regression tree implementation include RETIS [10], CORE [17], M5 [16], RT [20].

A tree-based modeling approach is determined predominantly by three components:

- the **splitting criterion** which is used to select the next split in the recursive partitioning,
- the **pruning method** that shrinks the overly large tree to an optimal size after the partitioning has finished in order to reduce variance,
- the **estimation method** that determines the prediction within a given leaf.

The most common choice for the splitting criterion is the least squares error (LSE). While this criterion is consistent with the objective of finding the conditional expectation, it can also be interpreted as a measure of the improvement of the approximation quality of the conditional distribution estimate. Tree induction searches for local neighborhood definitions that provide good approximations for the true conditional distribution $P(Y|\mathbf{x})$. So an alternative interpretation of the LSE splitting criterion is to understand it as a measure of dependency between Y and an x_i variable by evaluating the decrease of uncertainty (as measured by variance) through conditioning. In addition, the use of LSE leads to implementations with high computational efficiency based on incremental estimates of the errors for all possible splits.

Pruning is the most common strategy to avoid overfitting within tree-based models. The objective is to obtain a smaller sub-tree of the initial overly large tree, excluding those lower level branches that are unreliable. CART uses Error-Complexity pruning approach which finds a optimal sequence of pruned trees by sequentially eliminating the subtree (i.e., node and all its ancestors) that minimizes the increase in error weighted by the number of leaves in the eliminated subtree:

$$g(t, T) = \frac{E(t) - E(T_t)}{S(T_t) - 1} \tag{4}$$

where $E(T_t)$ is the error of the subtree T_t containing t and all its ancestors, and $E(t)$ is the error if it was replaced by a single leaf, and $S(T_t)$ is the number of leaves in the subtree. $E(.)$ is measured in terms of the splitting criterion (i.e., for standard CART it is squared error loss). Given an optimal pruning sequence, one still needs to determine the optimal level of pruning and Breiman et al. suggest cross validation on a holdout set.

Finally CART estimates the prediction for a new case that falls into leaf node l similarly to the kNN algorithm as the mean over the set of training responses D_l in the leaf:

$$\hat{y}_l(x) = \frac{1}{n_l} \sum_{y_i \in D_l} y_i \tag{5}$$

where n_l is the cardinality of the set D_l of training cases in the leaf.

Given our objective of quantile estimation, the most obvious adjustment to CART is to replace the sample mean estimate in the leaves with the quantile estimate using the empirical local estimate $\hat{G}_{D_l}(c)$ of $P(Y|\mathbf{x})$ as in (3).

A more interesting question is whether the LSE splitting (and pruning) criterion should to be replaced by a quantile loss. On one hand, finding splits that minimize the quantile loss on the training sample in the leaves corresponds directly to our prediction objective. On the other hand, having the best possible approximation of the conditional distribution can be expected to result in the best quantile estimates of the distribution and minimizing the distribution variance could lead to a better approximation than the direct optimization of quantile loss, in particular for very high quantiles. In addition, changing the splitting criterion to quantile loss causes severe computational problems. The evaluation of a split now requires the explicit construction of the two sets of predictions in each leaf, sorting both of them in order to find the correct quantile and the calculation of the loss. We will not consider the issue of efficiency any further, as there is already some related work by Torgo [20] on efficient implementations of trees that minimize mean absolute deviation (MAD), i.e., quantile loss for q=0.5. In our experiments below, we investigate the success of the two splitting criteria in terms of predictive performance only.

4. WALLET ESTIMATION AT IBM: THE MAP PROJECT

Our framing of wallet modeling as a quantile estimation task leads to a well-defined machine learning problem. While we can assess the relative model performance for different modeling approaches in terms of quantile loss (as we will do in Section 5), the fundamental question of how well our models perform as wallet estimators remains open. And in particular, we have no strong indication about the appropriate choice of the quantile for the IBM customer wallets. In this section we describe the Market Alignment Program (MAP), which demonstrates a major use for wallet estimates within IBM, and which supplied us with a unique opportunity to evaluate the success of wallet estimation models in capturing experts' notion of IBM customer wallets.

4.1 Market Alignment Program

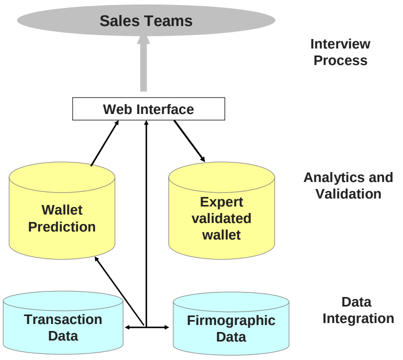

Figure 2: Overview of the MAP Tool

In 2005 IBM started an initiative called the Market Alignment Program ([13]) to address the major challenge of aligning sales resources with the best revenue-generating opportunities. The main objective of MAP is to drive the sales resources allocation process based on field-validated analytical estimates of future revenue opportunity. While expert knowledge is crucial to facilitate the sales process, the sole reliance on expert knowledge may lead to an overly strong focus on existing and large customers with limited growth opportunities. In order to facilitate the discussion with sales experts, initial model-based wallet estimates are used as a starting point in the assessment of future revenue opportunity. An integral part of the MAP process is the validation of the analytical estimates via an extensive set of workshops conducted with sales leaders. These interviews rely on a web-based tool to convey the relevant information, and to capture the expert feedback on the analytical models. The tool allows the sales team to input their estimates of revenue opportunity, as well as their reasons for recommending a change to the model results. As a side effect of this interview process, we now have at hand "validated" customer-wallet estimates against which we can evaluate our models. Figure 2 shows a high-level view of the MAP web-based tool.

The complete MAP process consists of:

1. Developing a consistent data model incorporating all relevant information for each customer, including historical transactions with IBM, company firmographics (e.g. annual revenue, number of employees, etc), and IBM sales coverage information

2. Estimating the wallets for all IBM customers within each major product group using a simple quantile kNN wallet model

3. Developing a web-based tool designed to display historical revenue along with the model-estimated revenue opportunities for each IBM sales account (an account consists of one or more IBM customers), as well as capture expert feedback on these estimates

4. Conducting workshops with sales leaders to validate

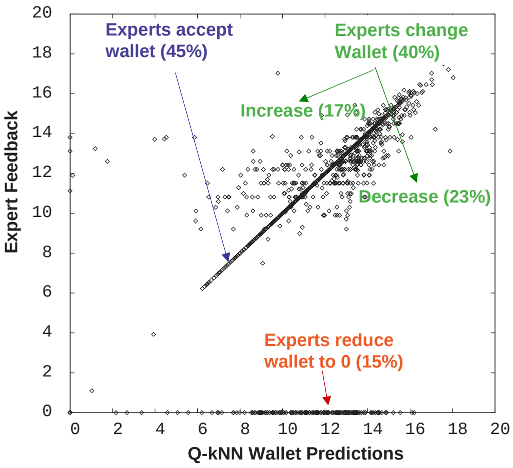

Figure 3: Expert Feedback versus the shown Q-kNN Wallet Predictions

the model-estimated future revenue opportunities for each sales account

5. Shifting sales resources to sales accounts with the largest validated revenue opportunities.

4.2 Analytical Details

For the initial round of workshops conducted in late 2005, the wallet estimates displayed in the MAP tool were generated using a simple and intuitive quantile kNN approach that follows our definition of REALISTIC wallet. For each of the approximately 100,000 customers available for this study, we identified a set of 20 similar companies, where similarity is based on the industry and a measure of size (either sales or employees, depending on the availability of the data). From this set of 20 firms, we discarded all companies with zero IBM revenue in the particular product group and reported the median of the IBM revenues of the remaining companies as the wallet estimate for this product group. The choice of the median (50th percentile) reflected a combination of statistical considerations and ad-hoc business constraints (such as conforming to a 'known' total market opportunity, i.e., sum over all companies). This estimated wallet was sometimes smaller than the realized 2004 IBM revenue for some companies. The final reported estimates were therefore taken as the maximum of the kNN model estimate and last years revenue (we refer to this operation as *flooring*). These wallet predictions were aggregated into opportunities by sales account according to the IBM internal account structure. The MAP workshops covered a total of about 1200 important sales accounts. Figure 3 presents the expert-validated opportunity for a major IBM software brand as a function of the calculated opportunity estimates from the 2005 workshops.

We can make a number of interesting observations here:

1. 45% of the opportunity estimates are accepted without alteration. The majority of the accepted opportunities are for smaller accounts. This shows a strong human bias towards accepting the provided numbers, and emphasizes the value of supplying estimates where the experts have little knowledge.

2. For 15% of the accounts, the experts concluded that there was NO opportunity - mostly for competitive reasons.

3. Of the remaining 40% of accounts, opportunity estimates were decreased (23%) slightly less often than they were increased (17%).

4. The horizontal lines reflect the human preference towards round numbers.

5. The opportunities and the feedback appear almost jointly normal in a log plot. This suggests that the opportunities have an exponential distribution with potentially large outliers, and that the sales experts corrected the opportunities in terms of percentage.

4.3 Evaluating Wallet Models

While the purpose of the MAP workshop was not primarily to validate our models, as a side effect we now have 1200 "true" wallets (based on experts' opinions) for 2006 at the sales-account level that we can use to compare and evaluate our different wallet modeling approaches. We decided to eliminate the 15% of accounts where the experts reduced the opportunity to zero for competitive reasons. This information is not available to the models and we did not want to bias our evaluation. It may be an interesting classification problem to identify accounts without opportunity, but we currently want to focus the quality of our estimates if there is an opportunity.

It is a well established fact that monetary quantities (like wallets) typically have a very long tailed exponential-type distribution. The few largest numbers, corresponding to biggest IBM customers, would typically dominate modeling and evaluation. And homoscedasticity assumptions underlying most modeling and evaluation approaches typically do not hold for monetary quantities. This is clearly shown by the experts' tendency to make wallet adjustments by percents rather than dollars (corresponding roughly to homoscedasticity on the log scale), as discussed above. On the other hand, success of models in a business environment is ultimately measured in dollars, not log-dollars. We therefore chose to evaluate model performance on three scales: error on original (dollar) scale, square root scale, and log scale. In addition to the sum of squared errors for each scale, we also considered the absolute error. This provides us with a total of six different performance criteria.

We adopted the modeling approaches discussed in the previous section, and to account for our lack of knowledge about what truly defines a customer wallet, we allowed the model parameters – such as the quantile being modeled, neighborhood size, etc. – to vary. In total, we built nearly 100 different models, counting all variations of model parameters, input variables, and different quantiles. We followed the same aggregation process and calculated the resulting opportunities for 2006 at the sales-account level for each

Table 1: Model performance in terms of number of times a wallet model was within the top 20 models across the 6 different performance metrics.

Model	Brand 1	Brand 2	Brand 3
Shown Q-kNN	6	5	6
Max Revenue 03-05	1	3	4
Linear Q-Regression	6	4	5
Q-kNN	1	0	2
Q-kNN + Flooring	3	6	6
Q-Tree	1	4	4

model. We finally ranked all models according to each of the 6 performance criteria, and compared how often a given model appears within the top 20 of all models. Table 1 shows the relative performance of the best variants and includes as reference points the performance of the shown Q-kNN model and of a very naive model that predicts simply for each acctoun the maximum revenue over the last 3 years.

The results in Table 1 support the following conclusions:

- Q-kNN performs very well after flooring but is typically inferior prior to flooring;
- 80th percentile seems to be the most appropriate quantile to capture experts' definition, across multiple approaches;
- Linear quantile regression performs consistently well (flooring has a minor effect);
- Models without last years revenue do not perform well.

Based on this analysis for three major product brands, we concluded that the linear quantile regression model showed the best overall performance, when using a quantile of 0.8. In other words, this quantile regression model provided the best agreement with the expert feedback collected during the initial 2005 MAP workshops. Hence, this model was selected to provide the revenue opportunity estimates for MAP workshops conducted in late 2006. The results from this iteration are not yet available.

5. EXPERIMENTS IN QUANTILE MODELING

In the sequel we compare experimentally the performance of different quantile estimation models that were presented in Section 3:

- **Linear Q-Regression:** linear quantile regression approach from Section 3.1 as developed by Koenker [11].
- **Quanting:** reduction approach from Section 3.1 to quantile estimation using ensembles of 100 classification trees.
- **Q-kNN:** quantile k-nearest neighbor algorithm from Section 3.2 with quantile rather than mean prediction in the neighborhood for k=50 use the default setting of the WEKA implementation including rescaling of the variables.
- Three versions of our quantile regression trees from Section 3.3 including **LSE Q-Tree** that uses least squared error splitting, **LSE Q-Tree** using quantile loss splitting, and finally **Bagged LSE Q-Tree** to make the results comparable in terms of variance error [4] to the quanting performance that combines an ensemble of 100 trees.

In addition we generate quantile estimates from the corresponding traditional modeling approaches including linear regression, kNN, and CART. These models attempt to model the mean and cannot perform well in high quantile estimation unless the distribution is highly skewed and the expected value happens to correspond to a high quantile. However, if we assume homoscedastic gaussian error model, then predicting the mean and predicting a high quantile are equivalent, as discussed in Section 2, and the 0.9th conditional quantile of Y would be just $E(Y|\mathbf{x})+1.28\sigma$. We adopt this approach as a means of converting estimates of conditional means into estimates of 0.9th conditional quantiles in our experiments. We estimate σ^2 from the test set as $E(y_i - \hat{y}_i)^2$.

As baseline, we also present the performance of the optimal **Constant** model that predicts for all observations the 90th percentile of the training set.

We collected in addition to the IBM domain a number of public datasets with similar motivation:

1. **Adult:** available from the UCI Machine Learning Repository [3] as a classification dataset. The data was originally extracted from the Census Bureau Database and describes individual demographic characteristics of such as age, education, sex and occupation. For the original dataset, the objective is to predict a label that indicates whether or not the individual's income is above \$50K. We have retrieved the original numerical income values from the Census Bureau Database and used the income as the dependent variable in the quantile regression. Our objective is to predict the 0.9th quantile of the conditional income distribution. This is an interesting piece of information, because it reveals what individuals can 'wish for' in terms of income, given their demographic characteristics.
2. **California Housing:** available from the StatLib repository [1]. It contains data on California housing characteristics aggregated at the block level (an average block group includes 1425.5 individuals living in a geographically compact area). The independent variables are neighborhood characteristics like median income, housing median age, longitude etc. The dependent variable is the median house value. Our objective is to predict the 0.9th quantile of the conditional house value distribution. This information is very valuable for house sellers and buyers, since it indicates what would be an 'upper bound' on the house value, given its characteristics.
3. **KDD-Cup 1998:** available at the UCI KDD Archive [2]. This dataset consists of records of individuals who

Table 2: Characteristics of the datasets including training and test size as well as the number of numeric and nominal variables.

Domain	Training	Test	Numeric	Nominal
KDD98	4840	4870	7	3
California	13760	6880	8	0
Adult	32560	16280	6	8
IBM	20000	63000	14	6

have made a donation in the past to a particular charity. Each example consists of attributes describing each individual's donation history over a series of donation campaigns, as well as demographic information. The dependent variable is the individual's donation amount in the most recent campaign. The original dataset contains 95412 training records and 96367 test records, but only 5% of the individuals donated in the current campaign. Our objective is to predict the 0.9th quantile of the conditional donation amount for individuals who donate. For this reason, we only use 4843 donor examples in the training set and the 4876 donor examples in the test set. Predicting a high quantile of the conditional donation distribution is important for 'anchoring', i.e., deciding how much to suggest as a possible donation value when soliciting donations. Anchoring is a well-established concept in marketing (e.g., [8, 9, 21]).

4. **IBM Wallet:** a subset of the data discussed in Section 4. It contains the purchase history and some firmographic characteristics (such as industry and number of employees) of companies that are IBM customers. The dependent variable is the amount that the company has spent with IBM in the most recent year. By modeling the 0.9th quantile of the conditional spend, we get an estimate of the REALISTIC wallet of the customer, as defined in [18]. We use both the original monetary values and also a log-transformed version, in which all numeric features are transformed to the log-scale, as discussed in Section 4.

The main domain characteristics including size of training and test sets and the number of numeric and nominal variables are shown in Table 2.

Since all of our datasets are reasonably large, we used the same training-test split for all modeling approaches according to the sizes in Table 2. The results in Table 3 report the average quantile loss on the test sample with the standard deviation of this average (estimated as the standard deviation of the quantile loss divided by the root of the number of test cases) in parentheses. Given the large size of all test sets and the central limit theorem we can argue that the average quantile loss is approximately gaussian and we can use the deviation to assess the uncertainty in these evaluation scores. The bold indication of the best models is based on pairwise t tests to the method with the best results. Any method that was not significantly worse than the best one is in bold.

Looking at Table 3, we can draw several interesting conclusions:

1. On these datasets, the quantile modeling approaches dominate the 'standard' modeling approaches. In fact, all standard methods perform worse than all quantile approaches on all datasets.

2. Not surprisingly, there is no clear winner among the quantile approaches. Surprisingly, however, one method (Linear Quantile Regression) is significantly better than all other quantile approaches on two datasets, and significantly worse than the best method on the other three! The special property of KDD98 and IBM which makes Linear Quantile Regressions successful is likely the presence of an independent variable which is highly linearly correlated to the response (last year's donation, and last year's IBM Sales, respectively). The linear regression successfully takes advantage of this correlation, while the other methods evidently make less efficient use of it.

3. The IBM dataset suffers from the highest evaluation noise by far (as reflected in the ratio of expected error to its standard deviation). This is another piece of evidence of the difficulty in evaluating customer spending on the original (Dollar) scale — evaluation is dominated by a few big customers and is not stable even with the huge test set we have.

4. The particularly bad performance of all standard models, where the predictions have been created by adding 1.28 times the error standard deviation, is probably caused by the extreme skew of the distribution of the response. This causes a similar skew in the distribution of the error and violates strongly the normality assumption underlying the justification to add 1.28 times the standard deviation. These models do no better at quantile estimation without this addition, though.

5. The two tree-based quantile models (LSE and QL Q-Tree) often produce remarkably similar results. In particular, using the 'wrong' but computationally efficient criterion (LSE) for splitting and pruning has no detrimental effect on the performance in terms of quantile loss. This supports our view that the role of splitting criteria is mostly to find a good approximation of the local density.

6. Both tree-based ensemble methods, Bagged LSE and quanting, perform similarly on all problems, and in fact do not have a significant difference in performance on any of them.

As discussed in Section 2, we can also evaluate our success in predicting a high quantile also by the percentage of test cases where the model predictions are higher than actual observations.

Table 4 shows the percentage of time that each model predicts a higher value than the observed response on the test set. As we can see, all numbers are between 82% and 91%, and they have a slight tendency for negative bias (i.e., predicting above the response slightly less than 90% of the time). Linear Quantile Regression clearly does the best job in this criterion. We hypothesize that this is due to its being the simplest method, which is least prone to overfitting, hence is most consistent in terms of test-set performance.

Table 3: Model performance in terms of quantile loss on test set with standard deviation of the estimate of the mean loss in parentheses.

Approach	KDD98	California	Adult	IBM	Log-IBM
Constant	2.51 (0.144)	25217 (259)	5976 (34.9)	412810 (86793)	0.543 (0.0047)
Linear Regression	1.95 (0.064)	14543 (250)	3564 (32.6)	555360 (21701)	0.363 (0.0035)
CART	1.86 (0.0831)	12774 (235)	3348 (36.1)	881310 (36826)	0.372 (0.0035)
kNN	2.22 (0.118)	14036 (258)	37144 (36.8)	1751200 (73683)	0.361 (0.0032)
Linear Q-Regression	**1.223** (0.061)	13901 (245)	3332 (30.7)	**66049** (13720)	0.279 (0.0036)
Quanting	1.383 (0.094)	**9797** (204)	**2837** (32.4)	108370 (46663)	**0.251** (0.0036)
Q-kNN	1.630 (0.115)	12532 (235)	3232 (35.5)	244910 (67661)	0.279 (0.0039)
LSE Q-Tree	1.321 (0.086)	11695 (254)	2915 (34.7)	92831 (29411)	0.259 (0.0039)
QL Q-Tree	1.325 (0.087)	11681 (231)	2979 (33.8)	94920 (29636)	0.257 (0.0036)
Bagged LSE Q-Tree	1.319 (0.092)	**9989** (217)	**2830** (30.1)	91032 (28353)	**0.254** (0.0036)

Table 4: Model performance in terms of the percent of observation above the prediction for the 90th quantile where the optimal performance would be 0.9.

Approach	KDD98	California	Adult	IBM	Log-IBM
Linear Q-Regression	**0.9054**	**0.9010**	**0.8960**	0.9027	**0.9003**
Q-kNN	0.8521	0.8833	0.8678	0.8548	0.8593
LSE Q-Tree	0.8286	0.8562	0.8736	**0.8997**	0.8676
Quanting	0.8830	0.8936	0.8692	0.8627	0.9063

Given our motivation for high-quantile estimation, as a way of estimating 'what can be hoped for', it is interesting to consider the difference between expected predictions of the standard methods and those of the quantile modeling approaches. For example, on the Adult dataset, this can tell us how much higher the salary request would be of a person using a quantile model to assess her prospects compared to someone using a standard approach and trying to estimate their expected salary. We would have liked to analyze these differences on our motivating problem of IBM wallet, but due to the extreme instability of these predictions, do not really trust them. On the Log-IBM dataset, interpretability is an issue. Thus, we chose to analyze the Adult dataset in more detail. We display in Figure 4 a histogram of the difference in predictions between the CART model (without the adjustment to prediction quantiles we discussed above) and Bagged LSE model on this dataset . As expected, practically all differences are non-negative. The mean difference is $18643 and the median $16715, representing the differences in salary expectations between the humble and the ambitious.

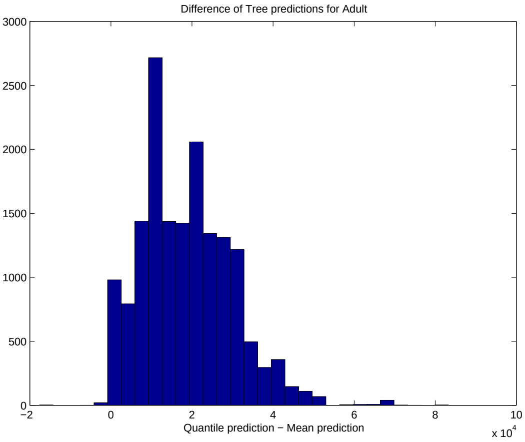

Figure 4: Histogram of differences in prediction between CART and Bagged LSE on Adult test set.

The corresponding dollar mean differences in prediction for KDD98, California and IBM are $6.24, $63757, $612380, respectively. This dramatically demonstrates the difference between predicting the mean and predicting the 90th quantile on these problems.

6. CONCLUSIONS

In this paper we have argued for the practical importance of high-quantile modeling in many problem domains, including wallet estimation, price/salary prediction and others. We reviewed the statistical considerations involved in designing methods for high-quantile estimation and described some existing quantile modeling methods, as well as our own adaptations of kNN and CART to quantile modeling.

Next, we described the MAP application and utilized the output from its first iteration to analyze which of a large candidate set of models is most consistent with IBM Sales executives' notion of wallet. One interesting conclusion from our analysis is that the experts relied quite heavily on the numbers we presented to them (which were the output of a simplistic 'first approximation' model). The second conclusion is that the experts' notion of customer wallet seems most consistent with a high-quantile model for the 0.8 quantile, compared to our previously proposed definition of the 0.9 quantile [18]. The model which performed best overall was a linear quantile regression model which (naturally) relies heavily on the previous-year observed sales revenue to predict current-year wallet.

We then performed an empirical study on several problems where high-quantile modeling is a well motivated goal. Our main conclusions are:

- The difference between predicting a conditional mean and predicting a conditional high quantile does indeed seem to be very significant from a practical perspective. As Figure 4 and the average prediction differences quoted there demonstrate.

- Quantile modeling approaches do indeed perform significantly better for high-quantile estimation than mean-modeling approaches, adapted to high-quantile estimation by a gaussian-based correction. This is not surprising given all we know about data mining domains, in particular the inappropriateness of gaussianity and homoscedasticity assumptions in these domains.

- The splitting criterion in tree induction methods is only indirectly related to the modeling objective. Using quantile loss rather than the standard least squares criterion to optimize splitting and pruning has little effect on the performance of quantile prediction. The critical modification for quantile estimation seems to be using the empirical quantile, rather than the empirical mean, for prediction in the leaves.

- Different quantile modeling approaches perform well for different datasets. In particular, the linear quantile regression approach seemed to have a unique behavior on our datasets — it was sometimes significantly the best, sometimes the worst by far.

In summary, there is a set of algorithms readily available to address the relevant issue of quantile modeling. The necessary adjustments to classical machine learning techniques such as tree induction are straight forward and result in reliable, interpretable, and efficient solutions.

Acknowledgments

We thank Sholom Weiss, Paulo Costa, Alexey Ershov and John Waldes from IBM for useful discussions on the topic of this paper.

7. REFERENCES

[1] StatLib: Data, Software and News from the Statistics Community. Department of Statistics, Carnegie Melon University, 2006. `http://lib.stat.cmu.edu/`.

[2] S. D. Bay. UCI KDD archive. Department of Information and Computer Sciences, University of California, Irvine, 2006. `http://kdd.ics.uci.edu/`.

[3] C. L. Blake and C. J. Merz. UCI repository of machine learning databases. Department of Information and Computer Sciences, University of California, Irvine, 2006. `http://www.ics.uci.edu/∼mlearn/MLRepository.html`.

[4] L. Breiman. Bagging predictors. *Machine Learning*, 24,2:123–140, 1996.

[5] L. Breiman, J. H. Friedman, Olshen, R. A., and C. J. Stone. *Classification and regression trees.* Wadsworth International Group, Belmont, CA, 1984.

[6] P. Chaudhuri and W.-Y. Loh. Nonparametric estimation of conditional quantiles using quantile regression trees. *Bernoulli*, 8(5), 2002.

[7] J. Friedman, T. Hastie, and R. Tibshirani. Additive logistic regression: A statistical view of boosting. *Annals of Statistics*, pages 337–374, 2000.

[8] J. Hammond, R. Keeney, and H. Raiffa. The hidden traps in decision making. *Harvard Business Review*, 76(5), 1998, Sep-Oct.

[9] D. Kahneman and A. Tversky. On the psychology of prediction. *Psychology Review*, 80:237–251, 1973.

[10] A. Karalic. Employing linear regression in regression tree leaves. In *Proceedings of the European Conference on Artificial Intelligence*, pages 440–441. John Wiley & Sons, 1992.

[11] R. Koenker. *Quantile Regression.* Econometric Society Monograph Series. Cambridge University Press, 2005.

[12] J. Langford, R. Oliveira, and B. Zadrozny. Predicting the median and other order statistics via reduction to classification. In *UAI*, 2006.

[13] R. Lawrence, C. Perlich, S. Rosset, J. Arroyo, M. Callahan, M. Collins, A. Ershov, S. Feinzig, I. Khabibrakhmanov, S. Mahatma, M. Niemaszyk, and S. Weiss. Analytics-driven solutions for customer targeting and sales force allocation. Under Review IBM Systems Journal, 2007.

[14] N. Meinshausen. Quantile regression forests. *Journal of Machine Learning Research*, 7:983–999, 2006.

[15] Morgan and Sonquist. Problems in the analysis of survey data and a proposal. *JASA*, 58:415–434, 1963.

[16] R. Quinlan. Combining instance-based and model-based learning. In *Proceedings of the Tenth International Conference of Machine Learning*, pages pages 236–243. Morgan Kaufmann, 1993.

[17] M. Robnik-Sikonja. CORE - a system that predicts continuous variables. In *Proceedings of ERK*, 1997.

[18] S. Rosset, C. Perlich, B. Zadrozny, S. Merugu, S. Weiss, and R. Lawrence. Wallet estimation models. In *International Workshop on Customer Relationship Management: Data Mining Meets Marketing*, 2005.

[19] I. Takeuchi, Q. V. Le, T. Sears, and A. Smola. Nonparametric quantile regression. *Journal of Machine Learning Research*, 7:1231–1264, 2006.

[20] L. Torgo. Functional models for regression tree leaves. In *International Conference on Machine Learning*, pages 385–393, 1997.

[21] B. Wansink, R. J. Kent, and S. J. Hoch. An anchoring and adjustment model of purchase quantity decisions. *Journal of Marketing Research*, 35(1):71–81, 1998.