NRC Publications Archive
Archives des publications du CNRC

A Probabilistic Model for Data Cube Compression and Query Approximation

Missaoui, Rokia; Goutte, Cyril; Choupo, Anicet Kouomou; Boujenoui, Ameur

National Research Council Canada   Conseil national de recherches Canada

Canada

NRC·CNRC

*A Probabilistic Model for Data Cube Compression and Query Approximation *

Missaoui, R., Goutte, C., Kouomou Choupo, A., Boujenoui, A.
November 2007

Canada

A Probabilistic Model for Data Cube Compression and Query Approximation

Rokia Missaoui
University of Quebec in Outaouais
P.O. Box 1250, Suc. B
Gatineau, Quebec, J8X 3X7, Canada
rokia.missaoui@uqo.ca

Anicet Kouomou Choupo
University of Quebec in Outaouais
P.O. Box 1250, Suc. B
Gatineau, Quebec, J8X 3X7, Canada
akchoupo@yahoo.fr

Cyril Goutte
Interactive Language Technology
NRC Institute for Information Technology
283, Alexandre-Taché
Gatineau, QC J8X 3X7, Canada
Cyril.Goutte@nrc-cnrc.gc.ca

Ameur Boujenoui
School of Management, University of Ottawa
136 Jean-Jacques Lussier
Ottawa, ON, K1N 6N5, Canada
boujenoui@management.uottawa.ca

ABSTRACT

Databases and data warehouses contain an overwhelming volume of information that users must wade through in order to extract valuable and actionable knowledge to support the decision-making process. This contribution addresses the problem of automatically analyzing large multidimensional tables to get a concise representation of data, identify patterns and provide approximate answers to queries.

Since data cubes are nothing but multi-way tables, we propose to analyze the potential of a probabilistic modeling technique, called non-negative multi-way array factorization, for approximating aggregate and multidimensional values. Using such a technique, we compute the set of components (clusters) that best fit the initial data set and whose superposition approximates the original data. The generated components can then be exploited for approximately answering OLAP queries such as roll-up, slice and dice operations. The proposed modeling technique will then be compared against the log-linear modeling technique which has already been used in the literature for compression and outlier detection in data cubes. Finally, three data sets will be used to discuss the potential benefits of non-negative multiway array factorization.

Categories and Subject Descriptors

H.2.8 [**Information Systems**]: Database Applications—*data mining*; H.4.2 [**Information Systems**]: Types of Systems—*decision support*

General Terms

Algorithms, experimentation

1. INTRODUCTION

In many database and data warehouse applications, users tend to (1) be drowned in data and even in knowledge discovered from data, (2) use more dimensions (variables) than necessary to explain a set of phenomena and check predefined hypotheses, and (3) analyze a generally heterogeneous population of observations/individuals. In order to reduce their memory overload and working space induced by their tendency for over-dimensioning and the inherent heterogeneity and huge volume of data, we propose to use a probabilistic modeling technique, called non-negative multi-way array factorization (NMF), to identify patterns and approximate data and query answers. Such a technique will then be compared to log-linear modeling (LLM).

Many topics have attracted researchers in the area of data warehousing: data warehouse design and multidimensional modeling, materialized view selection, efficient cube computation, query optimization, discovery-driven exploration of cubes, cube reorganization, cube mining, and so on. Recently, there has been an increasing interest for applying/adapting data mining techniques and advanced statistical analysis (e.g., cluster analysis, principal component analysis, log-linear modeling) for knowledge discovery [21, 19, 24, 18] and data compression in data cubes [3, 4, 5, 13]. In [24], an approach based on log-linear modeling is used to identify exceptions in data cubes by comparing anticipated cell values against actual values. In [4, 5], log-linear modeling is used for data compression.

Approximate query answering in data warehouses has been used in order to accelerate aggregate computation and query execution at the expense of some information loss. Existing work has been conducted based mainly on sampling [3, 13], clustering [29], wavelets [7] or maximum entropy principle [23]. Palpanas *et al.* [23] propose an approach based on information entropy to (i) detect deviations, and (ii) estimate the original multidimensional data from aggregates for approximate query answering purposes. A wavelet-based

approach is used in [7] to approximate query answering, and proves to be more effective than sampling techniques. In a similar spirit, [26] uses the probability density distribution of data in order to propose a compressed representation of data cubes which reduces data storage and leads to approximate answers to aggregate queries. Wavelet based techniques have been used either for progressive evaluation of some specific OLAP queries [2, 7] or as a sampling technique. As in [26], Vitter *et al.* [27] use wavelets for compressing sparse data cubes and getting approximate answers of aggregate queries. In [22], an approach towards the approximation of the answer to OLAP queries and the identification of classification and characteristic rules is proposed using the rough set theory. It allows the user to get an approximate answer to his query either in a restricted mode using a cube lower approximation or in a relaxed mode using cube upper approximation.

The NMF approach was initially developed for analyzing two-dimensional data [20], and generalizes easily to higher dimensional data [28]. It has been applied to analyze various data such as image, music, text or medical signals (see Section 6 of [15] for further references). To the best of our knowledge, NMF has never been applied to the analysis and exploration of data cubes.

The paper is organized as follows. First, we describe a real-life running example in Section 2 and provide in Section 3 some background about probabilistic modeling using NMF or LLM. Section 4 aims to show the potential of the selected modeling technique for cube approximation, compression and OLAP querying. Section 5 provides a discussion of the two modeling techniques while Section 6 summarizes our findings and future work.

2. AN ILLUSTRATIVE EXAMPLE

We now introduce the running example that will be used in the sequel to illustrate the use of NMF for cube modeling and approximation. It is based on a study conducted on a sample of 214 Canadian firms listed on the Stock Market and aimed at establishing links between corporate governance practices and other variables such as the shareholding structure [6, 15]. Governance is defined as the means, practices and mechanisms put in place by organizations to ensure that managers are acting in shareholders' interests. Governance practices include, but are not limited to, the size and the composition of the Board of Directors, the number of independent directors and women sitting on the Board as well as the duality between the position of CEO and the position of Chairman of the Board. Based on the collected data, a data warehouse has been constructed with sixteen dimensions and an initial set of fact tables for data cube mining and exploration. Table 1 is a fact table which provides the number of firms according to four dimensions: USSX, DUALITY, SIZE, and QI. USSX$\in \{$Yes, No$\}$ indicates whether the firm is listed or not on a US Stock Exchange. DUALITY $\in \{$Yes, No$\}$ indicates whether the CEO is also the Chairman of the Board. SIZE $\in \{1, 2, 3, 4\}$ represents the size of the firms in terms of the log of their assets. The values are 1 (< 2), 2 (≥ 2 and < 3), 3 (≥ 3 and < 4) and 4 (≥ 4). QI expresses the index of corporate governance quality and takes one of the following values (from worst to best quality): Low ($< 40\%$), Medium (≥ 40 and $< 70\%$), and High ($\geq 70\%$).

USSX	SIZE	DUALITY: No			Yes		
		QI:Low	Med	High	Low	Med	High
No	1	0	7	0	4	3	0
	2	7	21	12	6	12	4
	3	11	13	11	4	4	2
	4	0	3	1	0	2	0
Yes	1	0	1	2	0	0	0
	2	4	12	0	7	10	1
	3	4	4	14	5	8	2
	4	0	3	7	0	2	1

Table 1: Data cube for the Governance example.

3. PROBABILISTIC DATA CUBE MODELS

Consider a n-dimensional data cube $\mathbf{X}$, and assume that the measure in each cell is a frequency $x_{i_1 i_2 \ldots i_n} \in \mathbb{N}$.[1] Without loss of generality and to simplify notation, we will use $n = 3$ from now on. The three dimensions are denoted A, B and C, with respective modalities $\{a_i\}_{i=1 \ldots I}$, $\{b_j\}_{j=1 \ldots J}$, and $\{c_k\}_{k=1 \ldots K}$. Each cell x_{ijk} of the 3-dimensional cube $\mathbf{X} = [x_{ijk}]$ therefore contains the count of the number of observations of the three modalities (a_i, b_j, c_k).

Information contained in a high dimensional data cube may be difficult to access or visualize. For example, it may be hard to identify which dimensions interact, or which parts of the cube display consistent patterns. Probabilistic models such as NMF and log-linear models can discover such interactions and patterns, by finding a concise representation that provides a faithful approximation of the data. The use of a concise model reduces both storage space and processing time, while a good approximation ensures that queries performed on the model return reasonably accurate answers. A probabilistic model assigns a probability $P(i, j, k)$ to the observation of the tuple of modalities (a_i, b_j, c_k). The fit of the model to the observations is measured by the log-likelihood:

$$\mathcal{L} = -\log P(\mathbf{X}|\theta) = -\sum_{ijk} x_{ijk} \log P(i, j, k), \qquad (1)$$

The most flexible, or *saturated*, model estimates each $P(i, j, k)$ based on the data, i.e., $\widehat{P}(i, j, k) = x_{ijk}/N$, with $N = \sum_{ijk} x_{ijk}$. This model essentially reproduces the data without modeling any interesting or systematic effect. In our example, this means that it is not possible to discover that the quality of the governance QI is strongly dependent on the duality. On the other hand, the *independence model* assumes that all variables are independent: $\widehat{P}(i, j, k) = \widehat{P}(i)\widehat{P}(j)\widehat{P}(k)$, with Maximum Likelihood estimates: $\widehat{P}(i) = \sum_{jk} x_{ijk}/N$, (and similarly for j and k). This model is parsimonious (i.e., has a small number of parameters) but too simple (and often unrealistic) since it does not model interactions between variables. In our example, it means that the independence model will ignore the strong interaction between QI and DUALITY, i.e., it yields similar probability for high governance quality, regardless of the value of DUALITY.

Useful models usually fall somewhere in between these two extremes: they strike a balance between expressive power (good approximation) and parsimony (few parameters).

---

[1]This assumption can be easily relaxed to take into account values other than frequencies.

3.1 Non-negative Multi-way Array Factorization

The model associated with NMF is a mixture of conditionally independent multinomial distributions. It generalizes *Probabilistic Latent Semantic Analysis* [17], commonly used for text applications, to more than two dimensions:

$$\begin{aligned} P(i,j,k) &= \sum_{m=1}^{M} P(m)P(i|m)P(j|m)P(k|m) \quad (2)\\ &= \sum_{m=1}^{M} \mathbf{W}^m \otimes \mathbf{H}^m \otimes \mathbf{A}^m, \end{aligned}$$

where m is a given component and M the number of components. The latter controls the flexibility of the model. More components allow more free parameters and a better approximation of the data, but too much flexibility yields over-training, i.e., a model that over-represents artifacts of the data at the expense of the underlying phenomenon. The parameters $\mathbf{W}^m$, $\mathbf{H}^m$ and $\mathbf{A}^m$ are the m-th columns of the $I \times M$ matrix $\mathbf{W} = [P(m)P(i|m)]$, the $J \times M$ matrix $\mathbf{H} = [P(j|m)]$ and $K \times M$ matrix $\mathbf{A} = [P(k|m)]$ respectively. As $P(i,j,k) \approx \frac{1}{N}\mathbf{X}$, matrices $\mathbf{W}$, $\mathbf{H}$ and $\mathbf{A}$ are a factorization of cube $\mathbf{X}$. Since they contain probabilities, the factors are also non-negative. The mixture model in Equation 2 is therefore an instance of *Non-negative Multi-way array Factorization* [28], a generalization of Non-negative *Matrix* Factorization [20]. Parameters are estimated by Maximum Likelihood, which is conveniently done using the Expectation-Maximization (EM, [11]) algorithm. Several alternative algorithms exist for NMF in 2D [12], but algorithms currently proposed for $n > 2$ minimize either the Kullback-Leibler divergence (equivalent to Maximum Likelihood, see [14]) or squared error [28]. The parameter estimation procedure is as follows:

- Initialize mixture weights $P(m)$ and profiles $P(i|m)$, $P(j|m)$, etc. to uniform distributions plus small random perturbation.

- Iterate over $t = 0 \ldots$:

$$P^{t+1}(m) \leftarrow P^t(m) \sum_{ijk} \left(\frac{x_{ijk}}{N}\right) \frac{P^t(i|m)P^t(j|m)P^t(k|m)}{P^t(i,j,k)}$$

$$P^{t+1}(i|m) \leftarrow \frac{P^t(i|m)P^t(m)}{P^{t+1}(m)} \sum_{jk} \left(\frac{x_{ijk}}{N}\right) \frac{P^t(j|m)P^t(k|m)}{P^t(i,j,k)}$$

and similarly for $P^{(t+1)}(j|m)$ and $P^{(t+1)}(k|m)$ [15].

- until the increase in likelihood is smaller than a small threshold (e.g., 10^{-6}).

The main modeling issue is therefore to select the correct number of components. Although the NMF literature often ignores this issue, the probabilistic modeling framework offers tools for selecting the appropriate model complexity. One solution is to rely on information criteria such as AIC [1] or BIC [25]. Defining the (log) likelihood ratio as twice the difference in log-likelihood between the fitted model and the saturated model, $G^2 = 2\sum_{ijk} x_{ijk} \log \frac{\hat{x}_{ijk}}{x_{ijk}}$, and df the number of degrees of freedom, i.e., the difference in the number of free parameters between the fitted model and the saturated model, $df = IJK - (I + J + K - 2) \times M$, we have:

$$AIC = G^2 - 2df \quad \text{and} \quad BIC = G^2 - df \times \log N \quad (3)$$

The number M of components with the lowest AIC (or BIC) is then selected.

As components are represented by profiles along the various dimensions, $P(i|m)$, $P(j|m)$ and $P(k|m)$, each component may select a subset of modalities, by setting other probabilities to zero. One key feature of NMF is its ability to identify dense sub-cubes (expressed by components) inside the original, possibly sparse, data cube. Such a feature can then be exploited to approximate query answers using appropriate components (see Subsection 4.3) or to conduct further analysis on components using, for example, log-linear modeling.

3.2 Log-Linear Modeling

As indicated earlier, log-linear modeling has been used in the field of data warehousing for data approximation and outlier detection. The principle of this modeling technique [9] is to decompose the log-probability of given cell value as an additive sum of terms involving all combinations of subsets of dimensions:

$$\log P(i,j,k) = \lambda + \lambda_i^A + \lambda_j^B + \lambda_k^C + \lambda_{ij}^{AB} + \lambda_{ik}^{AC} + \lambda_{jk}^{BC} + \lambda_{ijk}^{ABC} \quad (4)$$

where λ is the overall mean (in the log domain) and all other parameters model the effect of a subset of the dimensions, e.g. λ_{jk}^{BC} models the joint effect of observing $B = b_j$ and $C = c_k$, regardless of variable A. In order to have an identifiable model, the marginals of the parameters are constrained to zero: $\sum_i \lambda_i^A = \sum_j \lambda_j^B = \sum_k \lambda_k^C = 0$, $\sum_i \lambda_{ij}^{AB} = \sum_j \lambda_{ij}^{AB} = 0$, etc.

When all terms are present in Equation 4, this corresponds to the saturated model. The complexity of the model may be controlled by removing some parameters (or setting them to zero), which adds the corresponding degrees of freedom. Of course, as parameters are removed, the model deviates progressively from data. Therefore, selecting the appropriate model is again a compromise between parsimony and approximation. This may conveniently be done using iterative model selection methods such as forward selection or backward elimination. In *backward elimination*, one starts with a large model, for example the saturated model, and iteratively eliminates interactions (from higher to lower order), as long as the associated increase in G^2 is not significant at a pre-specified level (usually 95%), until none of the remaining interactions may be eliminated. *Forward selection* proceeds the opposite way by starting with a small model (generally the independence model) and then adding interactions of increasing order. The parameters of the model may be obtained either by directly optimizing the likelihood with a Newton-Raphson algorithm [16], or by using the Iterative Proportional Fitting (IPF) procedure [10], a dedicated method for modeling multi-way frequency tables. These techniques may also be used together with information theoretic criteria such as the AIC or BIC (see Equation 3). Note that in the context of LLM, AIC seems to be preferred [9].

The most popular log-linear models are *hierarchical* models: all lower-order effects within higher-order ones are necessarily included. A common notation for hierarchical log-linear models is therefore to specify only terms with the highest order interaction. For example, $\{A * B, A * C\}$ denotes the hierarchical model containing $A * B$ and $A * C$ as well as first order terms A, B and C.

3.3 Rates of compression and approximation

The goal of probabilistic modeling is to produce an approximation of the data cube that uses as few parameters as possible. We will therefore be interested in measuring the compression rate and the approximation quality for each model. The compression rate is linked to the ratio of the number of free parameters f in the model and the total number of cells in the cube, N_c, and is defined as:

$$R_c = 1 - \frac{f}{N_c} = \frac{df}{N_c} \qquad (5)$$

With df the number of degrees of freedom defined above. By definition, the saturated model has a compression rate of 0% since $f = N_c$.

Typical probabilistic models offer a direct way to control the compression rate by tuning the model complexity. For NMF, the compression decreases linearly as the number of components grows. In our three dimensional example, $N_c = I \times J \times K$ and, with M components, $f = (I+J+K-2)*M$, such that:

$$R_c^{NMF} = 1 - M\frac{I+J+K-2}{IJK} \qquad (6)$$

In the situation where the *components* are sparse, i.e., many parameters are zero,[2] an arguably better way to measure the compression offered by the model is to take into account the non-zero parameters only. In such a situation, Equations 5 and 6 are *lower bounds* that offer a minimum guaranteed compression rate for a model with the given number of components.

The approximation quality may be measured in various ways such as the mean squared error or the mean absolute error. In the context of probabilistic models, it is measured by the log-likelihood (Equation 1) or, equivalently, the deviance G^2. The deviance offers a "standard" way to measure the approximation quality: perfect approximation yields a deviance of 0 and higher deviance means worse approximation, so it is a good way to compare various models of the same data cube. However, there is to our knowledge no way to offer a guaranteed approximation level *a priori* for a given data cube and model complexity (i.e., number of components in NMF or number of interactions in LLM). The optimal approximation error depends on the level of noise in the data. In typical situations, the noise level is unknown before modeling, therefore the resulting approximation level can not be guaranteed. Note however that for most probabilistic models, increasing the model complexity will systematically improve the approximation of the data on which the model is estimated. If a given approximation level is required, it is straightforward to increase the model complexity, e.g., by adding components in NMF, until the minimum deviance is reached.

4. DATA COMPRESSION, APPROXIMATION, AND RETRIEVAL

Besides the GOVERNANCE data cube described in Section 2, we used two synthetic data sets to illustrate the potential of NMF for data approximation/compression as well as approximate query answering. One of the sets is a simplified version of a dimension table called CUSTOMER of FOOD

---

[2]Near-zero parameters may usually be pruned from the model with no loss in approximation.

Features	Cubes		
	GOVERNANCE	CUSTOMER	SALES
Nb. of dimensions	4	5	3
Nb. of cells	48	2400	528
Nb. of records	214	10281	5191
Highest nb. modalities	4	8	44
Density	63%	37%	50%

Table 2: Features of the three data cubes. "Density" is the ratio of non-zero cells.

MART data delivered with Analysis Services of Microsoft SQL Server. From the CUSTOMER table (10281 records) we constructed a data cube of five dimensions: STATUS, INCOME, CHILD, OCCUPATION and EDUCATION. STATUS $\in \{1,2\}$ indicates whether the customer is single (value equal to 1) or not. INCOME takes eight possible values indicating the level of income (e.g., 1 for income between $10K$ and $30K$, and 8 for income $\geq 150K$). CHILD $\in \{0,1,2,3,4,5\}$ represents the number of children. OCCUPATION takes five possible values indicating the customer's occupation (e.g., 1 for a manual worker and 5 for a manager). EDUCATION refers to customer's education level and can take five possible values (e.g., 1 for partial high school studies and 5 for graduate studies). The other data set was also extracted from FOOD MART data and concerns a data cube of SALES according to product category (44 values), time (in quarters) and country (USA, Canada and Mexico). This cube has a relatively small set of dimensions but one of them has a large number of modalities (members).

Table 2 summarizes the features of the three data cubes in terms of their dimensionality, the number of members per dimension, the number of records/instances, the number of cube cells as well as the density. The first cube was selected because it represents real data while the other ones were retained for their relatively higher size or dimensionality.

4.1 Model Estimation

We illustrate model estimation for both NMF and LLM through the GOVERNANCE cube.

For NMF, the information criterion AIC identifies three components (see Figure 1). Therefore, the user can see these components as a way to cluster the initial cube into homogeneous data sets, and hence can focus his analysis on some identified components rather than the whole population (cube). The first component C_1 represents only 16% $(P(C_1) = 0.16)$ of the observations and corresponds to companies with no duality (i.e., they have distinct CEO and Chairman of the Board), listed in a US stock exchange and, with high governance quality. This component represents what we could call the "virtuous" companies. The second component represents the rest of the companies listed on a US Stock Exchange, for which the governance quality is lower than in component C_1. Finally, the last and largest component contains small to moderate-sized companies which are not listed in the US Stock Exchange.

The parsimonious model generated from this cube using LLM is {QI*SIZE*USSX, QI*DUALITY}. This means that (i) the link between QI, SIZE and USSX needs a three-way interaction, expressing the fact that the relationship between governance quality and USSX is not the same in each value of firm size, and (ii) DUALITY is only involved in a two-way interaction with QI, indicating that the presence or absence of duality in governance quality is the same for all
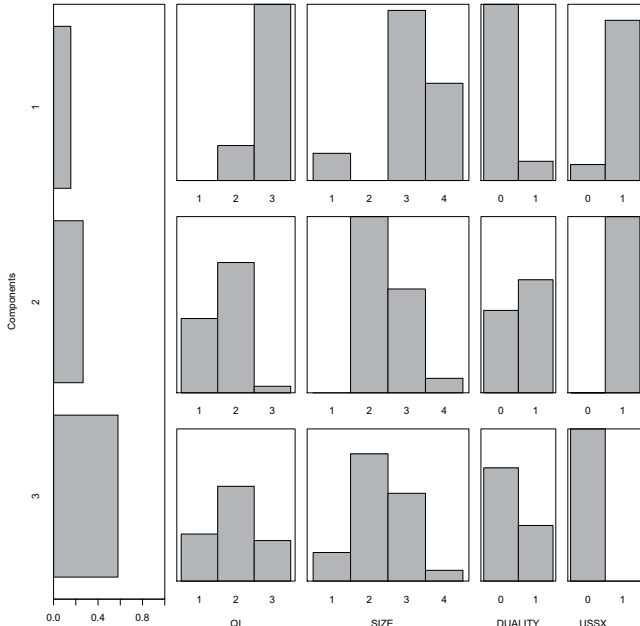
Figure 1: Parameters of the 3-component NMF model. The tall left pane shows $P(c)$. Each row shows $P(\mathbf{QI}|c)$, $P(\mathbf{Size}|c)$, $P(\mathbf{Duality}|c)$ and $P(\mathbf{USSX}|c)$ scaled for each c.

Figure 2: The compression-approximation trade-off for the Governance cube.

Size groups and USSX values. The deviance for this model is $G^2 = 23.06$, with 21 degrees of freedom, and a p-value of 0.341. This means that this parsimonious model is not significantly different from the saturated model in terms of approximation, but it uses only 26 (48 - 21 -1) free parameters instead of 48.

4.2 Compression and Approximation

Figure 2 illustrates the well-known trade-off between compression and approximation on the Governance cube. The X-axis is the number of parameters in the model (lower means more compression) while the Y-axis is the likelihood ratio (lower means better approximation). The dotted lines are contours of the AIC. An ideal model with good approximation using few parameters would be in the bottom left corner.

From this figure, one can see that the *independence* and *saturated* models are two extremes, and other probabilistic models are positioned in between. Moreover, as the number of NMF components increases, the approximation improves, but the number of parameters grows. One model that reaches a good compromise between compression and approximation is the 3-component model, which has the best AIC of all NMF models. Note that the parsimonious log-linear model reaches a different, and seemingly better (for *that* cube) compromise.

In order to empirically analyze the trade-off between compression and approximation, we consider our three example cubes and three models (NMF with BIC, NMF with AIC and LLM), and compute for each model the compression rate R_c (Equation 5) and the deviance G^2.

Table 3 shows that in the Governance cube, all models compress the data by one half to two-third. The Customer cube compresses well, from 76.4% for LLM to about 95% for
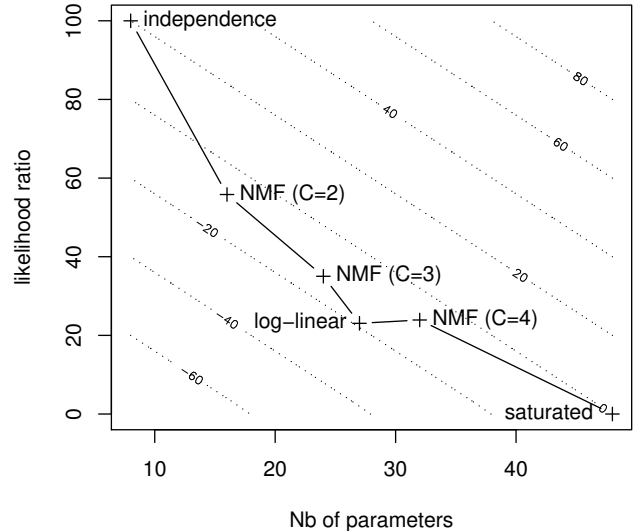
NMF. Finally, the Sales cube is harder to compress. Using BIC, the NMF yields a compression rate of only 26% for the third cube, while LLM does not manage to compress the data since there exists a three-way interaction between the three existing dimensions.

These experimental results confirm that BIC yields a more parsimonious representation than AIC. This is known from theoretical considerations, as the BIC penalty is larger. However, results also seem to indicate that NMF tends to compress the data more than LLM. This comes of course at the price of a worse approximation, as indicated by G^2 in Table 3. In all cases, the deviance is much larger for the BIC-optimal NMF than for the AIC-optimal NMF and LLM, indicating a worse fit.

These results suggest that NMF and LLM are able to balance the conflicting goals of compression and faithful representation of the data in different ways and to different degrees, with LLM focusing more on good approximation, while NMF produces more compact representations of the data. In the most extreme case presented here, the BIC-optimal NMF represents the 2400 cells of the Customer cube using only 109 free parameters. This represents a very large space saving.

The approximation of cubes allows also a better understanding and an easier exploration of data as indicated below.

4.3 Approximate Query Answering

In this section, we show how OLAP queries [8] may be answered within the NMF framework using information about the generated components. We consider the following operators: selection (*Dice*), projection (*Slice*) and aggregation (*Roll-Up*). Many other OLAP operators (*Rotate*, *switch*, etc.) are concerned primarily with visualization and/or permutations of dimensions and modalities, and therefore do not carry any significant change to the model.

A *Roll-up* will aggregate values either over all the modalities of one or several dimensions, or over subsets of modal-

GOVERNANCE (N_c=3x4x2x2), N=214

Model	sub-cubes	param.	R_c	G^2
NMF (best BIC)	2	16	66.7%	56
NMF (best AIC)	3	24	50.0%	35
LLM	2	26	45.8%	23

CUSTOMER (N_c=2x8x6x5x5), N=10281

Model	sub-cubes	param.	R_c	G^2
NMF (best BIC)	5	110	95.4%	1020
NMF (best AIC)	6	132	94.5%	917
LLM	4	567	76.4%	595

SALES (N_c=44x4x3), N=5191

Model	sub-cubes	param.	R_c	G^2
NMF (best BIC)	8	392	25.8%	715
NMF (best AIC)	-	528	0%	0
LLM	-	528	0%	0

Table 3: Output of NMF and LLM for three different data cubes. N_c **is the number of cells in the table, N is the number of records,** R_c **is the compression rate, and** G^2 **is the deviance.**

		Modalities				
Dimension	Data	C_1	C_2	C_3	C_4	C_5
STATUS	1, 2	1, 2	1, 2	1, 2	1, 2	1, 2
INCOME	1-8	4-8	1-3	1-3	2, 3	1-4, 6, 8
CHILD	0-5	0-5	0-5	0-5	0-5	0-5
OCCUPATION	1-5	4, 5	1-5	1, 2	1, 2	4, 5
EDUCATION	1-5	1-5	3	1, 2	1-3	4, 5

Table 4: Modalities of the 5-dimensional data cube "Customer" and its five components C_1 to C_5. The following convention is used: i, j **means that only modalities** i **and** j **are non-zero for the corresponding dimension and component, and** $i - j$ **means all modalities between** i **and** j.

ities. This is easily implemented on the model by summing over the corresponding probabilistic profile. In addition, when a roll-up is performed over all modalities of a dimension, by definition the sum over the entire probabilistic profile is 1, i.e. $\sum_k P(k|m) = 1$, meaning that this probabilistic profile is simply dropped from the model. An example will clarify this. Without loss of generality, let us assume a three-dimensional cube $\mathbf{X} = [x_{ijk}]$ with corresponding 3D NMF model $P_x(i, j, k) = \sum_m P(m)P(i|m)P(j|m)P(k|m)$. A roll-up over dimension k will result in a 2-dimensional cube $\mathbf{Y} = [y_{ij}]$. The corresponding NMF model is $P_y(i, j) = \sum_k \sum_m P(m)P(i|m)P(j|m)P(k|m) = \sum_m P(m)P(i|m)P(j|m)$. The model estimate for the rolled-up cube $\mathbf{Y}$ is therefore obtained "for free" from the model of the original cube $\mathbf{X}$ with essentially no additional computation. Note that the resulting model for cube $\mathbf{Y}$ is not optimal in the sense that it does not necessarily maximize the likelihood on that cube. However, the resulting estimated counts $\widehat{y}_{ij} = N.P_y(i, j)$ (with $N = \sum_{ijk} x_{ijk}$) are *exactly* the same as what we would obtain by first forming the estimated cube for $\mathbf{X}$ using $\widehat{x}_{ijk} = N.P_x(i, j, k)$, and doing a roll-up on that cube. Doing the roll-up on the model lightens the computational requirements. In the example given above, instead of order $4.M.I.J.K$ operations[3] for forming the entire cube and $I.J.K$ for the aggregation operation, only $3.M.I.J$ operations are required.

The *Slice* and *Dice* operations are similar in the sense that they select a portion of the cube and differ mainly on the extent of the selection. The availability of a probabilistic model potentially cuts the processing in at least two ways. First, only the necessary cells may be approximated, instead of computing the entire approximate cube and performing the *Slice* or *Dice* explicitly on the full cube. Second, during the approximation, it is possible to discard the components that are not relevant to the current query.

As an illustration we will use the NMF model of five components selected by BIC for the CUSTOMER cube. Table 4

---

[3] Recall that M stands for the number of components.

shows modalities of the CUSTOMER cube and its five components C_1 to C_5. The table shows, for each component, the modalities of each dimension that are "active", i.e. have a probability significantly higher than zero. For example, in component C_2, all values of OCCUPATION are probable, but for components C_3 and C_4, values of OCCUPATION higher than 2 have a zero probability. This table will be helpful in the design of our OLAP query plans. For example, when the query involves cells with INCOME=3, component C_1 may be ignored; for OCCUPATION=3, the only relevant component is C_2.

Let us consider the following queries:

1. *Slice*: Number of customers according to STATUS, INCOME, CHILD, and OCCUPATION for customers with EDUCATION= 4 (bachelor's degree).

2. *Dice*: Number of customers according to STATUS, INCOME, and OCCUPATION for customers with EDUCATION= 4 (bachelor's degree) or 5 (graduate studies) where CHILD $\in \{3, 4, 5\}$.

3. *Roll-Up$_1$*: Number of customers according to INCOME, OCCUPATION, and EDUCATION (a subset of dimensions of the CUSTOMER cube).

4. *Roll-Up$_2$*: Number of customers according to the five dimensions (STATUS, INCOME, CHILD, OCCUPATION and EDUCATION) with a roll-up on the INCOME hierarchy (e.g., three intervals: 1-3, 4-5, and 6-8).

The first two queries are processed using only components C_1 and C_5—all other are zero for EDUCATION=4 or EDUCATION=5. The *Dice* query can then be processed as follows: (i) perform *Dice* query (i.e., selection on EDUCATION and CHILD) on C_1 to get C_1'; (ii) compute, into C_5', the number of customers according to STATUS, INCOME, and OCCUPATION for customers with CHILD $\in \{3, 4, 5\}$; (iii) add C_1' to C_5' to get the approximate answer.

For the third query, a three dimensional model is available directly from the original five dimensional NMF by dropping the profiles for CHILD and EDUCATION from the model. All five components are used, but for three dimensions only.

Finally, the fourth query concerns a roll-up on the hierarchy of the INCOME dimension. It can be answered by selecting the appropriate components and aggregating data for each interval of INCOME. In that way, irrelevant components can be ignored for each interval of INCOME (component C_1 for interval 1-3, and C_2 to C_4 for 4-5 and 6-8).

Table 5 reports mean (mE) and standard deviation (ΔE) of the approximation error (absolute difference) observed

Query	Error			
	mE	ΔE	mE_0	ΔE_0
Slice	0.98	2.23	0.08	0.35
Dice	2.20	4.83	0.00	0.00
Roll-Up$_1$	2.77	5.93	0.00	0.00
Roll-Up$_2$	1.49	3.12	0.05	0.43

Table 5: Mean (mE) and standard deviation (ΔE) of the absolute error by cell. mE_0 and ΔE_0 are the mean and standard deviation on the cells with zero count.

between the result of each query performed on the original data, and the approximation calculated as presented above, directly on the model. We also check whether the model approximates well empty cells in the cube by calculating the approximation error limited to these empty cells (mE_0 and ΔE_0).

Table 5 shows that, mean absolute error by cell (mE) is the smallest for *Slice* query. This situation is explained by the fact that *Slice* query processing does not need cells aggregation which implies error accumulation. In *Roll-Up$_2$* query, error is partially accumulated since a partial aggregation is performed while building intervals on INCOME. (mE_0) and (ΔE_0) values are very small, which indicates that zero-cells are generally well-estimated.

5. DISCUSSION

The present work shows that NMF and LLM display similarities and differences. As NMF and LLM are probabilistic models, they associate a probability to each cell in the data cube. This is very helpful to detect outliers by comparing the observed count in one cell with the expected frequency according to the modeled probability (see [15, 24]).

Both models break down a complex data analysis problem into a set of smaller sub-problems. In the context of data warehousing, this means that instead of exploring a large multi-dimensional data cube, the user can analyze a few sets of smaller cubes. This reduces distracting and irrelevant elements and eases the extraction of actionable knowledge.

NMF can identify homogeneous dense regions inside a sparse data cube and find *relevant modalities* within each dimension for each sub-cube. LLM, on the other hand, can identify *important correlations* between dimensions. While NMF expresses the original data cube as a *superposition* of several homogeneous sub-cubes, LLM expresses the original data cube as a *decomposition* into a set of sub-cubes expressing strong associations among dimensions. In both cases, the sub-cubes allow the user to focus on one particular component/association at a time rather than on the population as a whole.

Although both models can handle a variety of multi-way arrays, there are also key differences in the way they use their parameters [15]. In NMF, the number of parameters scales linearly in the number of components, and the scaling factor is the sum of the number of modalities on all dimensions (up to a small constant, e.g., $I + J + K - 2$ for three dimensions). On the other hand, the number of parameters in LLM depends on lower-order products of the number of modalities in each of the dimensions involved in the retained interactions. As a consequence, LLM is better suited for modeling tables with many dimensions, each with rela-

tively few modalities, like in some data cubes. NMF will get the largest benefits from situations where there are relatively few dimensions, each with large numbers of modalities, such as textual data.

6. CONCLUSION

In this paper we advocate the use of non-negative multi-way array factorization for the approximation, compression and exploration of data cubes. We also compare this technique with log-linear modeling for cube approximation and compression. Both techniques reach a compromise between smaller memory footprint and good approximation. Finally, we show how NMF can handle OLAP query answering by selecting the appropriate components to consider. This is particularly useful for queries involving selection and/or roll-up on dimensions.

Our future work concerns the following topics: (i) improve the efficiency of the model selection and parameter estimation procedures in order to scale up to very large data cubes, (ii) incrementally update an NMF model when a set of aggregate values are added to the data cube such as a new dimension or new members of a dimension (e.g., a new month); and (iii) explore and intensively experiment the potential of NMF for approximating large data sets, and for preprocessing data to get homogeneous and dense sub-cubes that could then be analyzed individually using LLM for example.

7. ACKNOWLEDGMENTS

8. REFERENCES

[1] H. Akaike. A new look at the statistical model identification. *IEEE Transactions on Automatic Control*, 19(6):716–723, 1974.

[2] J. L. Ambite, C. Shahabi, R. R. Schmidt, and A. Philpot. Fast approximate evaluation of olap queries for integrated statistical data. In *Proceedings of the First National Conference on Digital Government Research*, 2001.

[3] B. Babcock, S. Chaudhuri, and G. Das. Dynamic sample selection for approximate query processing. In *SIGMOD '03: Proceedings of the 2003 ACM SIGMOD international conference on Management of data*, pages 539–550, New York, NY, USA, 2003. ACM Press.

[4] D. Barbara and X. Wu. Using loglinear models to compress datacubes. In *WAIM '00: Proceedings of the First International Conference on Web-Age Information Management*, pages 311–322, London, UK, 2000. Springer-Verlag.

[5] D. Barbara and X. Wu. Loglinear-based quasi cubes. *J. Intell. Inf. Syst.*, 16(3):255–276, 2001.

[6] A. Boujenoui and D. Zéghal. Effet de la structure des droits de vote sur la qualité des mécanismes internes de gouvernance: cas des entreprises canadiennes. *Canadian Journal of Administrative Sciences*, 23(3):183–201, 2006.

[7] K. Chakrabarti, M. Garofalakis, R. Rastogi, and K. Shim. Approximate query processing using wavelets. *The VLDB Journal*, 10(2-3):199–223, 2001.

[8] S. Chaudhuri and U. Dayal. An overview of data warehousing and olap technology. *SIGMOD Rec.*, 26(1):65–74, 1997.

[9] R. Christensen. *Log-linear Models*. Springer-Verlag, New York, 1997.

[10] W. Deming and F. Stephan. On a least squares adjustment of a sampled frequency table when the expected marginal totals are known. *The Annals of Mathematical Statistics*, 11(4):427–444, 1940.

[11] A. P. Dempster, N. M. Laird, and D. B. Rubin. Maximum likelihood from incomplete data via the EM algorithm. *Journal of the Royal Statistical Society, Series B*, 39(1):1–38, 1977.

[12] I. S. Dhillon and S. Sra. Generalized nonnegative matrix approximations with bregman divergences. In *Advances in Neural Information Processing Systems 17*, pages 283–290, 2005.

[13] V. Ganti, M.-L. Lee, and R. Ramakrishnan. Icicles: Self-tuning samples for approximate query answering. In *VLDB '00: Proceedings of the 26th International Conference on Very Large Data Bases*, pages 176–187, San Francisco, CA, USA, 2000. Morgan Kaufmann Publishers Inc.

[14] E. Gaussier and C. Goutte. Relation between PLSA and NMF and implications. In *SIGIR '05: Proceedings of the 28th annual international ACM SIGIR conference on Research and development in information retrieval*, pages 601–602, New York, NY, USA, 2005. ACM Press.

[15] C. Goutte, R. Missaoui, and A. Boujenoui. Data cube approximation and mining using probabilistic modelling. Technical Report NRC 49284, National Research Council, 2007. http://iit-iti.nrc-cnrc.gc.ca/publications/nrc-49284_e.html.

[16] S. Haberman. *Analysis of qualitative data, Volume 1*. Academic Press, New York, 1978.

[17] T. Hofmann. Probabilistic latent semantic analysis. In *UAI'99*, pages 289–296. Morgan Kaufmann, 1999.

[18] T. Imielinski, L. Khachiyan, and A. Abdulghani. Cubegrades: Generalizing association rules. *Data Min. Knowl. Discov.*, 6(3):219–257, 2002.

[19] L. V. S. Lakshmanan, J. Pei, and Y. Zhao. Quotient cube: How to summarize the semantics of a data cube. *In Proceedings of the 28th International Conference on Very Large Databases, VLDB*, pages 778–789, 2002.

[20] D. D. Lee and H. S. Seung. Learning the parts of objects by non-negative matrix factorization. *Nature*, 401:788–791, 1999.

[21] H. Lu, L. Feng, and J. Han. Beyond intratransaction association analysis: mining multidimensional intertransaction association rules. *ACM Trans. Inf. Syst.*, 18(4):423–454, 2000.

[22] S. Naouali and R. Missaoui. Flexible query answering in data cubes. In *DaWaK*, pages 221–232, 2005.

[23] T. Palpanas, N. Koudas, and A. Mendelzon. Using datacube aggregates for approximate querying and deviation detection. *IEEE Transactions on Knowledge and Data Engineering*, 17(11):1465–1477, 2005.

[24] S. Sarawagi, R. Agrawal, and N. Megiddo. Discovery-driven exploration of olap data cubes. In *EDBT '98: Proceedings of the 6th International Conference on Extending Database Technology*, pages 168–182, London, UK, 1998. Springer-Verlag.

[25] G. Schwartz. Estimating the dimension of a model. *The Annals of Statistics*, 6(2):461–464, 1978.

[26] J. Shanmugasundaram, U. Fayyad, and P. S. Bradley. Compressed data cubes for olap aggregate query approximation on continuous dimensions. In *KDD '99: Proceedings of the fifth ACM SIGKDD international conference on Knowledge discovery and data mining*, pages 223–232. ACM Press, 1999.

[27] J. S. Vitter and M. Wang. Approximate computation of multidimensional aggregates of sparse data using wavelets. In *Proceeding of the SIGMOD'99 Conference*, pages 193–204, 1999.

[28] M. Welling and M. Weber. Positive tensor factorization. *Pattern Recognition Letters*, 22(12):1255–1261, 2001.

[29] F. Yu and W. Shan. Compressed data cube for approximate olap query processing. *J. Comput. Sci. Technol.*, 17(5):625–635, 2002.