CMPUT 695 Fall 2004 Assignment 2 – Xelopes

Paul Nalos, Ben Chu

November 5, 2004

1 Introduction

We evaluated Xelopes, a data mining library produced by prudsys[1].

Xelopes is available for Java, C++, and CORBA in both GPL and high-performance non-GPL variants.

While Xelopes provides several different mining patterns models, we focused on association analysis, classification, clustering, and sequence analysis. This covers the bulk of Xelopes at a high level.

2 General Evaluation

We identified the following criteria as desirable in a data mining solution; Xelopes is evaluated against each point.

[1] http://www.prudsys.com/

Criteria	Evaluation
available platforms	excellent – cross platform (Java, C++, CORBA)
number of mining patterns	very good; includes Association, Classification, Clustering, Regression, Sequential, Statistics, Supervised, Time Series Prediction
number of algorithms per mining pattern	significant, varies by area
performance (memory and disk) vs. input data size	varies by algorithm
input formats	ARFF, CSV (highly configurable), PMML, relational (JDBC), extensible; ARFF and PMML include rich metadata and offer interoperability
output formats	PMML, somewhat extensible, mining results can be fed into other mining algorithms
supported standards	Xelopes supports a number of standards, and is heavily based on CWM (the Common Warehouse Metamodel).
API – learning curve	non-trivial
API – intuitive	yes, but complicated
API – flexible	outstanding flexibility
UI – responsive, provides feedback	N/A
UI – well designed, intuitive	N/A
UI – common operations require few steps and no data reentry	N/A
visualization	N/A
source available	yes (GPL)
cost (product and training / consulting)	The GPL version is free; we did not get a quote for the commercial version or for training and support.

2.1 Strengths

One major strength of Xelopes is its availability for many platforms. This allows us to ship our products based on different technologies for many different platforms.

In comparison to the other data mining tools under consideration, Xelopes provides good coverage of several data mining patterns rather than restricting the user to one or two specialized analyses.

Xelopes' input and output flexiblity ensures that it will be easy to integrate

our products with many different systems. Its standards compliance will reduce the cost of integrating with other standards compliant systems.

A key strength of Xelopes is its facility to automatically determine the parameters for various data mining algorithms. Many algorithms require parameter tuning which is time consuming and may require domain knowledge.

2.2 Weaknesses

Our analysis of Xelopes' performance, while inconclusive, raised some questions as to its scalability. A more detailed evaluation of Xelopes' performance is contained in the next section.

A definite weakness of the package is its lack of a user interface or any visualization capability.

The steep learning curve of the API could potentially increase development costs.

3 Mining Pattern Evaluation

3.1 Association Analysis

I used the IBM data generator[2] to produce 100k transactions, with all other parameters at their default values. I compared Xelopes, my course implementation, and Christian Borgelt's implementation[3] on a 500 MHz G3 Apple laptop.

With 80% confidence, and 0.1% support, the results were:

	Time	**Peak Real Memory**
My Implementation	1m 42s	183M
CB Implementation	12s	20M
Xelopes	>30m	>110M

I stopped Xelopes after 30 minutes. Clearly, this is an unacceptable result. The GPL version of Xelopes provides a suboptimal implementation of Apriori, which is what is measured here. The commercial version includes an optimized version of Apriori-Hybrid, which will hopefully perform much better.

I repeated the experiment with higher and higher support. Eventually, I increased the support to 1%, which produces no frequent itemsets or strong rules. Xelopes was able to compute this (null) result in 1m 55s; the complete test runs in 3m 26s due to some extra IO. Both other implementations complete in less than 8 seconds. This makes me suspect there may be significant overheads for using Xelopes with medium or large data sets, although this may not be the case.

[2] http://www.almaden.ibm.com/software/quest/Resources/datasets/syndata.html

[3] http://fuzzy.cs.uni-magdeburg.de/~borgelt/apriori.html (compiled from source)

3.2 Classification

Xelopes provides a number of algorithms for classification, including various types of decision trees, linear classification and regression, non-linear classification and regression, neural networks, support vector machines, and sparse grids.

I tested simple classification using axis-parallel decision trees. Xelopes comes with an example of classification of soybean plants. This data set includes 683 instances and 35 categorical attributes. Xelopes is able to train on part of the data set and validate the generated model against the other part in 20s, with a peak real memory usage of 20M. I scaled the training data by 10x using duplication, and the same test completed in 27s with no change in peak real memory usage. I conclude that Xelopes performs well, with this classification algorithm, on very small data sets.

I then repeated this experiment with Census data[4] from the UCI archive[5]. Some effort was required to massage the data and sample program to make this work.

This data set contains 199,523 instances, and the test set contains 99,762 instances. There are 41 categorical attributes. The test did not complete after 1 hour of computation on a 500 MHz G3 Apple laptop (real memory usage peaked at 310M), nor after 107 minutes of CPU time on a high-performance Intel Linux server (real memory usage peaked at 370M). I don't have a sufficient comparison point to know whether or not this is reasonable.

3.3 Clustering

Xelopes implements three types of the major clustering methods:

- Hierarchical Agglomerative Clustering
- Partitioning Clustering (k-Linkage)
- Center-based Clustering (k-Means)

The availability of algorithms for each of these methods varies. Hierarchical clustering is implemented through both a simple and a fast agglomerative algorithm. The fast algorithm runs in far less time but takes up twice as much memory. One k-linkage algorithm is provided for partitioning clustering and one k-means algorithm is provided for the center-based clustering method.

I evaluated the performance of the k-means algorithm center-based clustering method. I tested the algorithm using three different datasets on a 550 MHz G4 Processor with 256M RAM. The first dataset is the standard Iris plant atttributes used for clustering examples that comes with Xelopes. The second is the same dataset scaled 10 times. The third dataset is the same U.S. census data we used to test Xelopes' classification. The first two datasets were tested with k=3, while the census data used k=2.

[4] http://kdd.ics.uci.edu/databases/census-income/census-income.html
[5] http://kdd.ics.uci.edu/

Dataset	# Instances	Time	Peak Real Memory
Iris	150	9.169s	20M
Irisx10	1500	10.853s	20M
Census-Income	199,153	40m 45s	>70M

The clustering k-means algorithm completed within what seemed to be a reasonable amount of time on a very large dataset. However, there is no basis for comparison so this data is not necessarily meaningful.

3.4 Sequence Analysis

Xelopes allows for either simple sequential analysis or sequential basket analysis to be performed. The fact that Xelopes contains any models and algorithms for sequential analysis already puts it ahead of several other data mining tools in this regard. A cursory examination of the features of the other data mining tools under consideration shows that few, if any, support sequential pattern analysis.

I tested the simple sequential algorithm (based on Apriori) with a sample dataset of user clickthroughs on a production website that is included with Xelopes. The dataset consists of 659 sessions, with one or more items per session. On a G4 550 Mhz Processor with 256M RAM, the sequential model for the dataset was built in 18s, with a peak real memory usage of 20M.

Unfortunately, the format of the sequential dataset allowed to be processed through the input stream appears to be fairly specialized, making it difficult to discover similarly formatted datasets elsewhere for testing with the algorithm.

4 Conclusion

This product is very flexible, conforms to a number of standards, and provides a significant number of data mining algorithms. It is well factored, allowing the user to change between different types of algorithms and data formats with minimal rework. If the need is to embed data mining capabilities into our Java applications, Xelopes is a good fit.

There are some concerns raised by our analysis of the Java / GPL version of Xelopes. First, if we choose to deploy the GPL versions in our commercial products, we will be forced to release them under the GPL (i.e. release our source code and give away redistribution rights), which would detroy our ability to sell them for a profit. Second, we did observe unacceptable performance with the Java / GPL version for association analysis of reasonably sized data sets. The non-GPL versions comes with additional, optimized algorithms.

One advantage of the GPL versions is that they are free. The price for the non-GPL versions is not listed on the prudsys website; a quote would be required.

Additionally, Xelopes is not directly useful for non-embedded purposes. It comes with no UI or visualization capabilities. While the interface is quite flexible, it takes significant time to configure the system to execute a new data mining task (using existing data and algorithms).

Therefore, the GPL version is only worth acquiring if the user can spend the time and money to:

- learn the Xelopes framework,
- implement their own efficient mining algorithms,
- implement their own data interpretation/visualization tools as needed,
- and develop a product and release it under the GPL.

Based on these criteria, the GPL version does not meet our needs and should be rejected as a candidate for acquisition.

As an alternative, the non-GPL version should be considered after validating prudsys' performance claims.

4.1 Ideal Data Mining Tool

The negative and positive points of Xelopes lead into our opinion of what constitutes a good data mining tool:

- Programmatic or not, the tool should have a fairly easy learning curve and be distributed with concise and correct documentation.
- The tool should include a variety of efficient and robust data mining algorithms.
- The tool should be flexible, but not so much so that it becomes over generalized and is only mediocre at everything.
- The tool should have flexibility with regards to the input format of data, or have **one** file format specification.