Performance Improvement of Model Building in Oracle 11.1 Data Mining

An Oracle White Paper
May 2008
Performance Improvement of Model Building in Oracle 11.1 Data Mining

Executive Overview ................................................................. 3
Introduction .............................................................................. 3
Test Environment and Data .................................................... 4
MDL ......................................................................................... 5
Naive Bayes ............................................................................ 5
K-Means .................................................................................. 6
NMF ........................................................................................ 7
SVM ......................................................................................... 7
Results Summary ...................................................................... 8
Conclusion ............................................................................... 9
Appendix .................................................................................. 9
  Init.ora Parameters for Oracle Database 10g Release 2 ...... 9
  Init.ora Parameters for Oracle Database 11g Release 1 ... 10
Performance Improvement of Model Building in Oracle 11.1 Data Mining

Executive Overview

For many successful enterprises, data mining applications drive business initiative and generate revenue opportunities. The performance of data mining applications directly impacts their bottom line.

The data mining process involves building a model using a training data set as the first step. The model is then applied to new data sets to generate predictions. As new data comes in, models may need to be re-built before they can be re-applied. Model building is an iterative process.

Oracle Database 11g Release 1 has improved, and in some cases dramatically improved, the performance of model building in data mining. This technical white paper demonstrates the performance improvements by comparing model building in Oracle Database 11g release 1 with model building in Oracle Database 10g Release 2.

Introduction

Data mining is a technique that discovers previously unknown patterns and trends in large stores of data. For many successful organizations and businesses, data mining applications help managers and executives gain insights into their organization or business, make future predictions, and better understand their customers. Data mining applications have become so mission critical that their performance directly impacts the bottom line.

The data mining process involves building a model using a training data set as the first step. The model is then applied to new data sets to generate predictions.\(^1\)

Creating a data-mining model is a dynamic and iterative process. In many cases, after a model has already been built, one may need to rebuild the model on more recent data.

\(^1\) See the white paper “High Performance Scoring with Oracle 10.2 Data Mining” for scoring performance.
A wide range of high performance model-building techniques is offered by Oracle Data Mining to solve a wide variety of business problems. For supervised learning algorithms, Oracle Data Mining provides Naïve Bayes, Decision Trees, Support Vector Machines (SVM), Minimum Description Length (MDL) for attribute importance, and Generalized Linear Models (GLM)\(^2\). For unsupervised learning algorithms, Oracle Data Mining offers Enhanced k-Means clustering algorithm, Orthogonal Partitioning Clustering (O-Cluster), A Priori association-rules algorithm, and Non-negative Matrix Factorization (NMF) feature-selection algorithm.

In Oracle Database 11g Release 1, the performance of many model-building algorithms, namely, MDL, Naïve Bayes, Enhanced k-Means, NMF, and SVM, has been further improved, and in some cases, dramatically improved, while the performance of all other algorithms remains unchanged.

To highlight the performance improvements, we have run some tests comparing the model building in Oracle Database 11g Release 1 with model building in Oracle Database 10g Release 2. This technical white paper details these tests and performance comparison results.

In the next section, we briefly explain the test environment and the data set used for the tests. Each subsequent section details the model building tests and performance results of one data mining algorithm. The performance results of all algorithms are summarized in the end.

**Test Environment and Data**

Tests in this white paper were conducted on a system with 2x2.8 GHz Intel Xeon processors with Hyper-Threading technology simulating 4 CPUs. The system has 7 GB of memory. The operating system is Linux Red Hat release 4. The databases were created on an ASM disk group built with 10 disks in a single disk array (EMC Clariion CX300).

A 252-column (including an ID column and a target column) 100,000-row data set was used in all test cases. The target column in this data set has a binary value. The data has already been normalized. It is stored in a table called “wpdata”. In some cases, the data set needs to be binned before it can be used. Table “wpdata_binned” stores the same data after binning.

\(^2\) Generalized Linear Models is a new supervised learning algorithm offered in Oracle Database 11g Release 1.
MDL

Minimum Description Length (MDL) is a supervised algorithm that ranks attributes according to their significance in predicting a target. Oracle Data Mining uses MDL when users select attribute importance mining function.

To rank the 250 attributes in our testing data set, we invoke the model creation procedure with the attribute importance mining function specified. It uses the binned data. The following code was used in both Oracle Database 11g Release 1 and Oracle Database 10g Release 2.

```
BEGIN
  DBMS_DATA_MINING.CREATE_MODEL(
    model_name => 'AI_wp',
    mining_function =>
    dbms_data_mining.attribute_importance,
    data_table_name => 'wpdata_binned',
    case_id_column_name => 'id',
    target_column_name => 'target');
END;
/
```

As shown in the following table, building an attribute importance model using this data set in Oracle Database 11g Release 1 is 26 times faster than Oracle Database 10g Release 2.

<table>
<thead>
<tr>
<th></th>
<th>Elapsed Time (Sec.)</th>
</tr>
</thead>
<tbody>
<tr>
<td>Oracle Database 11g Release 1</td>
<td>16</td>
</tr>
<tr>
<td>Oracle Database 10g Release 2</td>
<td>423</td>
</tr>
</tbody>
</table>

Naive Bayes

Naive Bayes is one of the classification algorithms supported by Oracle Data Mining. The algorithm can be used for both binary and multi-class classification problems. Our test case builds a model over the testing data set for binary classification.

The code below is used for both Oracle Database 11g Release 1 and Oracle Database 10g Release 2. It uses the binned data. Naïve Bayes is the default algorithm when users specify classification as mining function.
BEGIN
  DBMS_DATA_MINING.CREATE_MODEL(
    model_name => 'NB_wp',
    mining_function => dbms_data_mining.classification,
    data_table_name => 'wpdata_binned',
    case_id_column_name => 'id',
    target_column_name => 'target');
END;
/

The elapsed time results for both Oracle Database 11g Release 1 and Oracle Database 10g Release 2 are captured in the following table.

<table>
<thead>
<tr>
<th></th>
<th>Elapsed Time (Sec.)</th>
</tr>
</thead>
<tbody>
<tr>
<td>Oracle Database 11g Release 1</td>
<td>16</td>
</tr>
<tr>
<td>Oracle Database 10g Release 2</td>
<td>264</td>
</tr>
</tbody>
</table>

In Oracle Database 11g Release 1, the performance of model building for Naïve Bayes algorithm has been dramatically improved – Oracle Database 11g Release 1 is 17 times faster than Oracle Database 10g Release 2 for this test case.

K-Means

The k-Means algorithm is a distance-based clustering algorithm that partitions the data into a predetermined number of clusters (provided there are enough distinct cases). Our test case builds a k-Means clustering model using default settings with 10 clusters.

The following code is shared between Oracle Database 11g Release 1 and Oracle Database 10g Release 2. K-Means is the default algorithm when users specify clustering mining function.

BEGIN
  DBMS_DATA_MINING.CREATE_MODEL(
    model_name => 'KM_wp',
    mining_function => dbms_data_mining.clustering,
    data_table_name => 'wpdata',
    case_id_column_name => 'id');
END;
/
As seen in the table below, Oracle Database 11g Release 1 is 2.5 times faster than Oracle Database 10g Release 2 for this test case.

<table>
<thead>
<tr>
<th></th>
<th>Elapsed Time (Sec.)</th>
</tr>
</thead>
<tbody>
<tr>
<td>Oracle Database 11g</td>
<td>118</td>
</tr>
<tr>
<td>Oracle Database 10g</td>
<td>295</td>
</tr>
</tbody>
</table>

NMF

Non-negative Matrix Factorization (NMF) is a state of the art feature extraction algorithm offered by Oracle Data Mining.

The code below builds an NMF model, and it was used in both Oracle Database 11g Release 1 and Oracle Database 10g Release 2.

```
BEGIN
  DBMS_DATA_MINING.CREATE_MODEL(
    model_name      => 'NMF_wp',
    mining_function => dbms_data_mining.feature_extraction,
    data_table_name => 'wpdata',
    case_id_column_name => 'id');
END;
/
```

As shown in the table above, Oracle Database 11g Release 1 is about 2 times faster than Oracle Database 10g Release 2 for this test case.

SVM

Support Vector Machines (SVM) is a supervised algorithm that can be used for both classification and regression tasks. The current implementation supports two kernel types – Linear and Gaussian.
In Oracle Database 11g Release 1, the performance of model building using SVM has been improved as well. We use SVM classification with Linear kernel as an example. We used the same code in Oracle Database 11g Release 1 and Oracle Database 10g Release 2 to build an SVM model.

```
CREATE_TABLE svmcl_settings (  
    setting_name VARCHAR2(30),  
    setting_value VARCHAR2(30));

BEGIN  
  INSERT INTO svmcl_settings VALUES  
    (dbms_dataMining.algo_name,  
    dbms_dataMining.algo_support_vector_machines);  
  INSERT INTO svmcl_settings VALUES  
    (dbms_dataMining.svms_kernel_function,  
    dbms_dataMining.svms_linear);  
  COMMIT;
END;
/

BEGIN  
  DBMS_DATA_MINING.CREATEMODEL(  
    model_name => 'SVM_Clas',  
    mining_function => dbms_dataMining.classification,  
    data_table_name => 'wpdata',  
    case_id_column_name => 'id',  
    target_column_name => 'target',  
    settings_table_name => 'svmcl_settings');  
END;
/

<table>
<thead>
<tr>
<th>Elapsed Time (Sec.)</th>
</tr>
</thead>
</table>
| Oracle Database 11g Release 1 | 150  
| Oracle Database 10g Release 2 | 363  

As shown in the above table, Oracle Database 11g Release 1 is 2.4 times faster than Oracle Database 10g Release 2 for this SVM model building test case.

**Results Summary**

We summarize the elapsed time and performance improvement results for all algorithms we have presented in this paper. In the following table, the elapsed time
results are shown in seconds. The improvements are calculated as (Elapsed Time in Oracle 10.2 / Elapsed Time in Oracle 11.1).

<table>
<thead>
<tr>
<th></th>
<th>MDL</th>
<th>Naïve Bayes</th>
<th>k-Means</th>
<th>NMF</th>
<th>SVM</th>
</tr>
</thead>
<tbody>
<tr>
<td>Oracle Database 11g Release 1 (Sec.)</td>
<td>16</td>
<td>16</td>
<td>118</td>
<td>186</td>
<td>150</td>
</tr>
<tr>
<td>Oracle Database 10g Release 2 (Sec.)</td>
<td>423</td>
<td>264</td>
<td>295</td>
<td>365</td>
<td>363</td>
</tr>
<tr>
<td>Improvements (Elapsed Time in 10.2 / Elapsed Time in 11.1)</td>
<td>26</td>
<td>17</td>
<td>2.5</td>
<td>2</td>
<td>2.4</td>
</tr>
</tbody>
</table>

**Conclusion**

As we have demonstrated through test cases in this technical white paper, the performance of model building in Oracle Database 11g Release 1 data mining has been improved, and in some cases, dramatically improved. Upgrading to Oracle Database 11g Release 1 can greatly improve the performance of data mining applications.

**Appendix**

*Init.ora Parameters for Oracle Database 10g Release 2*

```plaintext
aq_tm_processes = 0
audit_trail = false
compatible = 10.2.0.0
control_files = (+1jdgroupl/lj_control_001, +1jdgroupl/lj_control_002)
db_block_checksum = false
db_block_size = 8192
db_file_multiblock_read_count = 128
db_files = 500
db_name = wp102
db_writer_processes = 4
dml_locks = 5000
global_names = false
log_buffer = 4194304
log_checkpoints_to_alert = true
max_dump_file_size = unlimited
```
nls_date_format = YYYY-MM-DD
open_cursors = 600
optimizer_mode = CHOOSE
parallel_adaptive_multi_user = false
parallel_max_servers = 64
parallel_min_servers = 32
pga_aggregate_target = 4G
processes = 400
query_rewrite_enabled = true
recovery_parallelism = 8
replication_dependency_tracking = false
sga_max_size = 2G
sga_target = 2G
statistics_level = typical
transactions = 100
undo_management = auto
undo_retention = 400000
recovery_parallelism = 1
disk_async_io = true
filesystemio_options = SetAll

Init.ora Parameters for Oracle Database 11g Release 1

aq_tm_processes = 0
audit_trail = false
compatible = 11.0.0.0
control_files = (+ljdgroup1/lj_control_001, +ljdgroup1/lj_control_002)
db_block_checksum = false
db_block_size = 8192
db_file_multiblock_read_count = 128
db_files = 500
db_name = wp11
db_writer_processes = 4
dml_locks = 5000
global_names = false
log_buffer = 4194304
log_checkpoints_to_alert = true
max_dump_file_size = unlimited
nls_date_format = YYYY-MM-DD
open_cursors = 600
optimizer_mode = CHOOSE
parallel_adaptive_multi_user = false
parallel_max_servers = 64
parallel_min_servers = 32
pga_aggregate_target = 4G
processes = 800
<table>
<thead>
<tr>
<th>Parameter</th>
<th>Value</th>
</tr>
</thead>
<tbody>
<tr>
<td>query_rewrite_enabled</td>
<td>true</td>
</tr>
<tr>
<td>recovery_parallelism</td>
<td>8</td>
</tr>
<tr>
<td>replication_dependency_tracking</td>
<td>false</td>
</tr>
<tr>
<td>sga_max_size</td>
<td>2G</td>
</tr>
<tr>
<td>sga_target</td>
<td>2G</td>
</tr>
<tr>
<td>statistics_level</td>
<td>typical</td>
</tr>
<tr>
<td>transactions</td>
<td>100</td>
</tr>
<tr>
<td>undo_management</td>
<td>auto</td>
</tr>
<tr>
<td>undo_retention</td>
<td>400000</td>
</tr>
<tr>
<td>recovery_parallelism</td>
<td>1</td>
</tr>
<tr>
<td>disk_asynch_io</td>
<td>true</td>
</tr>
<tr>
<td>filesystemio_options</td>
<td>SetAll</td>
</tr>
</tbody>
</table>