ORACLE

Rのエンタープライズ対応

Rを使用してエンタープライズレベルのパフォーマンス、
スケーラビリティ、容易な本番デプロイメント、セキュリティを実現

2020年7月10日 | バージョン2.0

本書の目的

本書では、Oracle Machine Learning for R 1.5.1およびOracle Machine Learning for Spark 2.8.2リリースの機能の概要と強化された点を説明しています。本書は、Oracle Machine Learning for R 1.5.1およびOracle Machine Learning for Spark 2.8.2のビジネス上の利点の評価、およびデータ・サイエンス、機械学習、情報テクノロジーの各プロジェクトの計画立案を支援することのみを目的としています。

対象読者

このホワイト・ペーパーは、最高データ・サイエンティスト、データ・サイエンティスト、Rユーザー、およびOracle DatabaseのパワーをR言語とRエコシステムと組み合わせて活用することに関心がある情報テクノロジーの専門家の方々に役立つ内容となっています。

免責事項

本文書には、ソフトウェアや印刷物など、いかなる形式のものも含め、オラクルの独占的な所有物である占有情報が含まれます。この機密文書へのアクセスと使用は、締結および遵守に同意したOracle Software License and Service Agreementの諸条件に従うものとします。本文書と本文書に含まれる情報は、オラクルの事前の書面による同意なしに、公開、複製、再作成、またはオラクルの外部に配布することはできません。本文書は、ライセンス契約の一部ではありません。また、オラクル、オラクルの子会社または関連会社との契約に組み込むことはできません。

本書は情報提供のみを目的としており、記載した製品機能の実装およびアップグレードの計画を支援することのみを意図しています。マテリアルやコード、機能の提供をコミットメント（確約）するものではなく、購買を決定する際の判断材料になさらないでください。本書に記載されている機能の開発、リリース、および時期については、弊社の裁量により決定されます。

製品アーキテクチャの性質上、コードが大幅に不安定化するリスクなしに、本書に記載されているすべての機能を安全に含めることができない場合があります。

目次

はじめに

Rは、統計分析、データ探索とデータ操作、機械学習、可視化のためのソフトウェア機能の統合スイートであり、専有の統計環境に代わる選択肢として1994年に開発されました。このオープン・ソースのスクリプト言語は、データ・サイエンティスト、ビジネス・アナリスト、データ・アナリスト、および統計家にとって、重要な分析手段の1つとなっています。数百万に上る世界中のRユーザーが、数千ものオープン・ソースRパッケージを利用しており、製薬、バイオインフォマティクス、空間統計、金融市場分析、線形/非線形モデリングなどの幅広い分野で、Rエコシステムによって生産性を向上しています。

データ・サイエンティストは通常はパソコンやワークステーション上でRプログラムを実行しますが、大量のデータに対して高度な演算処理を素早く実行する必要性が増しています。オラクルは、Rを大規模なデータに使用できるようにするために、Oracle DatabaseやOracle Big Data環境に格納されたデータに対して、機械学習、統計分析、グラフィカル分析を実行できる幅広い機能を開発しました。これらのエンタープライズレベルの機能のおかげで、高いレベルのセキュリティ、スケーラビリティ、パフォーマンスが必要なプロジェクトが可能になり、オンプレミスまたはOracle Cloudの本番環境にRスクリプトを迅速かつ容易にデプロイできるようになりました。

Oracle Machine Learningは、Rの利点と、Oracle DatabaseとData Lake環境が持つパワーとスケーラビリティを併せ持っています。Oracle Machine Learning for R（OML4R）は、オンプレミスとOracle Cloudの両方のOracle Databaseに含まれています。Oracle Machine Learning for Spark（OML4Spark）は、オンプレミスのBig Dataソリューション向けのOracle Big Data Connectorsソフトウェア・スイートのコンポーネントであり、Oracle Big Data Serviceに含まれています。RプログラムとRパッケージをこれらの製品とともに使用すると、セキュアな環境で大量のデータを処理できます。顧客は機械学習モデルを構築し、ローカル・データ・ストアに対してそのモデルを実行したり、Oracle Databaseに格納されたデータに対してRコマンドやRスクリプトを実行したりすることができます。OML4Rを使用すると、Rユーザーが定義した関数をSQLから起動し、構造化された結果やイメージを返してアプリケーションやダッシュボードに即座にデプロイできます。

これらの機能は、今日の多くのビッグ・データ・プロジェクトでは特に重要です。データ・サイエンティストは、制御下でOracle Database内のデータにアクセスできるため、ITセキュリティ・ポリシーを徹底しながら、生産性を大幅に向上させることができます。オラクルの統合型アプローチでは、データ分析が簡素化され、データ移動が最小限に抑えられるか排除され、生データをすぐに使用可能な情報に変換する時間が短縮されます。さらに、オンプレミスのBig Data環境のオプション・コンポーネントであるOracle Big Data SQLにより、Rユーザーは、データベース表としてRから直接マッピングされたビッグ・データ・ソースを操作できるため、OML4Rで使用できるData Lakeソースの範囲が広がります。Oracle Big Data Serviceの一部であるOracle Cloud SQLを使用すれば、ユーザーはこれをOracle Cloudで行うことができます。

Oracle Machine Learningには、SQLユーザーとRユーザーの両者をサポートする2つのコンポーネントがあります。Oracle Machine Learning for Rを介して、Rユーザーはデータを移動することなく、標準的なR構文を使用してOracle Database内のデータを透過的に操作できるため、Oracle Databaseを高パフォーマンスの計算エンジンとして活用できます。ユーザーはR関数呼出しをSQLに変換することによって、インデータベース統計手法を利用してスケーラビリティとパフォーマンスを向上させることができます。OML4Rでは、十分に統合されたR APIが、強力なインデータベース機械学習アルゴリズムに提供されます。
ユーザー定義のR関数を、Oracle Databaseが起動して制御するRエンジンで実行することもできます。このユーザー定義R関数は、RまたはSQLのいずれからも呼び出すことができ、Comprehensive R Archive Network（CRAN）とその他のRパッケージを利用できます。Oracle Machine Learningのユーザーは、ビジネス・アナリストからデータ・サイエンティストに至るまでのユーザーを対象とする、オラクル製およびサードパーティ製のさまざまなR GUIやIDEオプションを使用できます。

Oracle Big Data Connectorsの1つであるOracle Machine Learning for Spark（旧名：Oracle R Advanced Analytics for Hadoop（ORAAH））により、Rユーザーは標準的なR構文を使用して、Apache HiveやApache Impalaのデータを透過的に操作でき、SparkSQLおよびSpark Dataframe関数を使用して、Apache Sparkのデータを操作できます。OML4Sparkでは、統合されたRインタフェースを介して公開されている多数のSpark MLlibアルゴリズムなど、Sparkベースの並列機械学習アルゴリズムも豊富に提供されます。さらにユーザーは、RからカスタムのMapReduceジョブを実行できます。マッパー関数とリデューサ関数をRで記述して、それらの関数でCRANパッケージを利用できます。

OML4RおよびOML4Sparkは、Oracle R Distributionと呼ばれるオラクルのオープン・ソースRの再配布パッケージ、およびオープン・ソースRのディストリビューションと組み合わせて使用できます。Oracle R Distributionには、高度な線形代数や行列処理に対応したIntelのMKLなどの高パフォーマンス・ライブラリを自動的に取り込むことができます。Oracle R Distributionはオラクルによってサポートされています。

このホワイト・ペーパーでは、開発者が以下を行うことができるようにすることで、オラクルがオープン・ソースRをいかに拡張しているかについて説明します。

- Oracle DatabaseまたはSpark/Hadoopのデータを透過的に分析および操作する
- データとタスクのパラレル処理によりOracle DatabaseでRスクリプトを実行する
- Rを通してインデータベースSQLベースのアルゴリズムをシームレスに使用する
- データベースでRモデルをスコアリングする
- SQLからRスクリプトを容易に実行する
- RをITソフトウェア・スタックに統合する

RをOracle DatabaseおよびBig Dataと統合することで、双方の利点、すなわち使い慣れた強力な統計環境と、スケーラビリティ、パフォーマンス、セキュリティの大幅な向上を実現できます。

概要

Oracle Databaseを利用したRコードの実行

データ・サイエンティストとその他のRユーザーは、SQLの知識がなくても、お好きなR IDEからRスクリプトを実行して、データの探索や準備を行い、データベースのデータに幅広い機械学習機能を実行できます。

Rの記述とインデータベース実行

オラクルは、Oracle Database内でR演算を実行できるRパッケージ一式を開発しました。この透過レイヤーにより、オラクルの表およびビューはネイティブのRオブジェクトであるかのように、R の*data.frame*のサブクラスである*ore.frames*を介して、R環境からアクセスできます。これにより、ユーザーは幅広いR機能を実行できます。

Rユーザーは、一般的なオープン・ソースR dplyrパッケージからオーバーロードされた機能を提供するパッケージOREdplyrを利用することもできます。これらの機能を使用することで、ユーザーはデータ・アクセス、スケーラビリティ、パフォーマンスなどの課題ではなく、データ分析に集中することができます。

透過レイヤーのおかげで、R開発者は使い慣れた環境、言語、ツールを使用できます。内部では、オーバーロードされたR関数が、データベースのパラレル処理、問合せの最適化、列の索引付けとパーティション化などの利点を生かし、統計機能の豊富なインデータベース・ライブラリを活用しながら、Oracle Database内で実行されます。Rユーザーは、標準的なRの開発スキルやツールを使用して、このような複雑な演算をデータベース内で実行できます。

インデータベース機械学習アルゴリズム

ユーザーは、予測分析手法を構築、評価、共有、デプロイしながら、パフォーマンスに優れたオラクルの並列分散機械学習アルゴリズムも活用できます。これらのインデータベース・アルゴリズムが表およびビューからテキスト列をアクセプトすることで、テキスト・マイニングが統合され、用語とテーマの抽出が自動化されます。抽出されたデータは、その後、モデルの構築やデータのスコアリングにおいて他の予測変数と結合されます。

データ・サイエンティストの生産性をさらに向上させるために、パーティション化モデルと呼ばれるモデル一式を自動的に作成できます。このパーティション化モデルでは、各コンポーネント・モデルがユーザー指定のデータ・パーティションに構築されます。単一の統合されたモデルを使用して、スコアリングを有効化して簡素化します。

データ・サイエンティスト、アナリスト、およびアプリケーション開発者は、データのある場所にアルゴリズムを持ち込むことで、データ量の増加に合わせて機械学習プロジェクトを容易に拡張できます。たとえば、ディシジョン・ツリー、サポート・ベクター・マシン、k-means、ニューラル・ネットワーク、スケーラブルな機械学習のためのランダム・フォレストといったネイティブな並列分散インデータベース・アルゴリズムを使用できます。オラクルの強力なエンジニアド・システムであるOracle Exadata上で実行すると、この処理はストレージ層で行われるため、さらに高速なデータ分析と予測が可能です。そのため組織は、オラクルのエンジニアド・システムがもたらす優れたパフォーマンスの利点をいっそう享受でき

ます。オラクルのアプローチには、次のような明白な利点があります。

- R内でデータの準備、分析、可視化を単独で行うことが可能
- 問合せの最適化、列の索引付け、パラレル処理、インメモリ実行とパーティション化のオプション機能を備えた高パフォーマンスの計算エンジンとしてデータベースを利用可能
- フラット・ファイル・データの管理を排除し、ストレージ、バックアップ、リカバリ、セキュリティに付随する複雑性を解消
- Rのメモリ制約を最小限に抑えることで、ビッグ・データ要件に対応
- RスクリプトをSQLから実行して、エンタープライズ・アプリケーションとダッシュボードの容易なデプロイメントと統合を実現

CRANパッケージの利用

オラクルの*組込みR実行*機能を使用すれば、データ・サイエンティストは、Comprehensive R Archive Networkリポジトリにある何千もの専門アルゴリズムを活用できます。自身でアルゴリズムを作成するか、既存のアルゴリズムをダウンロードしてから、これらのパッケージをデータベース・サーバー側のRエンジンにインストールできます。このアーキテクチャにより、データベースに対するデータのセキュアな送受信、および選択したアルゴリズムへのデータの直接フィードが容易になります。

データとタスクをパラレル処理する組込みインフラストラクチャを使用して、ユーザー定義関数のパラレル実行と関連のデータ・フィードを活用できます。たとえば、顧客表を郵便番号で分割し、複数のRエンジンをパラレル実行することで、R環境から離れることなく、各郵便番号の顧客グループを同時に処理できます。透過レイヤーからアクセス可能なものから、CRANパッケージからアクセス可能なものまで、多様な統計技術を使用するRスクリプトは、オラクルのインデータベースRスクリプト・リポジトリ内に構築して保存し、RまたはSQLから実行できます。

- Rで独自のパッケージを作成、またはCRANオープン・ソース・パッケージを利用
- Oracle Databaseの管理下で起動したRエンジンによって、ユーザー定義関数をデータベース・サーバー・マシンで実行
- Oracle Databaseのスケジューリング機能を使用して、SQLインタフェース経由でRスクリプトの"完全自動"実行を実現
- データとタスクをパラレル処理するユーザー定義R関数を実行して、大規模ジョブを高速化
- 結果をアプリケーション、ダッシュボード、レポートと統合

本番環境へのR分析のデプロイ

Oracle Machine Learningを利用すると、R開発者はOracle Databaseを使用して、SQL問合せ内でRスクリプトを実行できます。そのため、どのような分析アプリケーションの本番環境でも、容易にRスクリプトを操作できるようになります。Oracle DatabaseのSQL問合せに、データベースRスクリプト・リポジトリに登録されているユーザー定義R関数の呼出しを含めることができます。ユーザーはスクリプト名を使用して、そのスクリプトを呼び出す問合せを開始し、構造化されたイメージ結果を表で、または結合された結果をXMLで受け取ることができます。

ある通信プロバイダは、OML4Rを使用して複雑なアンケート調査を強化しました。この企業のアナリストは、ユーザー定義R関数をOracle Database内に保持し、データをフィルタリングして、パラメータ化した分析ダッシュボードを介して結果を表示しています。データベースとダッシュボード・インフラストラクチャはどちらも、このアーキテクチャの標準コンポーネントですが、SQLを基盤にRスクリプトと結合することで、さらに強化されました。これらの機能によって、Rは、高度な統計モデルをデータベースのデータに直接実行できる強力な言語となります。

HadoopとSparkを使用したRの実行

Oracle Machine Learning for Spark（旧名：Oracle R Advanced Analytics for Hadoop）は、Rパッケージ・フロントエンドを備えたコンポーネントであり、ビッグ・データ・クラスタのデータに対してクラス最高のSparkベースの機械学習アルゴリズムを提供するとともに、HDFS、Apache Hive、Apache Impala、Spark DataFramesに格納されたデータとHadoopへの透過的なアクセスを実現します。OML4Sparkを使用すると、Rモデルを構築し、大量のデータに対して効率的にスコアリングできるほか、R環境を離れることなくSparkインメモリ処理を利用できます。ユーザーはRにより、オラクルが提供する機械

学習アルゴリズム、厳選された多数のSpark MLlibアルゴリズム、およびCRAN Rパッケージを利用して、Data Lakeに格納されたデータを分析できます。

機械学習に関しては、OML4Sparkでは、実行時にビッグ・データ・クラスタとSparkの恩恵を受ける複数の並列分散アルゴリズムが提供されます。線形モデル、一般化線形モデル、多層パーセプトロン・ニューラル・ネットワークをはじめとするOML4Sparkのカスタム・アルゴリズムは、すべてのデータを一度にメモリ内に収める必要がないため、Sparkではより適切に拡張されます。また、類似のオープン・ソースSpark MLlib関数よりも高速に実行されます。OML4SparkではMLlibに対する拡張インタフェースが提供されますが、この拡張インタフェースはすべてのR計算式仕様を生かしており、SparkRが提供するインタフェースよりも優れています。

Sparkで実行されるアルゴリズムでは、どちらのモデル・ビルドもサポートされ、HDFS、Apache Hive、Apache Impala、Spark DataFrames、およびJDBCデータソースの形式の入力データセットを使用して（予測スコアリングが）適用されます。モデル自体は、別のクラスタで実行するために、HDFSおよびローカル・ファイル・システムにバイナリ形式で保存できます。

OML4Sparkは、OML4Rがサポートする同じ透過レイヤー機能を利用することで、Apache HiveやApache Impalaの表からアクセスできるデータ上でRコマンドを実行できるようにします。この透過レイヤーのおかげで、R開発者は使い慣れたR環境とRコマンドを使用でき、その内部では、関数が自動的にHQL（Hive Query Language）またはCloudera Imapala SQLに変換され、Hadoopクラスタでパラレル実行されます。

さらに、ユーザーはRから直接Spark Dataframesのデータを操作できるため、*join*、*append*、*aggregate*統計関数などの基本的なデータ変換関数をすべて利用できます。また、新しい列を作成したり、必要に応じて、Spark DataFramesのインメモリで直接SparkSQL問合せを実行したり、HIVEに登録されている表に対してSparkSQLを使用したりすることができます。

MapReduceを活用するRプログラムは、Hadoopクラスタにデプロイできるほか、Hadoopクラスタのデータ並列性がもたらすパフォーマンス上の利点を享受できます。ユーザーは、Hadoop内部の知識や、MapReduceコマンドライン・インタフェースやITインフラストラクチャの知識がなくても、これらのRスクリプトを作成、実行、保存できます。

Oracle Databaseを使用したビッグ・データのIoTユースケース

モノのインターネット（IoT）は、機械学習を適用する新たなビジネス機会を創出します。航空機、電車、車、半導体製造機械、大型ハドロン衝突型加速器、そして私たちの家など、あらゆる場所でセンサーがデータを収集しています。そのようなセンサーの一例として、家庭の電力使用量を定期的に、おそらく15分おき程度にレポートできるスマート・メーターが挙げられます。電力会社は、このデータを使用して、各顧客の電力使用パターンをモデル化できるだけでなく、個々の顧客の使用量を予測できます。すべての顧客に対して、電力会社は需要の総計を通常は数日または数週間単位で算出して予測できるため、職員の配置、電力の転送や購入などをより効率的に行うことができます。

顧客ごとに1つの予測モデルを構築することは、数百万もの顧客を抱えている可能性のある電力会社にとって、興味深い挑戦です。ある電力会社が100万の顧客を抱えているとします。スマート・メーターは、1年間に350億以上の読取り値を収集する可能性があります。ただし、それぞれの顧客が生成する読取り値は35,000ほどにすぎません。Rは、ほとんどのハードウェアで、35,000の読取り値を基に1つの予測モデルを容易に構築できます。ここで、各モデルの構築に必要な時間がわずか10秒であっても、これを順次処理すると、すべてのモデルを構築するにはおよそ116日かかることに注意してください。数日または数週間ごとに結果が必要なため、数か月の遅れは、このプロジェクトが役に立たないことを意味しています。オンプレミスまたはOracle CloudのOracle Exadataのような強力なハードウェアを利用して、これらのモデルをパラレル処理で構築すれば、たとえば並列度が128の場合、すべてのモデルを1日未満で計算できます。

ユーザーは、さまざまなRパッケージで提供されるパラレル処理を利用できますが、その際に考慮すべき要素がいくつかあります。たとえば、特定のモデルで障害が発生したらどうしますか。モデルは、顧客ごとに100万の個別のフラット・ファイルとして保存しますか。フラット・ファイルの場合、バックアップ、リカバリ、セキュリティにはどのように対処しますか。これらのモデルをどのように使用して顧客の電力使用量を予測しますか。また、予測データはどこに保存しますか。通常はアプリケーションやダッシュボードがSQLと連携する本番環境に、これらのRモデルをどのように組み込むことができますか。

OML4Rの組込みR実行機能を使用すれば、データ・サイエンティストは、ユーザー定義R関数で顧客ごとに1つのモデルを構築するというタスクに集中できます。この関数は、Oracle DatabaseのRスクリプト・リポジトリに保存されます。OML4Rでは、*ore.groupApply*といった組込みR実行関数の1つを使用して、この関数を起動できます。Oracle Databaseの組込みR関数は、複数のRエンジンを起動し、1つのデータ・パーティションを指定のデータベース表からデータ・サイエンティストが作成した関数にロードし、結果モデルを再びOracle Database内のRデータ・ストアに瞬時に保存する一連の処理を管理します。これにより、モデルを構築して保存するプロセスが大幅に簡素化されます。さらに、すでに実装されている標準的なデータベース・バックアップとリカバリの仕組みを使用すれば、別の専門プラクティスを考案する必要がなくなります。
これらのモデルを使用した電力使用量の予測は、類似の方法で処理されます。

このようなユーザー定義R関数を本番環境で使用する場合、モデルの構築と予測のどちらにおいても、ユーザーはデータ・サイエンティストが作成したR関数と同じものをSQLから呼び出すことができます。予測データは、アプリケーションやダッシュボード、他のSQL問合せで使用できるデータベース表として直ちに利用できます。さらに、R関数を呼び出すSQL文は、Oracle DatabaseのDBMS_SCHEDULERパッケージを使用して、定期的に実行されるようにスケジューリングできます。

データ・サイエンティスト、データ・アナリスト、アプリケーション開発者、および管理者は、Oracle Machine Learningの組込み機能を利用すれば、通常は新しいプロジェクトごとに作成される複雑なコードやテスト戦略を再作成する必要はありません。
代わりに、オラクルによるRとOracle Databaseの統合を活用して、アプリケーションやダッシュボードとともに使用するRベースのソリューションを容易に設計および実装し、エンタープライズ規模に拡張できます。

Oracle DatabaseとHadoopを使用したビッグ・データのユースケース

オラクルのビッグ・データ・テクノロジーは、Big Data環境、R、およびOracle Database間で容易にデータを移動できるように設計されています。データ・サイエンティストはJavaやScalaに頼ることなく、OracleやHadoopクラスタに格納されたデータにアクセスし、MapReduceプロセスや、Apache HiveまたはSparkの問合せをコーディングして、機械学習アルゴリズムをRで実行できます。以下で説明するように、この柔軟なアーキテクチャを利用することで、組織は大規模な表やデータセットを容易に分析できます。急を要する今日のビッグ・データ課題を解決する上で、RはSQLに続き、エンタープライズ分析における素晴らしい選択肢となりました。

ユースケース1：信用リスクの分析

銀行は絶え間なく新しいサービスを顧客に提供していますが、サービスの提供条件は顧客の信用状態によって異なります。支払っている金額は、貸付残高のうちの期限切れとなった最低金額か、それ以上か。支払いが遅れたことがあるか。与信限度額のどの程度を使用しているか。その他に与信枠をどのくらい持っているか。総合的な負債/収入比はどのくらいか。

これらの変数はすべて、各顧客に設定する与信額や顧客への提示条件に関するポリシーに影響を与えます。EquifaxやTransunionなどの信用情報機関は、個人の総合的な信用履歴を調査しますが、銀行は自行の顧客について、個々の取引レベルに至るまでのはるかに詳細な履歴一式を調査できます。ビッグ・データ分析においても、このレベルの精度でそのような大量のデータを分析する必要があります。

たとえば、ブラジルのあるオラクル顧客は、数億のレコードに対して複数のニューラル・ネットワーク・アルゴリズムを実行して、各顧客に関する数千もの属性を調査しています。以前は、この銀行は大量のデータを高速処理して意味のある統計情報を生成することに苦心していました。そこで同行は、OML4Sparkを使用して専門アルゴリズムを実行し、本番環境のHadoopファイル・システムや、Apache Hive、その他のツールを実行しているクラスタと同じクラスタ上で、このデータをパラレル処理により分析することで、この問題を解決しました。OML4Sparkを使用すると、データ・サイエンティストは、銀行が持つ大規模なHadoopファイル・システムに格納された表で、Rの分析、統計、モデルを実行できます。現在は、これらのファイル・システムとApache Hive表に対して、複雑な統計アルゴリズムを実行できるようになりました。

このアルゴリズムでは、R計算式オブジェクトを使用したモデル構築など、標準的なRのアプローチが使用されます。内部では、OML4Sparkが提供するインタフェースにより、銀行のクラスタ全体で複数のプロセッサを使用して、Sparkベースの実装またはMapReduceジョブがパラレル実行されます。データ・サイエンティストは、これらのMapReduceプロセスとSparkアルゴリズムをRで作成して、Hadoopに保存できます。また、Javaを使用することなく、これらのモデルによってレビュー、グラフ作成、分析を容易に行い、結果をOracle Databaseに送信できます。

ユースケース2：不正の検出

もう1つの一般的なユースケースとして、金融取引の分析による不正検出が挙げられます。銀行、小売業者、クレジットカード会社、電気通信事業者、および他の多くの大規模組織が、この問題に取り組んでいます。データをスコアリングして潜在的な不正を検出する場合は一般的に、顧客の口座内で発生した取引について調査します（スコアリングとは、機械学習モデルを使用した結果予測です）。

顧客の通常の行動を把握したら、通常とは異なるパターンや疑わしい取引を識別できるようになります。たとえば、通常はロサンゼルスで買い物をしている顧客が、突然ローマで一連の取引を行った場合、不正である確率が非常に高くなります。ただし、頻繁に旅行する顧客の場合、ローマでのアクティビティの急増は異常パターンと通常パターンのどちらでしょうか。過去の取引すべてを捕捉してそのパターンを調査すれば、通常の行動を反映するモデルを構築できます。

Rは、このような取引を分析できる予測モデルの作成に最適なアルゴリズムと環境を備えていますが、CRANパッケージのアルゴリズムは、通常はマルチスレッドではありません。そのため、これを実行するマシンのメモリと単一CPUの処理能力による制限を受けます。Rは通常、特別なパッケージやプログラミングがない場合、マルチプロセッサ・ラップトップでさえも、CPU能力を活用しません。

Oracle Machine Learningは、R言語を使用してスクリプトを定義し、Rスクリプト・リポジトリに保存し、データベースで実行することで、顧客の購入パターンの分析に伴う大量の演算処理要求に対処できます。

組織は、Oracle ExadataやOracle Big Data Applianceなどのエンジニアド・システムを活用してこの取組みをオンプレミスで拡大することも、Oracle Cloud Infrastructureのスケーラビリティに依存することもでき、Oracle Machine Learningの結果を、オンプレミスのOracle Analytics ServerやOracle Data Visualization Desktop、Oracle Analytics Cloudなどのアプリケーションと統合して、結果を表示できます。R開発者はRを使用することで、Hadoopに構築された不正モデルの結果を利用できます。また、そのモデルをOracle Databaseにデプロイすると、モデルは取引レベルで迅速に行動を予測でき、エンタープライズ・アプリケーションの一部となってリアルタイムの予測を実現します。

不正を予防する上で、リアルタイムの分析は重要です。8時間（ラップトップにデータをステージングして、前日の行動の詳細分析を実行するためにかかるであろう時間）より前に発生した不正な取引を特定することも重要ですが、この取引をモデルに対してリアルタイムでスコアリングして、取引を阻止できる、またはさらなる調査のためにフラグ付けできることの方が、はるかに有益であることは明らかです。

ユースケース3：顧客離れの防止

多くの事業にとって、とりわけ電気通信などの競争の激しい市場では、顧客離れは重要な問題です。たとえば、携帯電話のユーザーは、受信状態に問題があったり、通話が頻繁に途切れたりする場合、別のサービス・プロバイダを探すことを検討するかもしれません。現在のサービス・プロバイダは、常にユーザーの行動を分析して、他社に乗り換える可能性を予測しています。サービス・プロバイダには、「問題と行動が類似する顧客の90 %が、別のサービス・プロバイダに乗り換えた」ことを示す統計モデルがあります。このモデルをユーザーのデータに適用してスコアリングすることで、乗換えの可能性を明らかにすることができます。ユーザーのスコアリング・データにより、ユーザーは行動が類似する他の数百万の顧客と関連付けられます。

Oracle Machine Learningを使用すれば、顧客がWebページを閲覧したり、モバイル・アプリを使用したりする際に、これらのRモデルを実行して、現在の行動、および業務系データ・ストアやデータウェアハウスに対するリアルタイム分析に基づき、その場で提案を行うことができます。

スコアリングはバッチ処理によりオフラインでも実行できます。たとえば、特別なオファーや広告キャンペーンの対象とする顧客を判断するために、1億人の顧客のうちの誰が、多数のオファーのそれぞれに反応するかを予測する必要があるとします。十分な処理能力と適切な予測モデルがあれば、データ・サイエンティストは解約率だけでなく、解約の背後にある理由も見抜くことができます。ある電気通信会社は、OML4Sparkを使用して、支払い記録や通話プラン、サービス履歴を調査し、顧客データベース内の類似性と傾向を発見することで、より多くの情報に基づいた意味のある決定を下していました。OML4Sparkのおかげで、大規模なBig Dataクラスタでバッチ・ジョブをパラレル実行できました。

結論

ほとんどの組織は、エンタープライズレベルの厳格な制御を使用して情報をセキュアに保存するために、データベースに依存しています。オラクルは、Rを大規模な機械学習、分析ツール、ビッグ・データへの取組みとともに使用できるようにしました。開発者は、オラクルの卓越したスケーラビリティとパフォーマンスを利用して企業のビッグ・データ課題を解決する際に、使い慣れたR環境をOracle DatabaseやBig Dataクラスタ、分析ツールとともに使用できます。ファイル抽出に慣れたデータ・サイエンティストとアナリストは、データベース中心のアーキテクチャを取り入れることで、デスクトップのR実装からデータベースにデータを送信し、Oracle Database内のそのデータを処理できます。

RとSQL間の透過性のおかげで、データ・サイエンティストはOracleデータベース、Apache Hive、またはApache Impala内のデータにRを直接使用できるため、効率性が向上します。Rユーザーは、インデータベースSQL分析機能、機械学習機能、およびオープン・ソースのRパッケージをOracle Databaseと組み合わせて使用することで、データとタスクのパラレル実行を実現できます。

Oracle Machine Learningは、R環境を離れずに使用できます。CRANパッケージや作成済みの他のR資産を利用して、SQL文からユーザー定義R関数を呼び出すことで、Rベースの分析機能を本番アプリケーションや本番ダッシュボードにデプロイできます。ビッグ・データの問題では、OML4Sparkが持つスケーラビリティを、複数のBig Dataクラスタとともに活用できます。

Oracle Database、Oracle Big Data Connectors、またはOracle Linuxのライセンスを所有するお客様、およびOracle Database Cloud ServiceまたはOracle Big Data Serviceを実行中のお客様は、Oracle R Distributionに対するエンタープライズ・クラスのサポートを受けることができます。

要約すると、Oracle DatabaseやBig Dataとともに使用できるようにRを拡張すると、これまでとは逆に、分析機能をデータ側に持ち込むことができます。Oracle Machine Learningを使用してOracle DatabaseにR機能を送信し、OML4Sparkを使用してBig Dataクラスタ・ノード上でRからSparkジョブを呼び出すことで、データ・サイエンティストはデータ移動を最小限に抑え、生データをすぐに使用可能な情報に変換する際の待機時間を短縮できます。Oracle DatabaseとBig Data環境を統合すれば、コスト効率に優れた強力なビッグ・データ分析ソリューションが実現します。

参考資料

Oracle Machine Learning

https://oracle.com/machine-learning

Oracle Machine Learning for R

https://oracle.com/goto/R

オラクルの情報を発信しています

+1.800.ORACLE1までご連絡いただくか、oracle.comをご覧ください。

北米以外の地域では、oracle.com/contactで最寄りの営業所をご確認いただけます。

blogs.oracle.com facebook.com/oracle twitter.com/oracle

Rのエンタープライズ対応

2020年7月