Datenanalyse mit Oracle R Enterprise for Beginners

Dr. Nadine Schöne
Oracle Deutschland B.V. & Co. KG
Potsdam

Schlüsselworte

Oracle R Enterprise, Datenanalysen, Data Analysis, Reporting, Data Mining, R Programming Language, Statistik, Statistics, Graphics, Graphische Darstellung

Einleitung

Der Startpunkt einer Datenanalyse ist immer der Datensatz – und das Ziel sind Graphiken und Statistiken. Aber wie kommt man zum Ziel? Wir starten mit einem Beispieldatensatz, der in der Oracle Datenbank liegt. Zuerst verschaffen wir uns einen Überblick über dessen Struktur und Inhalt. Danach stellt sich die Frage nach der Datenqualität - inwiefern „lohnt“ sich eine Analyse? Wir beleuchten auch die Möglichkeiten und Fallstricke graphischer Darstellungen. Abschließend befassen wir uns mit statistischen Tests.

Datenerfassung und Datenverarbeitung

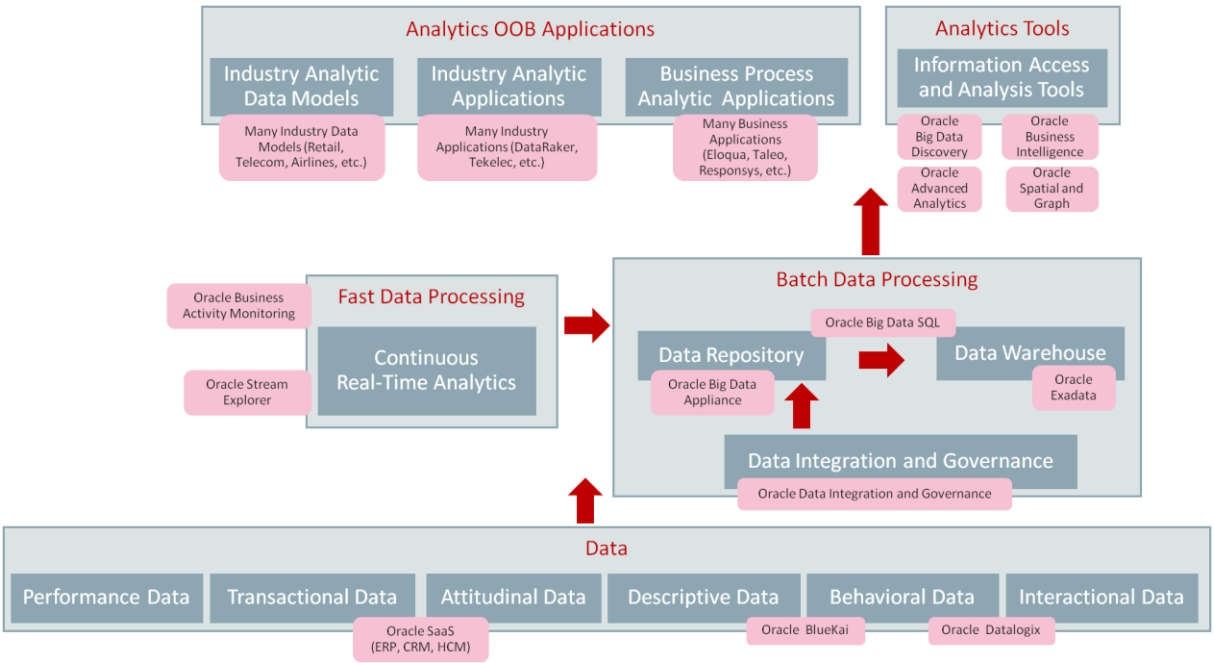

Abb. 1: Es gibt eine Vielzahl an Datenquellen und mindestens genauso viele Möglichkeiten diese zu erfassen, zu verarbeiten, zu speichern und zu analysieren. Die Datenbankoption Oracle Advanced Analytics (rechts oben unter „Analytics Tools“) bietet einen einfachen Einstieg in den Bereich der Datenanalysen.

Es gibt eine Vielzahl an Datenquellen und mindestens genauso viele Möglichkeiten diese zu erfassen, zu verarbeiten, zu speichern und zu analysieren. Abb. 1 liefert einen Ausschnitt der Möglichkeiten, die Oracle zur Erfassung und Verarbeitung von Daten liefert. Das Portfolio wird ständig erweitert, so sind z.B. weitere Oracle Cloud Services angekündigt.

Einen einfachen Einstieg in die Datenanalysen bietet die Datenbankoption Oracle Advanced Analytics (rechts oben in Abb. 1 unter „Analytics Tools“ zu finden). Diese beinhaltet neben dem Oracle Data Miner auch Oracle R Enterprise (ORE). Wir werden uns auf ORE fokussieren.

Oracle R Enterprise (ORE) und Open Source R

ORE ist eine von Oracle entwickelte Variante der Open-Source-Programmiersprache R. Open-Source R wurde von Statistikern speziell für die Datenanalyse entwickelt und wird kontinuierlich weiterentwickelt. R ist eine statistische Workbench, ein Data-Science-Ökosystem, und DIE lingua franca für Data Science.

Open-Source R wird in der Regel auf einem Client (PC oder Notebook) installiert. Daten müssen zur Verarbeitung mittels Flat File Export erst von der Datenbank in den R-Workspace importiert werden (siehe Abb. 2). Bei größeren Datenmengen vermindert dies die Analysegeschwindigkeit erheblich. Weiterhin ist die Parallelisierung von Berechnungen mit R nur händisch möglich, automatische Parallelisierung ist nicht vorgesehen.

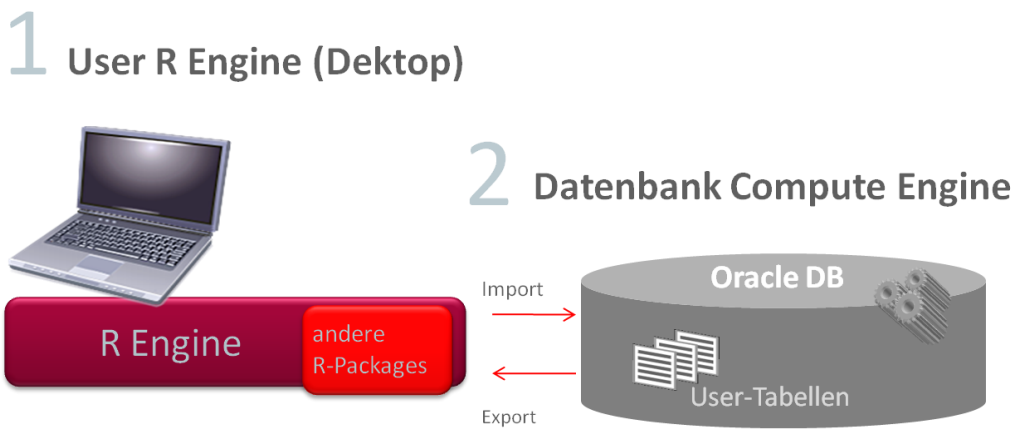

Abb. 2: Aspekte herkömmlicher R/Datenbank-Interaktionen

Im Gegensatz zu R nutzt ORE nicht nur eine Client-Engine, sondern nutzt durch eine zweite R-Engine (u.U. sogar mehrere) auf dem Datenbankserver auch die Rechenkraft der Datenbank („Collaborative Execution“-Modell, siehe Abb. 3). Somit entfällt mit ORE der Datenexport, wodurch ORE ungleich performanter als R ist. Außerdem ist eine Parallelisierung der Anfragen möglich, was die Performance weiter steigern kann.
Die Funktionalität von R wird permanent durch die Nutzer selbst erweitert: Gekapselte Funktionalität kann in Form von R-Packages für alle Nutzer auf einem zentralen Server (CRAN) zum Download

bereitgestellt werden. Mit ORE lässt sich sämtlicher nativer R Code verwenden. Auch alle über CRAN downloadbaren R-Pakete sind nutzbar.

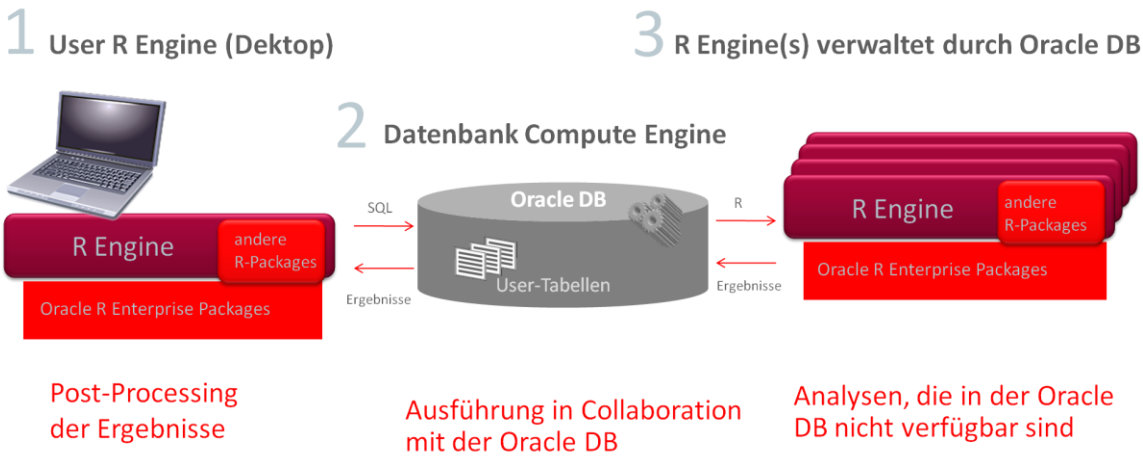

Abb. 3: "Collaborative Execution"-Modell mit ORE

Verbindung mit der Datenbank

Mit dem Befehl `ore.connect` erfolgt die Verbindung mit der Oracle Datenbank. Somit sind die Daten im Datenbankschema verfügbar. Außerdem können Processing Power, Memory und Speicherkapazität des Datenbankservers nun über das ORE Interface genutzt werden.

```
ore.connect(user="moviedemo",sid="orcl",host="localhost",password="welcome1
")
```

Neben User ID `user`, System Identifier `sid`, Host und Password können noch weitere Parameter übergeben werden. Das Argument `type` legt fest, ob eine Verbindung zu einer Oracle Datenbank oder zu Hive Tabellen in einem Hadoop Cluster hergestellt werden soll (default ist `"ORACLE"`). Das boolsche Argument `all` spezifiziert, ob ORE automatisch für jede Tabelle im Datenbankschema ein `ore.frame` Objekt mit Metadaten erzeugen soll (default ist `"FALSE"`).

Um `ore.frame` Objekte mit Metadaten zu erzeugen muss ore.sync aufgerufen werden. Der Aufruf von ore.attach schreibtt die Namen der `ore.frames` in den Serach Path auf.

```
ore.sync(schema="MOVIEDEMO",table="MOVIEAPP_LOG_AVRO")

ore.attach()
```

Die ORE Client Engine wandelt in ihrem Transparency Layer Anfragen an die Datenbank in SQL um, so dass die explizite Übergabe von SQL-Befehlen entfällt. Um mit ORE Datenanalysen zu implementieren braucht man kein SQL; Kenntnisse in R reichen aus.

Oracle Labs und FastR

Die Open-Source-R-Implementierung hat die Eigenschaft, dass sie R-Code Schritt für Schritt ausführt („interpretiert“), was hinsichtlich der Ausführungsgeschwindigkeit nicht optimal ist. R-Entwickler wenden aus diesem Grund häufig die Strategie an, die Performance-kritischen Teile ihrer Anwendungen in C, C++ oder Fortran neu zu schreiben und von R aus aufzurufen.

FastR ist eine Neuimplementierung von R in Java und baut auf Truffle und Graal auf. Anders als Open-Source-R übersetzt FastR häufig ausgeführte Teile des R-Codes in Maschinencode für die Hardware, so dass der Aufwand der Interpretierung wegfällt. Truffle, Graal und FastR sind Open-Source-Projekte.

Weitere Informationen

Neugierig geworden? Oder wollen Sie Beta Tester für FastR werden? Dann meden Sie sich bei mir.

Kontaktadresse:

Dr. Nadine Schöne
Oracle Deutschland B.V. & Co. KG
Schiffbauergasse 14
D-14467 Potsdam
Telefon: 0331 200 7190
E-Mail: nadine.schoene@oracle.com