Business News

DOAG Zeitschrift für die Anwender von Oracle Business- und BI-Lösungen

BS

Predictive Analytics – der Blick in die Zukunft

Visuelle Analyse
Am Beispiel der Panama Papers
Seite 5

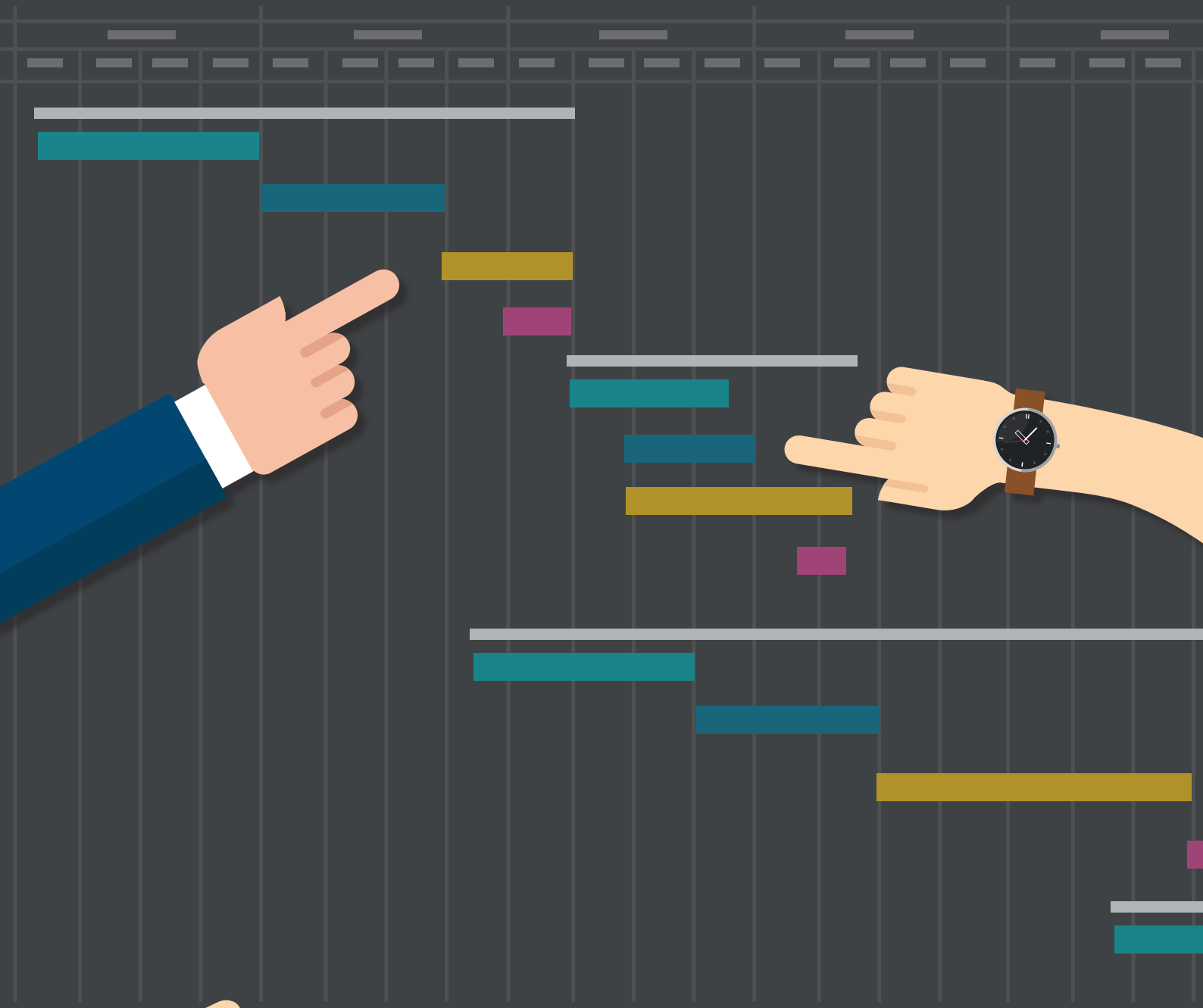
Process Mining
Enormes Potenzial für das Unternehmen
Seite 15

Machine Learning
Analysen und Massendaten
Seite 30

Herbst 2017 | ISSN 09 36-0360 | http://bs.doag.org

11. DOAG Primavera Community Day

14. & 15. März 2018 in Wien



Mehr Informationen und Anmeldung unter:

www.doag.org/go/primaveraday2018

DOAG AOUG AUSTRIAN ORACLE USER GROUP

Rolf Scheuch
DOAG-Vorstand und Leiter
der Data Analytics Community

Liebe Leserinnen und Leser,
endlich wird ein uralter Traum oder vielleicht doch ein Albtraum wahr. Predictive Analytics tritt mit dem Versprechen an, uns einen Blick in die Zukunft zu ermöglichen. Ein System nimmt einem die Arbeit ab, komplexe Zusammenhänge zu verstehen und Prognosen zu entwickeln. Es liefert uns gleichsam durch IT-Magie den Blick in die Glaskugel.

Unternehmenslenker atmen auf, der Traum des Algorithmic Driven Enterprise scheint realisierbar: Endlich Entscheidungen treffen, die auf validen Prognosen basieren – unerbittlich effizient, Fakten-basiert und ohne persönliche Vorteile von Maschinen erzeugt. Unfehlbar? Hier sollten wir eigentlich gewarnt sein. Weit mehr als zehn Milliarden Euto werden weltweit für Wetterforschung und -prognosen ausgegeben, aber wer ist tatsächlich mit der Validität der Prognosen zufrieden?

Was ist an dem Hype „Predictive Analytics" wirklich dran? Diese Ausgabe der DOAG Business News hat dieses Thema zum Schwerpunkt und unsere Autoren haben die vielen Facetten ausgeleuchtet. An dieser Stelle einen herzlichen Dank für das rege Interesse, in der DOAG Business News zu publizieren.

Das Heft deckt hervorragend die Vielzahl der technischen Herausforderungen wie auch der unterschiedlichen Lösungsansätze von Predictive Analytics ab. Die Artikel umfassen thematische Domänen wie Dokument-Analyse, Wett-Prognosen, Machine-Learning-Ansätze und künstliche Intelligenz bis hin zu den unterschiedlichen grundlegenden Strategien des Datenmanagements zur Verwaltung der meist riesigen, oft dezentralen Daten.

Ich wünsche viel Spaß beim Lesen und gute Anregungen für die tägliche Arbeit.

Ihr

Impressum

DOAG Business News wird von der DOAG Deutsche ORACLE-Anwendergruppe e.V. (Tempelhofer Weg 64, 12347 Berlin, www.doag.org), herausgegeben. Es ist das User-Magazin rund um die Applikations-Produkte der Oracle Corp., USA, im Raum Deutschland, Österreich und Schweiz. Es ist unabhängig von Oracle und vertritt weder direkt noch indirekt deren wirtschaftliche Interessen. Vielmehr vertritt es die Interessen der Anwender an den Themen rund um die ORACLE-Produkte, fördert den Wissensaustausch zwischen den Lesern und informiert über neue Produkte und Technologien.

DOAG Business News wird verlegt von der DOAG Dienstleistungen GmbH, Tempelhofer Weg 64, 12347 Berlin, Deutschland, gesetzlich vertreten durch den Geschäftsführer Fried Saacke, deren Unternehmensgegenstand Vereinsmanagement, Veranstaltungsorganisation und Publishing ist.

Die DOAG Deutsche Oracle-Anwendergruppe e.V. hält 100 Prozent der Stammeinlage der DOAG Dienstleistungen GmbH. Die DOAG Deutsche Oracle-Anwendergruppe e.V. wird gesetzlich durch den Vorstand vertreten; Vorsitzender: Stefan Kinnen. Die DOAG Deutsche Oracle-Anwendergruppe e.V. informiert kompetent über alle Oracle-Themen, setzt sich für die Interessen der Mitglieder ein und führen einen konstruktiv-kritischen Dialog mit Oracle.

Redaktion:
Sitz: DOAG Dienstleistungen GmbH
(Anschrift s.o.)
Chefredakteur (ViSdP): Wolfgang Taschner
Kontakt: redaktion@doag.org
Weitere Redakteure (in alphabetischer Reihenfolge): Mylène Diacquenod, Marina Fischer, Sebastian Höing, Fried Saacke, Rolf Scheuch, Dr. Frank Schönthaler

Druck:
adame Advertising and Media GmbH, Berlin,
www.adame.de

Fotonachweis:
Titel: © Warakorn Harnprasop/123RF
S. 5: © alexlmx/123RF
S. 10: © Trevisto AG
S. 15: © Khoon Lay Gan/123RF
S. 19: © aimage/123RF
S. 23: © rawpixel/123RF
S. 30: © Sean Gladwell/Fotolia
S. 37: © ppbig/123RF

Titel, Gestaltung und Satz:
Caroline Sengpiel
DOAG Dienstleistungen GmbH
(Anschrift s.o.)

Anzeigen:
Simone Fischer, DOAG Dienstleistungen GmbH
(verantwortlich, Anschrift s.o.)
Kontakt: anzeigen@doag.org
Mediadaten und Preise unter: www.doag.org/go/mediadaten

Process Mining versteht sich als Nachfolger von „Data Mining“ und „Predictive Analysis“

Mit den richtigen Methoden wichtige Erkenntnisse aus allen gesammelten Daten ziehen

3 Editorial

3 Impressum

4 Inserenten

5 Verborgenes sichtbar machen: Visuelle Analyse komplexer Daten am Beispiel der Panama Papers
Dr. Thorsten Liebig und Karin Patenge

10 Beat the Bookie – TripleA-DWH
Andre Dörr

15 Process Mining für jedermann
Stephan La Rocca

19 Wie verschaffe ich mir einen Überblick über meine Daten?
Dr. Nadine Schöne

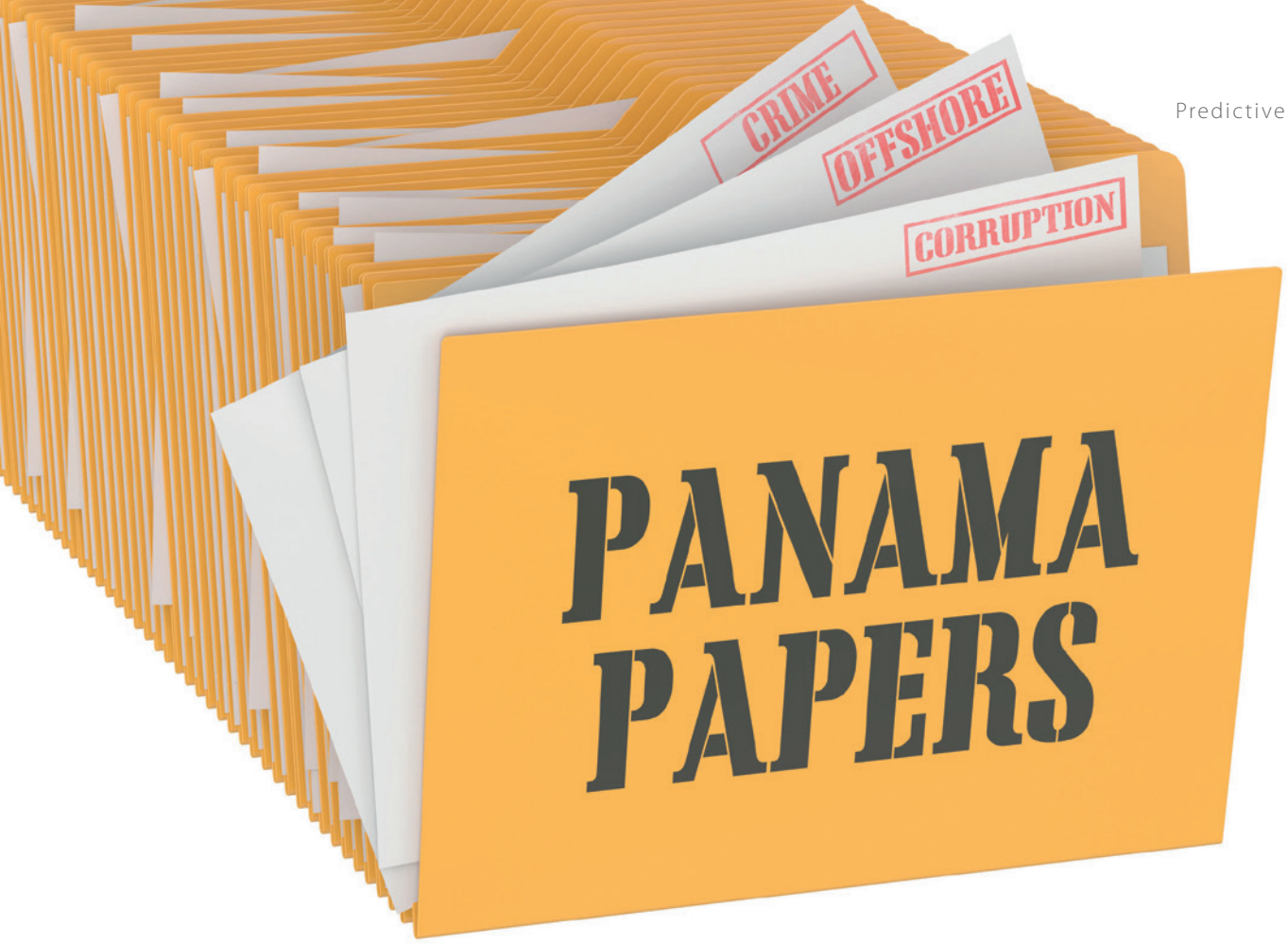
23 Daten als Chance
Sigrid Keydana

30 Predictive Machine Learning – Analysen und Massendaten
Alfred Schlaucher

37 Elastisch und skalierbar – Data Lake in der Oracle Cloud
Harald Erb

Die Herausforderungen bei der Anwendung von Machine-Learning-Verfahren auf große Datenmengen

Unsere Inserenten



Verborgenes sichtbar machen: Visuelle Analyse komplexer Daten am Beispiel der Panama Papers

Dr. Thorsten Liebig, derivo GmbH, und Karin Patenge, ORACLE Deutschland B.V. & Co. KG

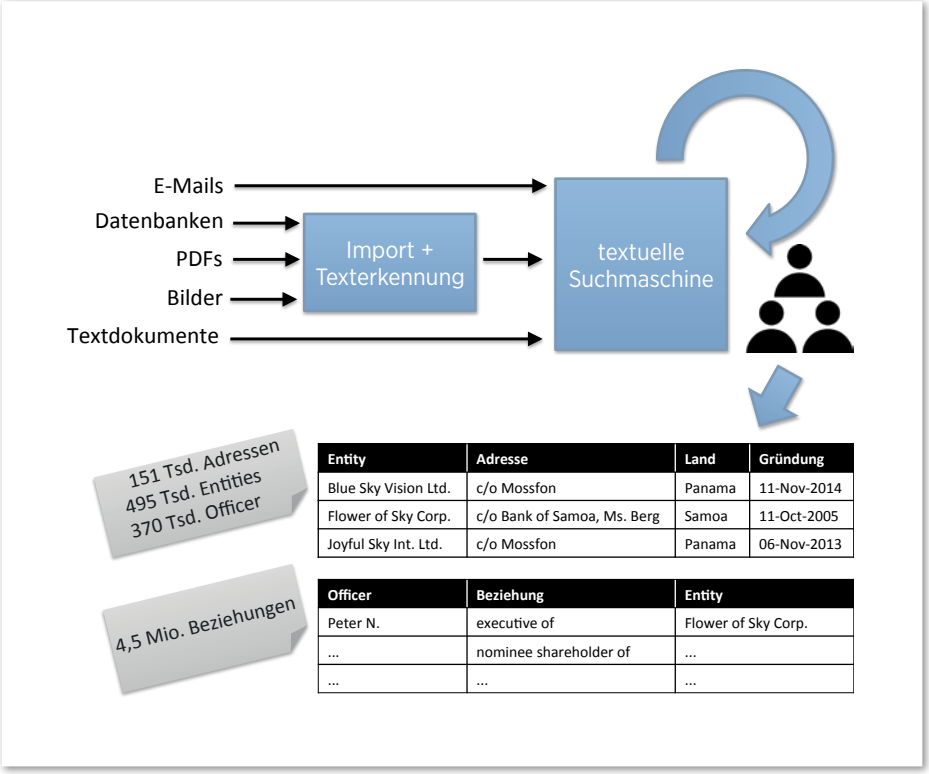
Komplex vernetzte Datenbestände sind schwer zu durchschauen, tauchen aber an geschäftskritischen Stellen überall auf. Der Artikel zeigt die Herausforderungen dieser Aufgabe und bietet einen Lösungsweg, um semantische Methoden mit einer innovativen Visualisierung zu verzahnen.

Data Analytics im Unternehmen hat das Ziel, geschäftsrelevante Informationen aufzuspüren, um begründete Schlüsse für unternehmerische Entscheidungen zu liefern. Klassische BI-Werkzeuge sind in diesem Zusammenhang dann hilfreich, wenn man die Struktur der Daten und seine Fragestellungen ausreichend kennt. Wie aber geht man vor, wenn der Datenbestand groß und komplex vernetzt ist und die aufschlussreichen Fragen aufgrund fehlender Übersicht nicht auf der Hand liegen? Wie lassen sich beispielsweise in den Panama Papers die Begünstigten einer Offshore-Unternehmung leicht finden, wenn deren Beteiligung bewusst über Strohmänner und indirekte Beteiligungen verschleiert ist?

Derartige Zusammenhänge in komplizierten, womöglich verteilten Datenbeständen aufzudecken, stellt eine Herausforderung dar. Dabei soll nicht der Eindruck entstehen, dass eine passende SQL-Abfrage oder das richtige BI-Dashboard keine wertvolle Informationen liefern können. Der kritische Punkt besteht oftmals darin, die richtigen Ideen für die aufschlussreichen Fragestellungen zu bekommen – insbesondere dann, wenn die zugrunde liegenden Daten oder das Datenschema nicht in allen Facetten bekannt sind.

Diese Situation ist immer wieder gegeben, da die Datenverwalter zumeist nicht die Auswerter der Daten sind. Häufig will man ja gerade eine Gruppe von Fachleuten, die keine Datenbank-Experten sind, dazu befähigen, eine Recherche zu betreiben, beispielsweise Steuerfahnder, die in den Panama Papers nach verschleiertem Besitz suchen. Ein anderer Anwendungsfall betrifft die Zusammenführung technischer Produktdaten mit Vertriebsdaten eines Automatisierungsherstellers, damit Produktmanager etwa die möglichen Risiken beim Auslauf, Austausch oder Lieferengpass von Produktteilen untersuchen können.

Diese beiden und weitere Anwendungsfälle lassen sich erfolgreich auf Basis semantischer Technologien und des darauf aufbauenden, interaktiven Visualisierungs- und Recherche-Werkzeugs SemSpect lösen. Der Artikel stellt diesen Lösungsansatz von der Aufbereitung und Anreicherung der Daten bis hin zur Analyse und Visualisierung der Daten am Beispiel der Panama Papers vor. Dabei wird auch erläutert, welche Rolle hierbei Ontologiesprachen wie OWL und RDFS spielen, wie sich Linked Open Data integrieren lässt und welche Oracle-Komponenten für die technische Umsetzung möglich sind.



Entity	Adresse	Land	Gründung
Blue Sky Vision Ltd.	c/o Mossfon	Panama	11-Nov-2014
Flower of Sky Corp.	c/o Bank of Samoa, Ms. Berg	Samoa	11-Oct-2005
Joyful Sky Int. Ltd.	c/o Mossfon	Panama	06-Nov-2013

Officer	Beziehung	Entity
Peter N.	executive of	Flower of Sky Corp.
...	nominee shareholder of	...
...	...	...

Abbildung 1: Extrahieren von Datenobjekten und deren Beziehungen am Beispiel der Panama Papers

11,5 Millionen verschiedenartige Dokumente

Anfang 2015 hat die Süddeutsche Zeitung 2,6 Terabyte an Daten in Form von 11,5 Millionen Dokumenten von einer anonymen Quelle erhalten. Diese werden inzwischen vom internationalen Konsortium investigativer Journalisten (ICIJ) verwaltet und mit Redaktionen unter anderem aus den USA, Frankreich und Spanien ausgetauscht. Ein sehr naheliegender Verdacht war, dass diese Daten Belege für die systematische Verschleierung von Besitzverhältnissen von Personen über Briefkastenfirmen mit Scheindirektoren enthalten.

Allerdings sind diese Belege verborgen in einer Vielzahl unterschiedlicher, größtenteils unstrukturierter Dokumente wie PDFs oder eingescannten Verträgen. Aus diesem Grund hat man Letztere mithilfe einer Texterkennung gemeinsam mit E-Mails, Textdokumenten etc. in die Suchplattform Apache Solr geladen. Nicht unähnlich zur alltäglichen Suche im Web mit Google wurden dort in einer Volltextsuche Informationen über Personen und deren Zusammenhänge parallel in den Redaktionen von Le Monde, Guardian, Süddeutsche Zeitung etc. Stück für Stück identifiziert. Die Informationen über Personen und Firmen sowie eine textuelle Beschreibung ihrer Beziehungen sind in Tabellen und einer Graph-Datenbank gespeichert.

Abbildung 1 zeigt diesen Arbeitsablauf schematisch. Auf diese Weise sind knapp eine Millionen Datenobjekte (Adressen, Firmen, Personen) und etwa 4,5 Millionen Beziehungen zwischen diesen entstanden, die inzwischen der Öffentlichkeit als CSV-Dateien sowie als Dump einer Graph-Datenbank (Neo4j) zugänglich sind.

Für journalistische Zwecke war es ausreichend, nach bestimmten Namen und deren direkten Verbindungen zu suchen. Wie aber findet man in den verfügbar gemachten Daten die Personen von Firmen mit Bezügen nach Deutschland? Welche anderen Firmen haben indirekte Beteiligungen zu solchen Firmen? Welche Muster sind hier erkennbar?

Man könnte meinen, dass diese Recherche direkt in der Graph-Datenbank möglich sein müsste. Leider sind die Daten jedoch sehr uneinheitlich erfasst, also mit und ohne Wortpräfixe. Zudem kommen verschiedene Abkürzungen sowie Synonyme vor. Auch unterschiedliche Schreibweisen und nicht zuletzt viele Schreibfehler erschweren die Analyse.

Datenkonsolidierung und Datenschema ergeben einen Wert

Selbst Daten nur einer Anwendung sind häufig aufgrund mehrerer Erfasser und unterschiedlicher Eingabemethodik nicht einheitlich und daher schwer auszuwerten. Die verfügbar gemachten Daten zu den Panama Papers sind ein Paradebeispiel hierfür. Was lässt sich tun? In diesem Fall kamen Werkzeuge zur Datenkonsolidierung und Anreicherung zum Einsatz und es wurde ein semantisches Datenmodell entwickelt. Dieses Vorgehen unterscheidet sich dabei nicht von anderen Anwendungsfällen aus Industrie oder Behörden.

Primäres Ziel ist es, ein quellübergreifendes, fachlich stimmiges Datenmodell zu erstellen und die Daten anhand dieses Modells zu konsolidieren. Man setzt hier konsequent auf ontologiebasierte Modelle mit formaler Semantik und wissenschaftlich bewiesenen Eigenschaften zu Komplexität und Berechenbarkeit. Die Web Ontology Language (OWL) oder das Resource Description Format (RDF), beides offizielle Standards des W3C, sind hier die Sprachen der Wahl.

Für die Panama Papers wurde auf OWL gesetzt. Für die Erstellung von Datenmodellen in OWL, die aus Klassen, Beziehungen und Datenwerten („class“, „object property“ und „data property“) bestehen, wurde der Ontologie-Editor Protégé eingesetzt. Im Fall

der Panama Papers waren die zentralen Klassen schnell gefunden. Der Datenbestand behandelt „Firmen“ (entities), „Mittelsmänner“ (intermediaries), „Angestellte“ (officer) sowie „Länder“ und „Adressen“.

Das Beziehungsmodell, das diese Klassen in Relation setzt, erfordert etwas Überblick über die eingetragenen Verknüpfungstexte. Aus den knapp dreihundert verschiedenen Verknüpfungen sind für das Ontologiemodell dreizehn inhaltlich unterschiedliche Beziehungen identifiziert und definiert worden. *Abbildung 2* zeigt die Klassen und Beziehungen zu den Panama Papers, so wie sie in Protégé dargestellt sind. Für Steuerfachleute ist dieses Modell eventuell zu grob oder auch fachlich schief. Hier liegt die Stärke des semantischen Ansatzes. Die Ontologie ist sehr flexibel, weil sie von der technischen Datenhaltung nicht so abhängig ist wie etwa das relationale Modell. Auch zur Laufzeit ist es möglich, Klassen zu verfeinern oder neu einzuführen.

Das Besondere an einem ontologischen Modell ist zudem, dass Wissen über die Daten im Schema verankert werden kann. Es steht zur Zeit des Datenzugriffs automatisch zur Verfügung. Neben einer Hierarchie von Klassen und Beziehungen lassen sich weitere implizite Datenabhängigkeiten mithilfe sogenannter „Axiome“ definieren. So lässt sich beispielsweise sehr leicht und nachvollziehbar ausdrücken, dass Firmen in Deutschland jene sind, deren Gerichtsbarkeit in Deutschland liegt.

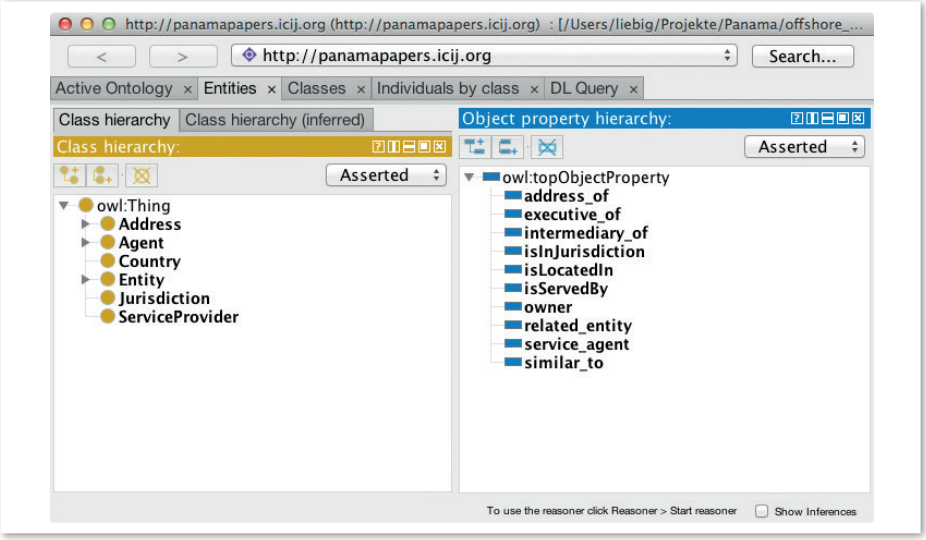


Abbildung 2: Klassen und Beziehungs-Hierarchie im Ontologie-Editor Protégé

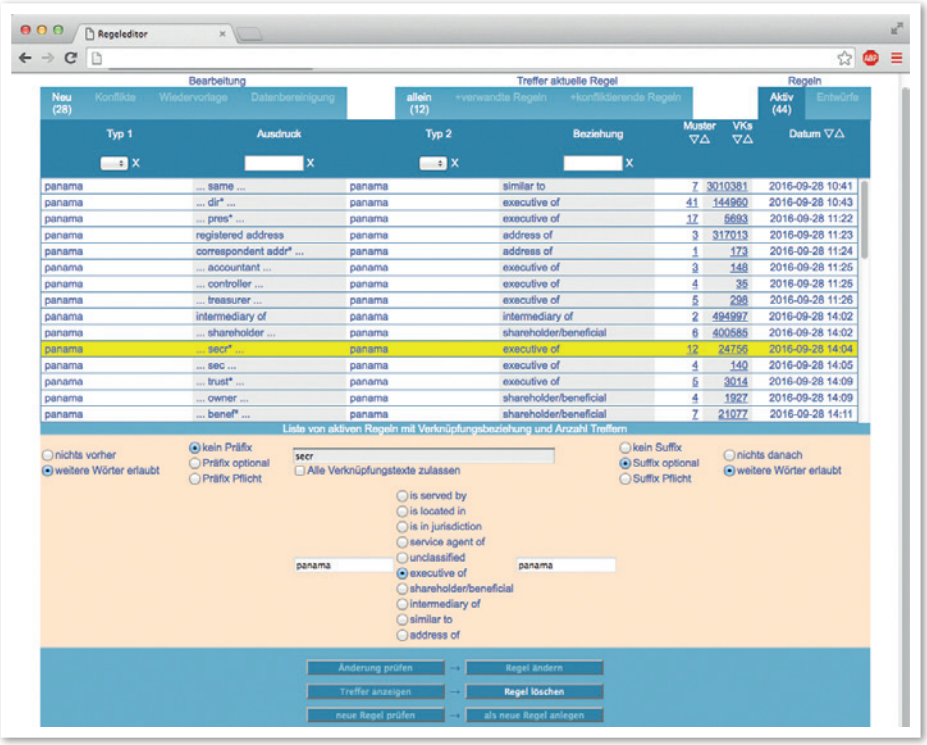


Abbildung 3: Regelbasiertes Konsolidierungswerkzeug mit Panama-Paper-Daten

Aus Spalten wird eine Bedeutung

Für die Abbildung der 4,5 Millionen einliegenden Verknüpfungsaussagen auf die dreizehn Beziehungen des Ontologie-Modells kam ein interaktives Datenkonsolidierungswerkzeug zum Einsatz. Dabei werden inhaltlich gleiche Verknüpfungsaussagen zusammengefasst und vom Benutzer über einfach zu definierende Muster in eine Beziehung des semantischen Datenmodells überführt.

Die 4,5 Millionen Verknüpfungen aus dem Panama Papers konnten beispielsweise mit diesem Werkzeug mit weniger als vierzig Regeln auf die dreizehn Beziehungen des Modells in etwa einer Stunde vereinheitlicht werden. *Abbildung 3* zeigt die Benutzerschnittstelle dieses Werkzeugs.

Dabei ist es gleichzeitig möglich, die Daten mit zusätzlichen Informationen aus eigenen oder offenen Quellen anzureichern. Hier ist es von Vorteil, dass OWL – die Web Ontology Language – kompatibel zu den Formaten und Standards von Linked Open Data (LOD) ist. Auf diese Weise kann man Daten aus Wikidata, GeoNames etc. einfach und schnell für den eigenen Datenbestand nutzen.

Graph oder Ontologie?

Was unterscheidet Graphen und Ontologien? Das in Graph-Datenbanken vorherrschende Datenmodell ist der sogenannte „Property Graph“. Er besteht aus Knoten und Kanten, die jeweils individuelle Eigenschaften (Typ, Name etc.) – ihre sogenannte „Properties“ – besitzen können. Daten auf Grundlage einer Ontologie haben ebenfalls eine Graph-Struktur aus Objekten und Beziehungen, die auch Eigenschaften erhalten können.

Abgesehen von syntaktischen Feinheiten besteht hier große Übereinstimmung. Die Unterschiede und die Wahl des Modells bestehen daher eher in den Zielen der Auswertung. Graph-Datenbanken sind dafür gemacht, Topologiefragen wie Erreichbarkeit, Zentralität oder Influenz effizient zu beantworten, wohingegen bei Ontologien das Ableiten von impliziten Fakten im Vordergrund steht.

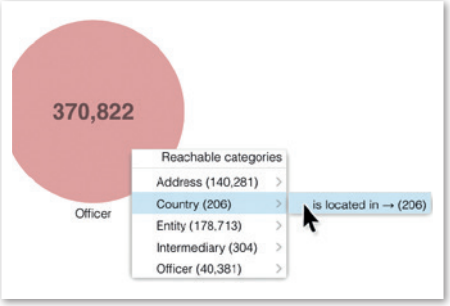


Abbildung 4: Die „Officer" in den Panama Papers und deren verknüpfte Klassen mit Beziehungen

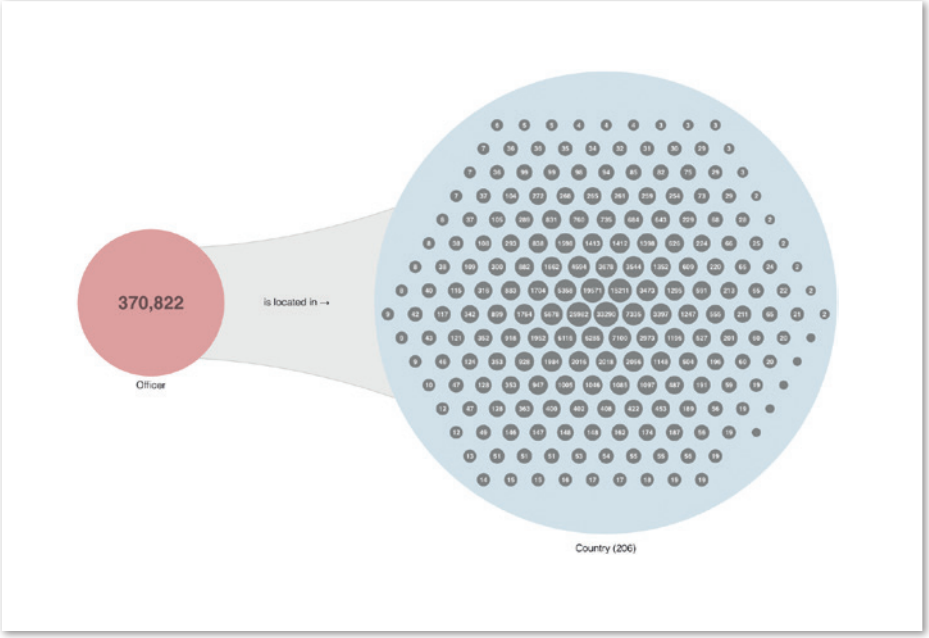


Abbildung 5: Exploration der „Angestellten" (officer) auf ihre Länder

Überblick zuerst – Details nach Bedarf

Wie lässt sich nun ein Graph (Netzwerk) aus Millionen von Objekten (Knoten) und Beziehungen (Kanten) verstehen und erforschen? Nach den Erfahrungen der Autoren liegt der Schlüssel in der datengetriebenen Exploration und kontextsensitiven Visualisierung. SemSpect ist ein Werkzeug, das diese Strategie konsequent umsetzt und auch Nutzer ohne Erfahrung in Abfragesprachen anhand der visuellen Darstellung der Daten befähigt, anspruchsvolle Recherche-Aufgaben in komplexen Datenbeständen zu bewältigen.

Ein Beispiel sind die „Angestellten" (officer) aus den Panama-Paper-Daten. Wie sind diese mit Datenobjekten aus den anderen Klassen verbunden? In SemSpect werden die mehr als 370.000 Angestellten zusammengefasst und als Gruppe (grafisch als Kreis) dargestellt. Für diese Gruppe lassen sich direkt aus einem Kontextmenü heraus die erreichbaren Klassen und Beziehungen anzeigen *(siehe Abbildung 4)*.

Durch Auswahl einer Beziehung zeigt die Gruppe ihre verbundenen Datenobjekte, ebenfalls in einer Gruppe zusammengefasst *(siehe Abbildung 5)*. Auf diese Art und Weise kann eine Exploration der Daten vorgenommen werden, die sich visuell wie ein (umgestürzter) Baum von links nach rechts aufspannt. Im Unterschied zu gewöhnlichen Graph-Visualisierungen sind in SemSpect Datenobjekte und einzelne Beziehungen aggregiert dargestellt, um auch bei großen Datenmengen die Übersicht zu bewahren.

Datenobjekte wie die Länder der Angestellten sind als einzelne Punkte innerhalb einer Gruppe dargestellt, solange ihre Anzahl unterhalb eines einstellbaren Schwellwerts liegt. Die Zahl in einem Datenobjekt gibt die Anzahl der verknüpften Objekte in der Vorgängergruppe an. Wird ein Datenobjekt selektiert, sind dessen Detail-Informationen sowie die direkt und indirekt verknüpften Objekte (sofern dargestellt) hervorgehoben.

Wenn man eine Gruppe selektiert, werden die Untergruppen sowie konfigurierbare Eigenschaften der Gruppe – die sogenannten „Facetten" – angezeigt. Damit lassen sich die Gruppen einer Exploration selektiv filtern. Für jede Gruppe und jede Gruppenverknüpfung ist auch eine Tabellenansicht verfügbar. In dieser lässt sich die Exploration ebenfalls etwa über Suchbegriffe filtern. Auf diese Weise können Schritt für Schritt durch Expansion des Expansionsgraphen und interaktive Filterung sehr komplexe Anfragen formuliert und deren Ergebnisse sofort grafisch dargestellt werden. *Abbildung 6* zeigt links die Gesamtoberfläche von SemSpect mit dem Klassenbaum und gespeicherten Explorationen, in der Mitte die aktuelle Exploration und Tabellenansicht sowie rechts die Facetten einer Gruppe.

Es gibt weitere Funktionen in SemSpect, die die Recherche unterstützen (Markierungen, Speicherung von Zwischenergebnissen etc.). Für das Reporting lassen sich Gruppen und ganze Explorationen strukturiert exportieren beziehungsweise als Bilder abspeichern.

Die Technik dahinter

SemSpect ist eine Client-Server-Anwendung mit einer HTML5/JavaScript-Web-UI, optimiert für die Desktop-Benutzung (IE11 oder aktueller Chrome- beziehungsweise Firefox-Browser), sowie einem REST-Backend in Java 1.8. Die Schlussfolgerungs- und Anfrage-

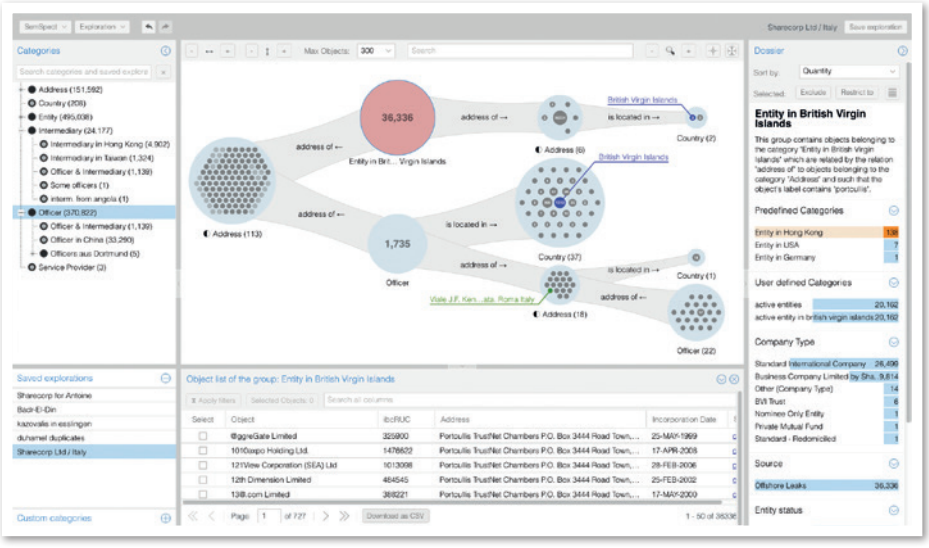


Abbildung 6: Beispiel-Exploration in der Darstellung in SemSpect

Bearbeitung im Backend übernimmt GraphScale, eine patentierte RDFS/OWL-Technologie für Datenbanken. Prinzipiell arbeitet GraphScale mit relationalen und NoSQL-Datenbanken. Generell gilt, dass sich Graph-Datenbanken schon aufgrund der in diesem Artikel erwähnten Verwandtschaft von Ontologien und Graphen am besten eignen.

Es existieren aktuell Anbindungen an Oracle-Datenbank 12c Spatial and Graph, an Oracle Big Data Spatial and Graph mit der Oracle-NoSQL-Datenbank für die Datenhaltung sowie an Neo4j. Auch wenn GraphScale keine native Unterstützung von OWL oder RDFS voraussetzt, bleibt zu bemerken, dass Oracle-Datenbank 12c Spatial and Graph als einziges Backend diese Standards und auch die Abfragesprache SPARQL nativ beherrscht.

Das vorgestellte Datenkonsolidierungswerkzeug ist ebenfalls eine Web-Anwendung und kann mit unterschiedlichen Datenquellen betrieben werden. Im vorliegenden Fall der Panama Papers lief das System auf einem Oracle WebLogic Server.

Fazit

Viele Daten haben inhärent eine Graph-Struktur; die Panama Papers sind ein gutes Beispiel dafür. Nutzt man diese Tatsache und setzt auf semantische Methoden und darauf abgestimmte Visualisierungs- und Abfragesysteme, lässt sich Verborgenes in Daten sichtbar machen. Sozialwissenschaftler aus Frankreich haben mit SemSpect in den Panama Papers auf Anhieb Hypothesen zu Netzwerken von vermögenden und einflussreichen Personen validieren können. Offen ist zum jetzigen Zeitpunkt, welche Verarbeitungsmethoden und Auswertungsverfahren das BKA und die Steuerbehörden für die kürzlich angekauften Rohdaten der Panama Papers einsetzen.

Aber auch im Maschinenbau, in der Pharmaforschung, in sozialen Medien etc. tauchen Datenbestände als Graphen auf. Das bedeutet im Umkehrschluss nicht, dass relationale Datenbanken obsolet sind, sondern es deutet darauf hin, dass für das tiefe Verständnis komplexer Daten und die Nutzung des darin verborgenen Wissens ergänzende Ansätze benötigt werden, die für Graphen optimiert sind.

Der in diesem Artikel exemplarisch anhand der Panama Papers beschriebene Ansatz semantischer Methoden mit einer für Graphen optimierten Benutzerschnittstelle ist nach Erfahrung der Autoren auf andere Anwendungsbereiche direkt übertragbar. Die bereits nach diesem Muster produktiv gesetzten Lösungen stiften Nutzen in Industrie und Behörden durch eine höhere Effektivität und Erkenntnisgewinn bei der Recherche sowie einen größeren Benutzerkreis durch die intuitive Benutzerschnittstelle.

Die IT-Landschaft muss dabei nicht völlig ausgetauscht werden. Der vorgestellte Lösungsansatz arbeitet, wie im vorherigen Paragraph beschrieben, sehr gut mit bewährten System-Komponenten wie der Oracle-Datenbank mit ihrer Erweiterung für Graphenmodelle zusammen. Die Erweiterung des Oracle-Portfolios in Bezug auf Datenmanagement-Plattformen mit Oracle Big Data sowie dem Aufsatz Big Data Spatial and Graph ermöglicht auch die Speicherung und Auswertung von Graphen auf einer NoSQL-Datenbank beziehungsweise Apache-Hadoop/HBase. Dies ist ein wichtiger Schritt zur Vereinigung etablierter und innovativer Technologien.

Weiterführendes Material

- Ontologien – eine Definition: *https://de.wikipedia.org/wiki/Ontologie_(Informatik)*
- Panama Papers mit SemSpect: *http://panama.semspect.de*
- Weitere SemSpect-Demos:
  - SpringerNature SciGraph: *http://scigraph.semspect.de*
  - US Legislative von GovTrack.us: *http://govtrack.semspect.de*
- GraphScale Schlussfolgerungstechnologie: *http://derivo.de/products/graphscale*
- Die Oracle-Datenbank jenseits von Entity-Relationship-Modellierung – vorhandenes Wissen repräsentieren und neues mit RDF Graph generieren, DOAG News, Ausgabe 06/2015: *https://www.doag.org/formes/pubfiles/7603500/docs/Publikationen/DOAGNews/2015/06-2015/06-2015-DOAG_SOUG_News_Karin_Patenge-Die_Oracle-Datenbank_jenseits_von_Entity-Relationship-Modellierung_vorhandenes_Wissen_repr%C3%A4sentieren_und_neues_mit_RDF_Graph_generieren.pdf*
- Spatial and Graph Analytics with Oracle Database 12c Release 2: *http://www.oracle.com/technetwork/database-options/spatialandgraph/spatial-and-graph-wp-12c-1896143.pdf*
- Protégé: *http://protege.stanford.edu*

Dr. Thorsten Liebig
liebig@derivo.de

Karin Patenge
karin.patenge@oracle.com

Beat the Bookie – TripleA-DWH

Andre Dörr, Trevisto AG

Sportwetten sind in den letzten Jahren zu einem Milliardengeschäft geworden. Es vergeht kaum eine Werbepause während einer Sportübertragung, in der kein TV-Spot eines Wettanbieters zu sehen ist. Doch ist es überhaupt möglich, mit Sportwetten Geld zu verdienen?

Dieser Artikel zeigt am Beispiel der Vorhersage von Fußball-Ergebnissen, wie auf Basis des Architektur-Konzepts „TripleA-DWH" (Advanced-Agile-Analytical-DWH) ein Predictive System aufgebaut werden kann. Die Grundidee des Konzepts beruht auf der Verschmelzung von Data Vault 2.0 und Predictive Analytics.

Mit normalem Fachwissen ist es nicht möglich, systematisch Geld mit Sportwetten zu verdienen. Die Quoten der Buchmacher sind sehr genau und werden schnell angepasst, da heutzutage so gut wie jede Information im Internet verfügbar ist. Um gegenüber dem Buchmacher im Vorteil zu sein, muss man sich mit Statistiken und Vorhersagemodellen beschäftigen. Ab diesem Zeitpunkt bewegt man sich auf dem Gebiet von Predictive Analytics.

Predictive Analytics bedeutet unter anderem, unterschiedliche Vorhersagemodelle zu entwickeln, zu testen und auszuführen. Idealerweise wird dies durch eine gute System-Architektur unterstützt. Data Vault 2.0 löst aktuell klassische DWH-Architekturen (Inmon, Kimball) ab, da es einige Nachteile dieser Ansätze ausgleichen kann. Verbindet man nun die beiden Themengebiete Data Vault 2.0 und Predictive Analytics, ergibt sich ein Architektur-Konzept, das man als „TripleA DWH" (Advanced Agile Analytical) bezeichnen kann. Im Folgenden wird zunächst erläutert, wie der Aufbau dieser Architektur aussieht und worin genau die Vorteile liegen. Darauf aufbauend wird anhand eines Beispiels für ein Vorhersagemodell von Fußball-Ergebnissen die Arbeitsweise mit solch einer Architektur aufgezeigt.

Architektur-Konzept

Die komplette Architektur basiert auf der Data-Vault-2.0-Referenz-Architektur und sie besteht aus vier Schichten *(siehe Abbildung 1)*. Der Stage Layer übernimmt die gleichen Funktionen wie bei klassischen DWH-Architekturen. Er dient als temporärer Zwischenspeicher innerhalb des Systems, bevor die Daten in die nächsten Schichten verarbei-

tet werden. An dieser Stelle können zum Beispiel bereits Datentyp-Überprüfungen durchgeführt werden. Danach werden die Daten in den Raw-Data-Layer übertragen. Diese Schicht wird zum dauerhaften, integrierten und historisierten Speichern aller Rohdaten verwendet. Damit bildet sie den „Single Point of Facts". Aufbauend darauf werden die Rohdaten im Analytical Layer mit weiteren Informationen – den Features und Ergebnissen der Vorhersagemodelle – angereichert. Bei den Features handelt es sich um die Variablen und Prädiktoren, die für eine Vorhersage benötigt werden. Im Information Layer werden die Daten für abnehmende Systeme aufbereitet. Die Ergebnisse verschiedener Modelle können beispielsweise in Berichten miteinander kombiniert werden, oder es werden beispielsweise dimensionale Datenmodelle für BI-Tools zur Verfügung gestellt.

Agile

Für den Raw Data Layer und den Analytical Layer werden die Datenmodellierungstechniken von Data Vault 2.0 genutzt. Ein entscheidendes Charakteristikum der Data-Vault-Modellierung ist die Trennung der Daten in Objekte („Hubs"), Beziehungen („Links") und Kontexte („Satelliten"). Dadurch besitzt Data Vault Eigenschaften, die eine agile Entwicklung innerhalb eines DWH ermöglichen:

- *Pattern Based Loading*
  Data Vault kennt nur drei verschiedene Typen von Tabellen: Hubs, Links und Satelliten. Jeder Typ besitzt die gleiche Grundstruktur. Die Ladeverfahren aller Typen sind standardisiert. Diese Standardisierung ermöglicht eine automatisierte Generierung sowohl der Data-Vault-Strukturen als auch der benötigten ELT-Prozesse zur Beladung eines Datenmodells.

- *Zero Impact*
  Data Vault verfolgt einen harten Zero-Impact-Ansatz. Dies bedeutet, dass die Anbindung neuer Datenquellen oder die Erweiterungen bestehender Datenquellen keinen Einfluss auf existierende Strukturen und Prozesse haben. Neue Vorhersagemodelle können beispielweise dem Analytical Layer hinzugefügt werden, indem sie als neuer Kontext mit dem entsprechenden Objekt verbunden werden. Dies sorgt nicht nur dafür, dass Erweiterungen eines bestehenden Modells extrem flexibel sind, sondern auch dafür, dass sogar Regressionstests entfallen können.

- *Sandbox Prototyping*
  Die Entwicklung und Optimierung von Vorhersagemodellen ist eine komplexe Aufgabe, die in der Regel mithilfe von separaten Tools (etwa R-Studio) durchgeführt wird. Die Einführung eines Sandbox Prototyping bietet hier die Möglichkeit, dieses Vorgehen in einen agilen Prozess zu überführen. Dabei wird wieder die Flexibilität eines Data-Vault-Datenmodells genutzt. Dem Endanwender wird eine vom restlichen Verarbeitungsprozess gekapselte Sandbox innerhalb der Datenbank zur Verfügung gestellt. Features und Ergebnisse von Vorhersagemodelle, die sich in der Entwicklung befinden, lassen sich als neuer Kontext (Satelliten) innerhalb dieser Sandbox abspeichern. Eine Integration in die bestehenden Daten findet dabei automatisch statt. So können bestehende und neue Vorhersagemodelle miteinander verglichen werden. Erweist sich ein Vorhersagemodell als effektiv, wird es im Anschluss in den Standardverarbeitungsprozess integriert.

Analytical

Viele Datenbankhersteller (wie Oracle, Microsoft, Exasol) haben mittlerweile das Potenzial und die Relevanz von statistischen Programmiersprachen erkannt und bieten an, die Option „R Code" innerhalb der Datenbank auszuführen. Dies bedeutet architektonische Vorteile für den Aufbau eines Predictive Systems.

Bisher wurden die Berechnungen der Vorhersagemodelle meist auf externen Servern durchgeführt. Dazu war es nötig, alle erforderlichen Features zunächst zu exportieren und die Ergebnisse der Vorhersage danach wieder zu importieren. Diese Schritte können nun komplett entfallen. Die Ausführung der Vorhersagemodelle kann direkt auf den Daten innerhalb der Datenbank erfolgen. Dies verringert die Komplexität der Architektur und der Verarbeitungsprozesse eines Predictive Systems.

Advanced

Wie bereits erwähnt, bildet Data Vault 2.0 die Grundlage dieses Architektur-Konzepts. Dessen Datenmodellierungsmethoden beinhalten eine entscheidende Verbesserung gegenüber der Version 1.0: Es werden Hash-Keys statt künstlicher Schlüssel für die eindeutige Identifizierung eines Objekts genutzt. Die Identifizierung durch Hash-Keys erlaubt es, ein Data-Vault-Modell über mehrere Technologien hinweg zu modellieren und zu implementieren. Dadurch besteht die Möglichkeit, in der Gesamt-Architektur ein relationales DBMS und NoSQL-Technologie zu kombinieren *(siehe Abbildung 2)*.

Damit ergeben sich Potenziale für verschiedenartige Use Cases. Im einfachsten Fall könnte eine NoSQL-Technologie als reiner Stage Layer genutzt werden. So lässt sich eine einfachere Schnittstellen-Anbindung durch das Nutzen von Schema-on-Read umsetzen. Eine weitere Möglichkeit ist das effektive Auswerten von unstrukturierten Daten. Dabei werden die unstrukturierten Informationen in einer NoSQL-Technologie gespeichert. Die

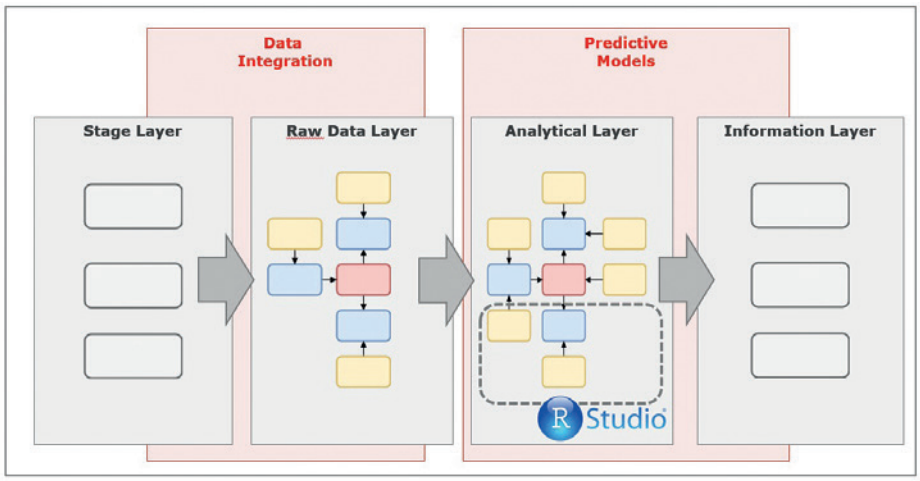


Abbildung 1: Die Layer-Übersicht

strukturierten Informationen liegen weiterhin in einem relationalem DBMS. Da die Datenmodellierung jedoch übergreifend über beide Technologien stattfindet, können strukturierte und unstrukturierte Daten gemeinsam ausgewertet werden.

Dieses Architektur-Konzept kann als Grundlage für diverse Predictive Systeme genutzt werden. Im Folgenden wird am Beispiel der Entwicklung eines Vorhersagemodells für Fußball-Ergebnisse aufgezeigt, wie sich auf Basis dieser Architektur Modelle entwickeln, validieren und implementieren lassen.

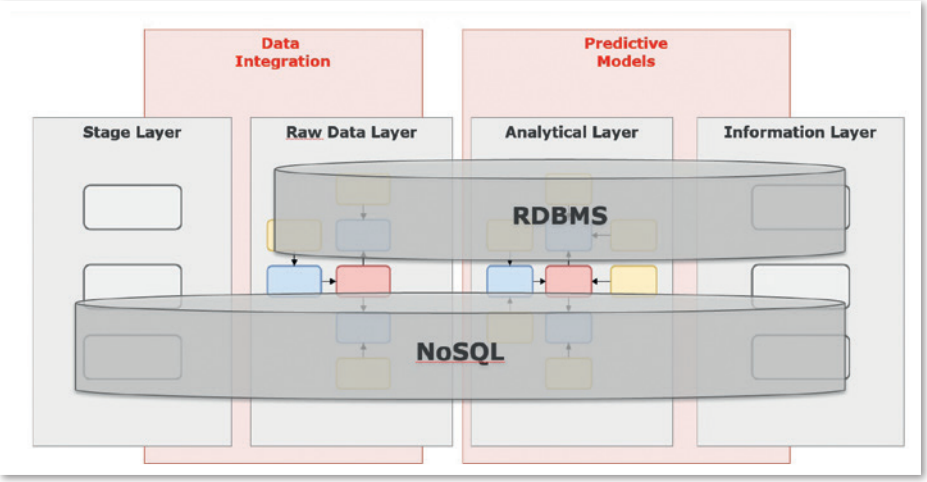


Abbildung 2: Integration SQL und NoSQL

Vorhersagemodell für Fußball-Ergebnisse

Es existieren verschiedenste grafische Darstellungen für den Entwicklungsprozess eines Vorhersagemodells. Alle Darstellungen haben jedoch gemeinsam, dass sie aus ähnlichen Einzelschritten bestehen. Auf Basis der Problem- und Ziel-Definition wird das Vorhersagemodell entwickelt und optimiert, bis das gewünschte Ergebnis erreicht ist. Danach kann das Modell im produktiven Einsatz genutzt werden *(siehe Abbildung 3)*.

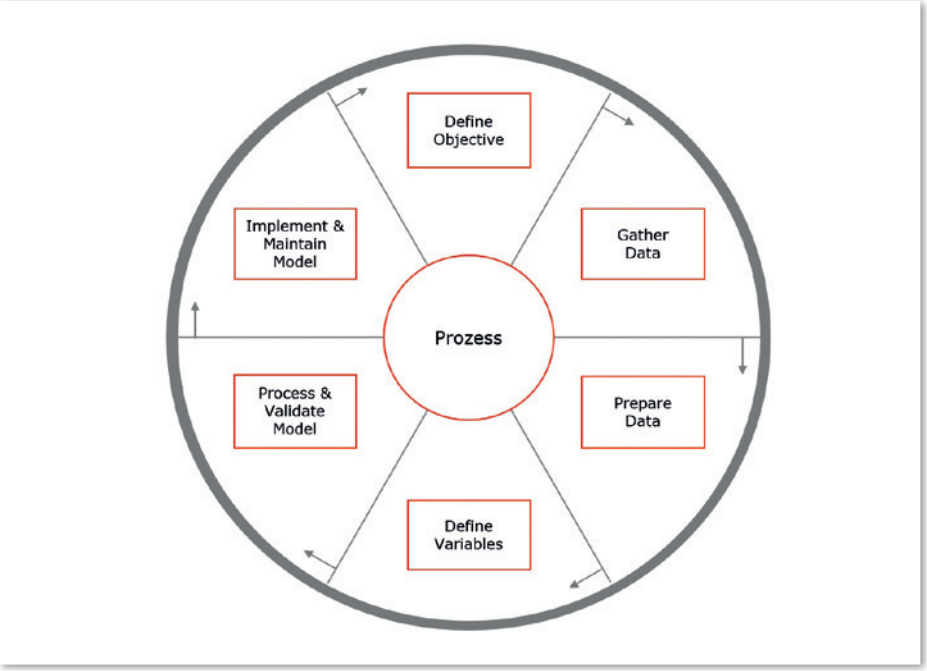


Abbildung 3: Entwicklungsprozess für Vorhersagemodelle

Der erste Schritt besteht darin, Problemstellung und Ziel genau zu definieren („Define Objective"). Dies ist notwendig, um darüber Klarheit zu erlangen, um was für ein Vorhersage-Problem es sich handelt. Regressionsprobleme erfordern beispielsweise andere Methoden als Klassifikationsprobleme. Bei der Vorhersage von Fußball-Ergebnissen (Heimsieg, Unentschieden, Auswärtssieg) handelt es sich um ein typisches Klassifikationsproblem. Einen Ansatz für dieses Klassifikationsproblem lieferten Dixon & Coles (Modelling Associated Football Scores and Ineffieciences in the Football Market, 1995). Sie beschreiben die Anzahl der Tore in einem Fußballspiel als eine Poisson-Verteilung *(siehe Abbildung 4)*.

Die Poisson-Verteilung ist eine diskrete Wahrscheinlichkeitsverteilung für eine bestimmte Anzahl von unabhängigen Ereignissen in einem festen Zeitintervall mit konstantem Mittelwert. Für die Berechnung werden nur zwei Parameter benötigt: der erwartete Mittelwert („λ") der Verteilung und die Anzahl der Ereignisse („X"). In den Bundesliga-Saisons von 2011 bis 2016 sind im Durchschnitt pro Spiel 2,89 Tore gefallen. *Abbildung 5* zeigt die reale Verteilung der Tore. Ist die Wahrscheinlichkeit für eine gewisse Anzahl von Toren bekannt, die das Heim- und das Auswärtsteam schießt, kann damit auch die Wahrscheinlichkeit für Heimsieg, Unentschieden oder Auswärtssieg berechnet werden.

$$P(X = x) = \frac{\lambda^x e^{-\lambda}}{x!}$$

Abbildung 4: Formel für Poisson-Verteilung

Nachdem die Problemstellung und das Ziel genauer definiert sind, besteht die nächste Aufgabe darin, Datenquellen zu suchen, die die benötigten Daten für eine Vorhersage liefern („Gather Data"). Dies können externe und interne Datenquellen sein. Das Internet stellt dabei natürlich die größte Quelle dar. Einige Firmen haben ihr gesamtes Geschäftsmodell auf das Sammeln und Verkaufen von Daten ausgelegt. In Bezug auf Fußball-Daten ist hier zum Beispiel Opta zu nennen. Im vorliegenden Beispiel wurde die Internet-Seite „football-data.co.uk" genutzt, die historische Fußball-Statistiken für mehr als zwanzig Ligen und bis zu zwanzig Jahren bereitstellt. Da das Ergebnis eines Vorhersagemodells stark von der Qualität der Daten abhängig ist, müssen diese Daten vor der Verarbeitung geprüft und notfalls korrigiert werden („Prepare Data"). Sind die Daten vollständig? Gibt es Lücken in den Daten? Sollten diese Lücken gefüllt werden? Existieren Ausreißer, die das Er-

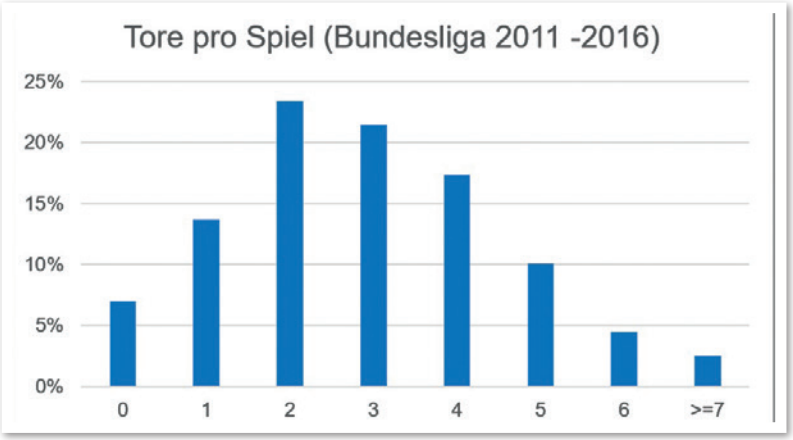


Abbildung 5: Torverteilung Bundesliga (2011 – 2016)

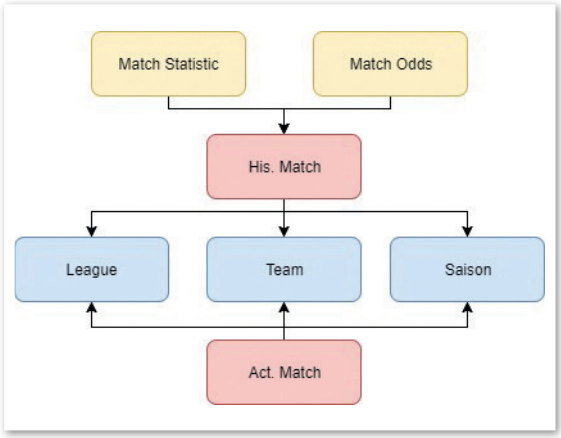


Abbildung 6: Raw-Layer-Data-Vault-Modell

gebnis verfälschen können? All dies sind Faktoren, die Einfluss auf die Genauigkeit eines Modells haben.

Bei der Implementierung eines Vorhersagemodells auf der TripleA-DWH-Architektur ist jedoch noch ein weiterer Schritt notwendig. Die verschiedenen Datenquellen sind in ein Data-Vault-Modell zu überführen, um die Flexibilität der Architektur für den weiteren Entwicklungsprozess nutzen zu können. *Abbildung 6* zeigt das entstehende Data-Vault-Modell für die verwendeten Rohdaten.

Die Objekte „Liga", „Team" und „Saison" sind als Hubs (blau) definiert. Ein Match stellt die Beziehung (rot) zwischen den drei Objekten dar – zwei Teams spielen in einer Liga in einer Saison zu einem gewissen Zeitpunkt gegeneinander. Diese werden unterschieden in historische Spiele und aktuelle Spiele. Die historischen Spiele dienen dem Simulieren von Vorhersagemodellen. Auf die aktuellen Spiele müssen die entwickelten Modelle angewandt werden. Für die historischen Spiele existieren zwei verschiedene Kontexte (gelb) – die Spiel-Statistik und die Wettquoten bei verschiedenen Wettanbietern. Dieses Datenmodell bildet die Basis für die weiteren Schritte und wird nach und nach erweitert.

Einer der wichtigsten Schritte vor der Erstellung eines Vorhersagemodells ist die Feature- oder auch Variablen-Selektion („Define Variables"). Die Verwendung schlechter Variablen für ein Vorhersagemodell führt auch zu schlechten Ergebnissen. Dabei sollte das Motto „weniger ist manchmal mehr" beachtet werden. Zu viele Variable können zu einem Overfitting führen oder das Modell unnötig verkomplizieren. Bei Machine-Learning-Algorithmen verlängert sich die Trainingszeit mit der Anzahl der Variablen.

Ein Beispiel für ein schlechtes Feature bei einer Fußball-Vorhersage ist der Ballbesitz, wie das Champions-League-Finale 2012 zwischen Bayern München und Chelsea London gezeigt hat. Für die Vorhersage von Fußball-Ergebnissen mit der Poisson-Verteilung werden Variablen benötigt, welche es ermöglichen, die erwartete Anzahl von Toren für die Heim- und die Auswärtsmannschaft zu berechnen.

In dem Vorhersagemodell von Dixon & Coles werden dafür die Angriffs- und Verteidigungsstärke der jeweiligen Mannschaften genutzt. Diese Variablen repräsentieren das Verhältnis der Anzahl der Tore beziehungsweise Gegentore eines Teams zum Liga-Durchschnitt. Wird mithilfe dieser Variablen beispielsweise die erwartete Tore-Anzahl für das Spiel von Bayern München gegen Schalke 04 am 4. Februar 2017 berechnet, ergeben sich für die Heimmannschaft 2,57 und für die Auswärtsmannschaft 0,35 erwartete Tore. *Abbildung 7* zeigt die Wahrscheinlichkeitsverteilung für die Heim- und die Auswärtsmannschaft. Das Spiel endete 3:0, dem Ergebnis mit der zweithöchsten Wahrscheinlichkeit.

Um die Simulation eines Vorhersagemodells durchzuführen, sind die Variablen für alle verfügbaren historischen Daten zu berechnen. *Abbildung 8* zeigt das um den Satelliten „Attack Defence Strength" erweiterte Datenmodell.

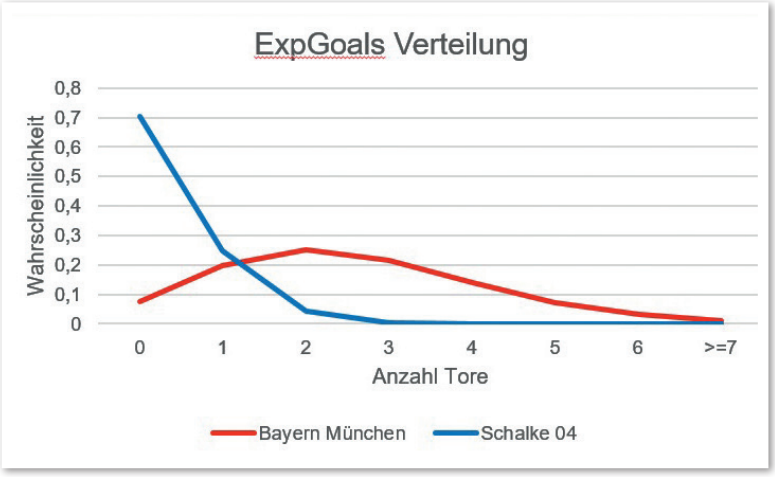


Abbildung 7: Verteilung erwartete Tore Bayern München gegen Schalke 04 (4. Februar 2017)

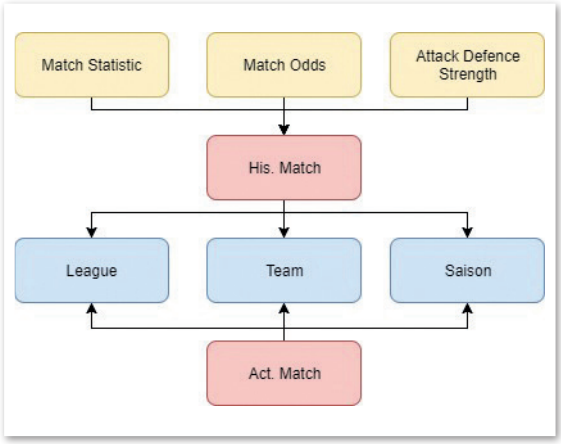


Abbildung 8: Analytical-Layer-Data-Vault-Modell mit Feature-Berechnung

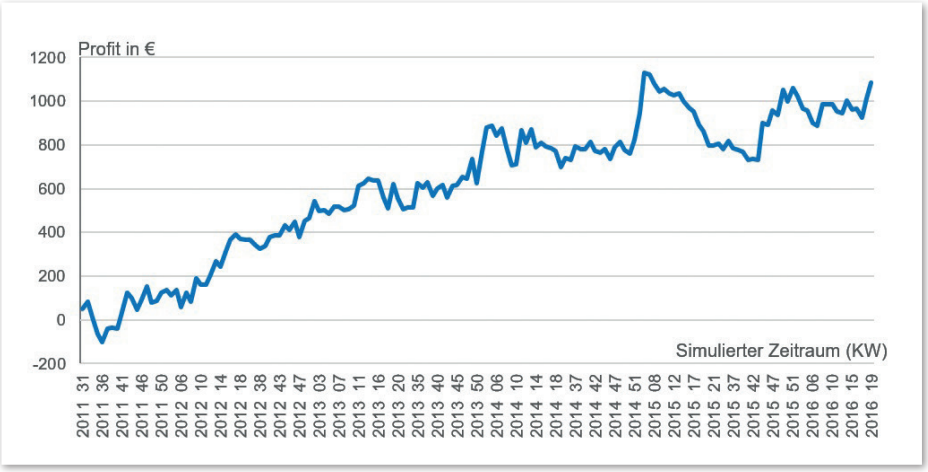


Abbildung 9: Modell-Simulation für Bundesliga 2011-2016

Hier zeigt sich die Flexibilität eines Data-Vault-Datenmodells. Neue Features werden einfach als neuer Kontext („Satellit") in das Datenmodell integriert, ohne dass bestehende Strukturen und Prozesse angepasst werden müssen. Genauso verhält es sich, wenn Features nicht mehr nötig sind. Dann wird der entsprechende Satellit einfach wieder entfernt.

Nachdem die historischen Features zu Verfügung stehen, kann das Vorhersagemodell unter Verwendung des Sandbox Prototyping getestet und optimiert werden („Process & Validate Model"). Der Optimierungsprozess unterscheidet sich dabei abhängig von der verwendeten Vorhersagemethode. Eine lineare Regression kann beispielsweise durch das Verwenden von Polynomen der Features oder das Wechseln auf eine robuste lineare Regression optimiert werden. Für die Optimierung eines Modells ist stets eine Analyse der Gründe einer schlechten Vorhersage notwendig.

Auch die Vorhersage von Fußball-Ergebnissen mithilfe der Poisson-Verteilung besitzt einige Nachteile, die ausgeglichen werden müssen. Im Vergleich zu den reellen Torverteilungen ist die vorhergesagte Wahrscheinlichkeit für null Tore zu gering. Dies kann durch das Nutzen einer Zero-Inflated-Poisson-Verteilung verbessert werden. Zudem ist die Wahrscheinlichkeit für Unentschieden zu niedrig. Werden jedoch die historischen Unentschieden-Statistiken als Korrekturfaktor für das Modell genutzt, kann auch dieser Nachteil ausgeglichen werden.

Abbildung 9 zeigt die Simulation des optimierten Vorhersagemodells gegen die Wettquoten des Buchmachers Bet365. Auf Basis der vorhergesagten Wahrscheinlichkeiten werden unterbewertete Wetten identifiziert und auf diese gesetzt. Das Modell liefert bei einer Anzahl von mehr als 1.500 Wetten mit einem Einsatz von 10 Euro pro Wette einen Profit von 1.120 Euro (Y-Achse) über den simulierten Zeitraum (X-Achse). Das entspricht einer Verzinsung von sieben Prozent des eingesetzten Kapitals.

Hat sich der Prototyp eines Vorhersagemodells durch Tests und Simulationen als erfolgreich herausgestellt, kann das Modell in den Standard-Verarbeitungsprozess integriert werden („Implement & Maintain Model"). An dieser Stelle wird die R-Integration der verschiedenen Datenbanken genutzt. Das Vorhersagemodell oder das trainierte Modell wird ähnlich wie die berechneten Features als zusätzlicher Kontext („Satellit") in das Datenmodell integriert.

Die Flexibilität der Data-Vault-Modellierung ist auch hier ein Vorteil. Es können verschiedene Vorhersagemodelle auf einfache Art und Weise nebeneinander implementiert werden. Im Information Layer lassen sich danach Berichte erstellen, um die Ergebnisse verschiedener Vorhersagemodelle zu kombinieren und für Auswertungen zur Verfügung zu stellen.

Abbildung 10 zeigt das erweiterte Datenmodell mit der implementierten Poisson-Vorhersage. Das Vorhersagemodell muss nur für die aktuellen Spiele berechnet werden. Daher sind die Feature-Berechnungen und die Poisson-Vorhersage als zusätzliche Satelliten für die aktuellen Spiele implementiert.

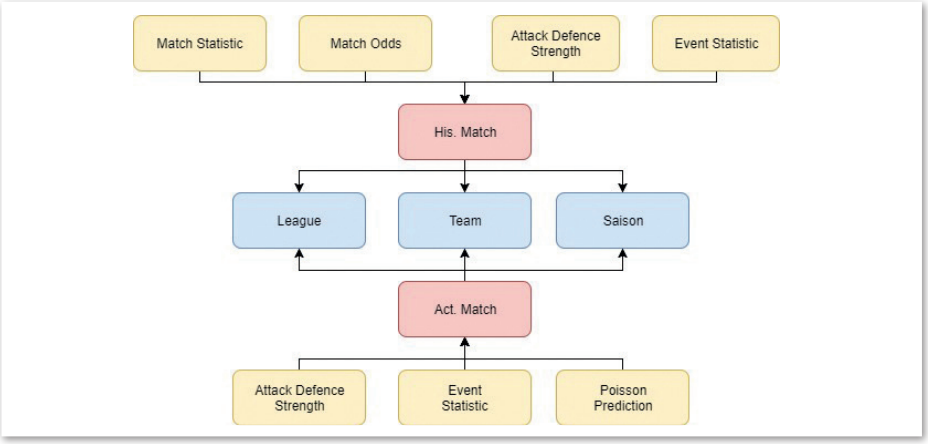


Abbildung 10: Analytical-Layer-Data-Vault-Modell mit implementierter Vorhersage

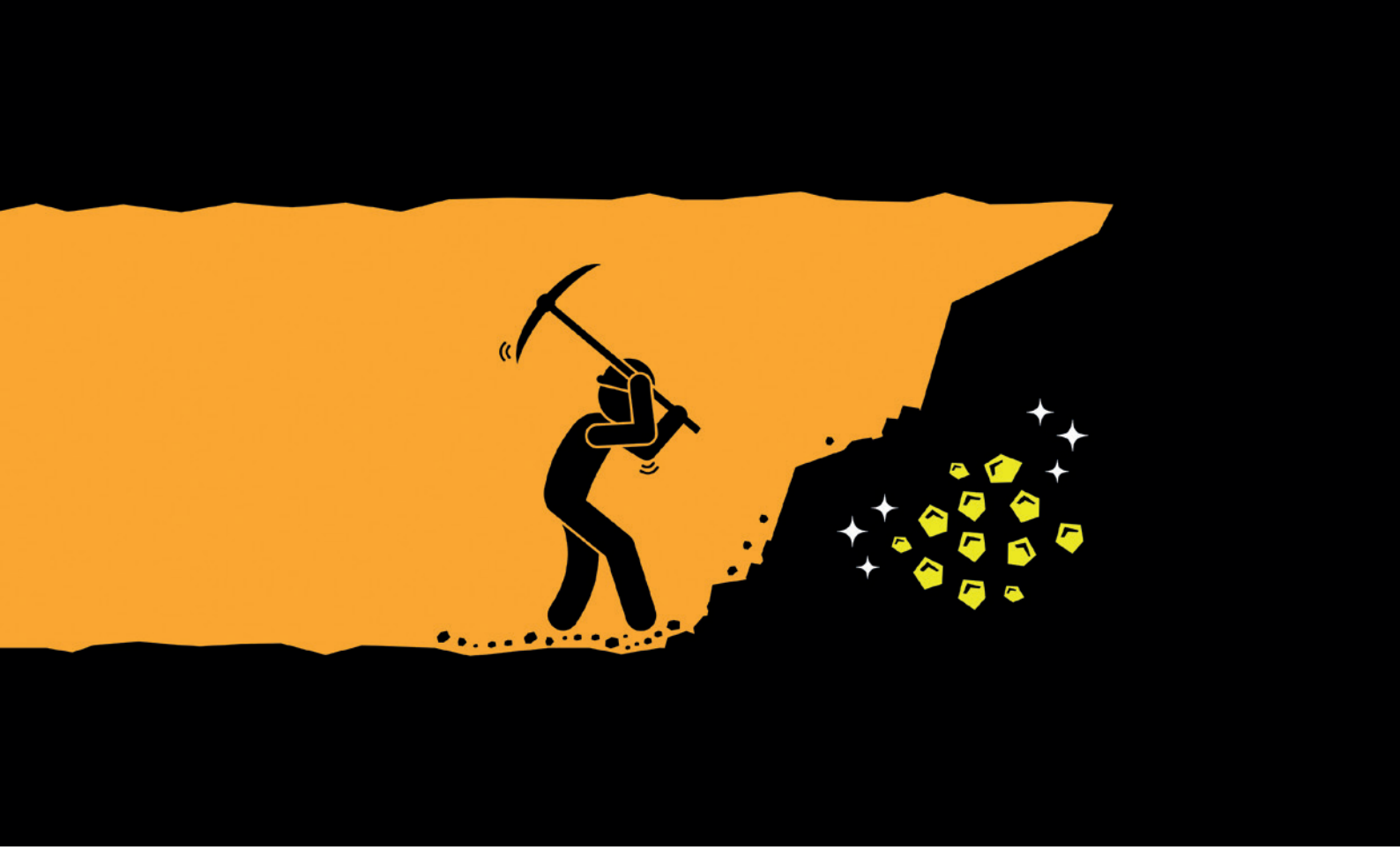
Erkenntnisse

Es hat sich gezeigt, dass bereits ein sehr einfaches Vorhersagemodell reicht, um gegenüber dem Buchmacher im Vorteil zu sein. Während der Entwicklung und Implementierung des Vorhersagemodells für Fußball-Ergebnisse hat das Architektur-Konzept „TripleA DWH" klare Vorteile bei der Flexibilität und Erweiterbarkeit des Gesamtsystems gezeigt. Data Vault 2.0 bietet damit nicht nur Vorteile für Standard-DWH-Implementierungen, sondern auch für den Aufbau eines Predictive Systems. Eine Integration mehrerer Vorhersagemodelle ist ohne Problem möglich. Die Möglichkeit, R Code direkt auf den Daten auszuführen, führt dazu, dass die Gesamt-Architektur eines Predictive Systems weit weniger komplex ist als bei der Berechnung auf separaten Servern.

Quellen

- Data Vault 2.0: *http://danlinstedt.com/#*
- „Seven Steps to Effective Predictive Modeling": *http://oliviagroup.com/training/predictive-modeling-training*
- Dixon & Coles, „Modelling Associated Football Scores and Inefficiencies in the Football Betting Market": *http://www.math.ku.dk/~rolf/teaching/thesis/DixonColes.pdf*

Andre Dörr
andre.doerr@trevisto.de



Process Mining für jedermann

Stephan La Rocca, PITSS GmbH

Process Mining versteht sich in der Evolution des Business-Intelligence-Ökosystems als legitimer Nachfolger von „Data Mining" und „Predictive Analysis", führt aber noch ein deutliches Schattendasein. Dabei steckt enormes Potenzial in dem Ansatz, die Prozesse, die sich hinter der Änderung von unternehmensrelevanten Daten verbergen, genauer betrachten, analysieren und vergleichen zu können. Es ist eine Art Schulterschluss von IT und Business-Abteilung, der im Zeitalter der Digitalisierung notwendig ist.

Warum eigentlich Process Mining? Sollte sich ein Unternehmen nicht bewusst sein, welche Prozesse systemtechnisch unterstützt werden und welche nicht? In den meisten Fällen sicherlich, aber ist man wirklich sicher? Lassen sich Ad-hoc-Fragen beantworten wie „Ist meine Bestellung in 85 Prozent aller Fälle innerhalb von zwei Tagen erfüllt?", „Was sind die Gründe bei den Ausnahmen?", „Gibt es andere Möglichkeiten, um den Auftrag zu erfüllen und werden diese verwendet?", „Wenn ja, wie oft?" und „Sind alle Alternativen im System abgebildet?"

Process Mining hilft, solche und ähnliche Fragestellungen zu beantworten, kontinuierlich zu monitoren und deren Antworten zu verbessern. In größeren Unternehmen kommt dann recht schnell die Fragestellung hinzu, ob Prozesse standortübergreifend harmonisiert sind. Auch hier folgt, falls dies nicht der Fall ist, die Frage nach dem Warum.

Die ersten Erfahrungen zeigen darüber hinaus sehr interessante weitere Einsatzgebiete. Als Software-Hersteller möchte man beispielsweise wissen, wie die unterschiedlichen Kunden die Software einsetzen, um dadurch auf vermeintliche Schwachstellen, Schulungsbedarfe oder Potenzial für das nächste Release rückschließen zu können. Als Inhouse-Lösungsanbieter eruiert man hingegen vielleicht Standard-Lösungen oder plant eine Investition in eine neue Software-Entwicklung – hier drängt sich eine Betrachtung auf Prozess-Ebene nahezu auf. Sind alle bestehenden Prozesse in der neuen Welt abgedeckt? Was ist die Grundlage für eine Gap-Analyse oder die Requirement-Definition? Diese Fragen sollten mit den bestehenden und verwendeten Systemen beantwortet werden, nicht mit den alten Pflichtenheften in Papierform, die den Staub der Jahre tragen.

Ein letztes Beispiel für Process Mining ist die nicht gerade behagliche Situation eines Audits. Um wie viel einfacher würden solche Prüfungssituationen ablaufen, wenn man mit Process Mining aufzeigen könnte, dass alle Prozesse im Unternehmen in 99 Prozent

aller Fälle die vorgeschriebenen Wege aus den Qualitätsnormen durchlaufen. In den Process-Mining-Typen bezeichnet man diesen Vergleich zwischen einem vorhandenen Modell und dem tatsächlichen Ablauf der Prozesse als „Conformance". Das wäre die Aufgabe eines Mausklicks.

Ein weiterer, aktuell weitverbreiteter Typ ist „Discovery", das dazu dient, aus den Daten die dahinterliegenden Prozesse zu identifizieren. Darüber hinaus wird vereinzelt nach der Analyse und dem Vergleich auch auf den Typ „Enhancement" zurückgegriffen, der dann die identifizierten Verbesserungen modelliert und zurückführen soll.

Die Grenzen im Process Mining

Anwendungen sind auf die Bedürfnisse des Benutzers, den Anwendungsfall und die Unterstützung der Geschäftsprozesse hin implementiert. Sie sind nicht darauf ausgelegt, sinnvolle Daten mit sehr hoher Qualität für ein Process Mining bereitzustellen, damit in diesem Tool Analysen und Visualisierungen durchgeführt werden können.

Um an Rohdaten für das Process Mining zu kommen, haben sich drei verschiedene Wege etabliert. Der einfachste Ansatz besteht darin, bestehende Logfiles zu konsumieren, um mit Audit-Daten oder Workflow-Tracking-Informationen den zugrunde liegenden Prozess aus diesen Audit-Daten zu rekonstruieren. Gerade Workflow-basierte Systeme, etwa auf Basis von BPEL, sind perfekt dafür geeignet. Die Qualität der Ergebnisse ist direkt mit der Existenz und der Datenqualität dieser Audit-Protokolle verbunden. Leider sind nicht alle Legacy-Anwendungen in der Lage, diese Daten zu liefern. Sogar Datenbank-zentrierte Anwendungen liefern in der Regel keine Audit-Daten auf Transaktionsebene in dieser notwendigen Art der Granularität.

In diesem Fall greift der zweite Ansatz, bei dem alle Key-User und Prozessleiter in umfangreichen Interviews gebeten werden, den Idealzustand der Prozesse zu beschreiben. Parallel dazu wird jedes Mal das Datenmodell geparst, um die Auswirkung des beschriebenen Prozesses innerhalb des Modells zu erkennen. Analysiert man zu einem späteren Zeitpunkt das Datenmodell, können anhand der Daten die dahinterliegenden Prozesse rückwirkend angenommen werden.

Dieser Ansatz ist sehr zeitaufwändig und mit Missverständnissen gepflastert. Daten könnten durch nicht analysierte Ereignisse anderweitig geändert worden sein. Zusätzlich stellen die Interviews keine Vollständigkeit der relevanten Prozesse sicher, ganz zu schweigen von der Tatsache, dass vielfach Prozesse keine signifikanten, eineindeutigen Änderungen im Datenmodell hervorheben. Es ist also mehr ein Process Guessing als ein Process Mining.

Dann hilft vielleicht der dritte Ansatz, der ein bisschen die „Beweislast" umdreht. Selbst noch recht jung in der ohnehin schon neuen Disziplin „Process Mining" ist der Versuch, ähnlich wie beim Data Mining mit noch weiter entwickelten Algorithmen aus dem Bereich der künstlichen Intelligenz Prozessmuster in den verschiedenen Pools von Daten zu identifizieren. Dafür benötigen diese Algorithmen eine tiefe Analyse und Lernphase, bevor sie mit Process Mining auf die Rohdaten losgelassen werden können. Es ist ein sehr innovativer, aber auch sehr zeitaufwändiger und kostenintensiver Ansatz.

Alle Methoden haben als Zwischenziel, dass die zu analysierenden Systeme einen möglichst akkuraten Event Stream an das Process-Mining-Tool übergeben. Als Format für einen Event Stream hat eine Arbeitsgruppe der IEEE-Organisation im Jahr 2010 Extensible Event Stream (XES) als Beschreibung übernommen und im letzten Jahr als Standard verabschiedet *(siehe „http://www.xes-standard.org")*. Dieser wird von allen gängigen Tools unterstützt und erlaubt eine freie Erweiterbarkeit auf Basis eigener Extensions, um Spezifika des Unternehmens und/oder Prozesses auch aus der Applikation an das Process-Mining-Tool zu übergeben. Darüber hinaus unterstützen die meisten Tools aber auch den Import via CSV-Dateien oder den direkten Zugriff auf Datenbank-Objekte.

Für die großen Systeme mit konstanten Datenmodellen sind bereits vorkonfigurierte Exportstrecken erstellt, die beispielsweise für einen klassischen Beschaffungsprozess alle notwendigen Event-Stream-Informationen aus der Applikation extrahieren. Hier gibt es für SAP, Microsoft Dynamics sowie Oracle E-Business Suite bereits für eine Reihe von Lösungen. Es existieren jedoch keine simplen Exportstrecken für Jedermanns-Software.

Da in den meisten Fällen die Individual-Lösung nicht zwingend eine Workflow-Engine im Hintergrund benutzt, die mit wenigen Erweiterungen dazu genötigt werden kann, hilfreiche Audit-Daten auszugeben, bleibt vielfach nur der Blick auf die Veränderungen im Datenmodell. Die sind allerdings, wie bereits argumentiert, mit viel Zeitverlust durch Interviews, Research und Vermutungen verbunden.

Eine Chance, dem zu entgehen, liegt darin, die Applikationssoftware mehr in die Verantwortung zu nehmen. Sollte es gelingen, den Anwender bei der Abarbeitung des Prozesses durch die Software zu beobachten und diese Informationen feingranular in einen Event Stream zu schreiben, wäre die Brücke zwischen Prozess und Daten sehr schnell genommen.

Ein Prozess wird als eine Abfolge von Interaktionen des Anwenders mit der Software definiert, die einen Rahmen etwa innerhalb einer Transaktion haben. Alternativ kann der Prozess durch eine Initialaktion eingeleitet werden, zum Beispiel durch einen bestimmten Button in der Applikation, die Wahl eines Menüeintrags etc., und durch eine passende, gleichwertige Aktion beendet werden.

Sollte es nun gelingen, diese Aktionen in den Event Stream mit dem verbundenen, primären Applikationsobjekt (eindeutige Bestellnummer oder Ähnliches) einzutragen, hat jedes Process-Mining-Tool im Anschluss ein leichtes Spiel. Idealerweise wird der Event Stream nicht direkt innerhalb der Applikation abgebildet, vielmehr werden die notwendigen Daten zunächst über die üblichen Log-Mechanismen weitergegeben. Das hat den Vorteil, dass das Framework sich bereits um die Aufgaben kümmert, die Dateien zu verwalten, Rotating sicherzustellen und integrative Methoden zum Schreiben dieser Informationen auf unterschiedlichem Log-Level performant zu ermöglichen. So ist mit minimalem Footprint eine Übergabe möglich. Ein Verändern des Log-Levels erlaubt darüber hinaus, ein Process Mining gezielt ein- oder auszuschalten.

Betrachten wir in der Welt von Oracle etwa die Applikationen auf dem WebLogic-Server, so bietet dieser mit dem Oracle-Diagnostic-Logging bereits umfangreiche Möglichkeiten, Informationen aus unterschiedlichen Applikationen über die Administrationskonsole verwalten zu können. Spezielle Logger-Klassen wie in ADF können diese Möglichkeiten mit einem konfigurierten ODL-Handler nutzen. Mit Oracle Forms 12c gelingt es gleichfalls, dass Forms-Applikationen mit dem bekannten Message-Built-In direkt Informationen an das ODL schicken können.

Etwas umfangreicher wird es bei JavaScript-basierten SPA, die etwa mit dem JET-Toolkit erstellt wurden. In der Regel reicht es aus, die Prozesse beginnend mit dem Zugriff auf den Service-Layer der Ap-

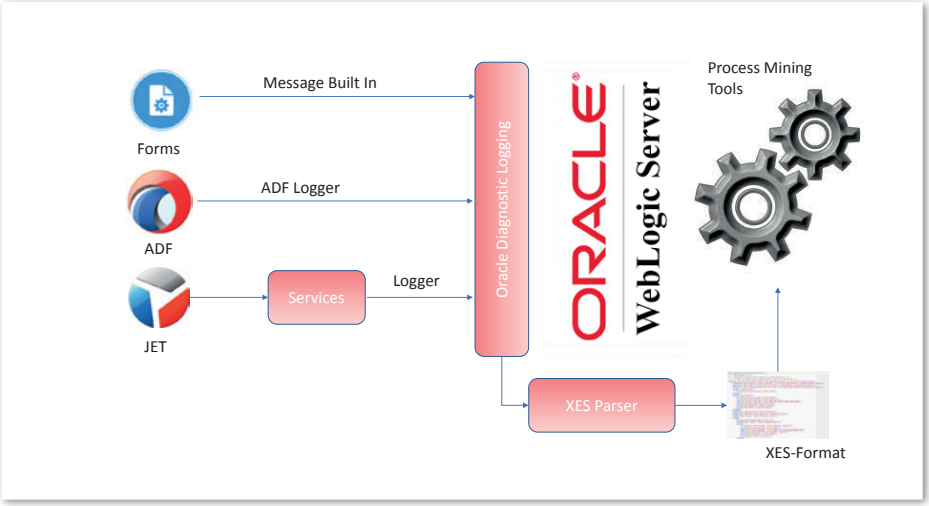


Abbildung 1: Event-Stream-Generierung in bestehenden Applikationen

plikation zu identifizieren. Der JavaScript-Part auf dem Client ist primär für die UI-Anpassungen verantwortlich, die für einen Prozess vernachlässigt werden können. Mit JavaScript auf dem Server beziehungsweise mit WebServices kann dann in gewohnter Weise wieder in das Logfile des Application-Servers geschrieben werden. Hier ist das primäre Objekt des Prozesses bekannt, allerdings besteht die Gefahr, dass – je nach Implementierung – durch die lose Kopplung von User Session und Services der Kontext für die Zusammengehörigkeit von einzelnen Prozess-Schritten verloren geht. Hier sind gegebenenfalls Anpassungen an der Software notwendig, die man aufgrund des Low-Footprint-Ansatzes auf jeden Fall vermeiden will *(siehe Abbildung 1)*.

Die Nutzung von Logging-Funktionalitäten flächendeckend in einer Applikation ist durchweg von dem eingesetzten Framework abgedeckt und hinterlässt keine proprietären Spuren fremder Hersteller. Soll das Event-Stream-Logging nachträglich in eine Applikation eingefügt werden, greifen Vererbungsmechanismen, Parser und Codegeneratoren.

Basiert das System auf Java, lassen sich mit klassischen Methoden der Vererbung in den Basisklassen Methoden einfügen, die die Event-Stream-Informationen ausschreiben. In Oracle ADF bieten sich beispielsweise die Framework-Extension-Classes dazu an.

Für JavaScript gibt es reichlich Code-Generatoren, die immer wieder und sogar kontextbezogen Code-Fragmente erstellen und einbinden können. Basieren dabei die Backend-Services auf Java-Web-Services, ist ein Java-Logger ohnehin verfügbar. Selbst für Oracle-Forms-basierte Applikationen können für jede PL/SQL-Unit nachträglich passende Message-Built-In-Aufrufe mit den relevanten Informationen eingetragen werden. Sollte man bereits Objektklassen in Oracle-Forms nutzen, lassen sich hier leicht Trigger-Vererbungen einsetzen. Alternativ kann man mit dem Forms-API-Master selbst strukturiert Message-Built-Ins dem PL/SQL-Code hinzufügen. Die Firma PITSS GmbH bietet darüber hinaus weitere Möglichkeiten an, diese Informationen auch strukturiert und kontextabhängig in die Forms- und Reports- und sogar Datenbank-Anwendung einzufügen.

Die Nutzung der systematischen Logging-Funktionalität erhöht automatisch die Qualität der Aufzeichnung gegenüber einem Mining auf dem Datenmodell, da sie zweifelsfrei die Aktion des Benutzers mit dem Prozess und den zugehörigen Daten verknüpft. Ein Reverse-Engineering, um von den Daten an die Ursachen zu kommen, ist nicht mehr notwendig. Ganz nebenbei lässt sich dabei auch ein wertvoller zusätzlicher Effekt erzielen, der den drei ursprünglichen Methoden verborgen bleibt: Das Fehlen von Einträgen im Event Stream deutet darauf hin, dass diese Prozess-Schritte, obwohl implementiert und vorhanden, nie benutzt wurden. Aufgrund von Unwissenheit, Fehlprogrammierung oder saisonalen Tätigkeiten? Nun ist man in der Lage, mit diesen Fragen zurück in die Fachabteilung zu gehen und über diesen Weg die Prozesse weiter zu verbessern.

Diese Anbieter bestimmen den Markt

Im Juni 2017 hat die BPM&O GmbH einen Markt-Monitor zum Thema „Process Mining" veröffentlicht und die Tools der Anbieter Celonis SE, Lana Labs GmbH, ProcessGold AG, QPR (Protema), Signavio, SNP und der Software AG bewertet. Die Übersicht ist nach einer kostenfreien Registrierung auf der Seite *„https://www.kurze-prozesse.de/2017/06/26/der-process-mining-markt-ist-stark-in-bewegung"* verfügbar.

Im Oktober 2016 bereits berichtete die Computerwoche darüber, dass Process-Mining-Tools stark im Kommen sind *(siehe „https://www.computerwoche.de/a/process-mining-richtig-einsetzen,3325694")*. Zitat:„Mit Process Mining gewinnt eine innovative Prozess-Analyse-Methode immer mehr an Bedeutung. Ihr Einsatz kann zu einer deutlichen Leistungssteigerung operativer Prozesse führen."

Erfreulich für den Markt hierzulande ist, dass vor allem deutsche Unternehmen, teilweise auch mit internationalen Investoren, den Markt bereits breit bedienen können. Parallel dazu sind auch gerade deutsche Hochschulen federführend in der Forschung (siehe dazu das Interview mit Tobias Rother von der Process Analytics Factory am Ende des Artikels).

Neben den kommerziellen Lösungen tummelt sich mit ProM von der Technischen Universität Eindhoven auch eine Open-Source-Lösung auf dem Markt *(siehe „http://www.processmining.org/prom/start")*. Auf der Webseite werden die Zusammenhänge im Process-Mining-Umfeld erklärt und mit ein paar Beispielen kann jeder schnell die ersten eigenen Schritte gehen. Für eine längere Reise in die Welt der Prozess-Analysen wird dann aber sicherlich ein professioneller Reisebegleiter notwendig.

Fazit

Der Artikel zeigt, dass es nicht länger den großen Applikationen vorbehalten ist, die neuesten Ergebnisse der Forschung für Process Mining nutzen zu können. Dazu sollte man einen Blick auf seine Applikation werfen und die strukturierte Ausgabe prozessrelevanter Informationen planen.

Stephan La Rocca
slarocca@pitss.com

Tobias Rother
Gründer und Geschäftsführer der Process Analytics Factory und einer der Pioniere im Process-Mining-Markt

Wie schätzen Sie den Markt für Process Mining in Deutschland ein?

Der Markt boomt derzeit. Das hat zunächst einmal technologische Gründe. Seit dem Jahr 2008 und der Verfügbarkeit der ersten kommerziell einsetzbaren Tools hat sich auf Anbieterseite sehr viel getan. Mit der ersten Generation von Process-Mining-Tools war es möglich, die Ist-Prozesse zu visualisieren und zu analysieren. Mittlerweile hat sich eine neue Generation von Tools im Markt etabliert, bei denen – rein technisch betrachtet – eine Fusion von BI- und Process-Mining-Software stattgefunden hat. Das hat gerade für Analysten im Unternehmen große Vorteile: Neben beeindruckenden Prozess-Visualisierungen werden kontextbezogen auch Key-Performance-Indikatoren sowie Metriken automatisch berechnet. Der Anwender nutzt eine intuitive Benutzeroberfläche mit einer Vielzahl von Filtermöglichkeiten für interaktive, grafische Auswertungen. Darüber hinaus hat das Thema insgesamt deutlich an Sichtbarkeit gewonnen. An deutschen Hochschulen wird am Thema „Process Mining" intensiv geforscht. Die meisten der im Markt angebotenen Tools werden in Deutschland entwickelt. Große Beratungshäuser haben begonnen, Process Mining als digitale Beratungskomponente bei ihren Kunden einzusetzen. BI- und Big-Data-Lösungsanbieter erweitern ihr Portfolio auf die Ebene der Geschäftsprozesse und nutzen Process-Mining-Verfahren im Rahmen von Digitalisierungsinitiativen bei ihren Kunden. Diese Entwicklung der vergangenen Jahre hat unter anderem auch dazu geführt, dass Anwender mittlerweile unter einigen sehr guten Process-Mining-Tools das für sie passende auswählen können.

Was sind die größten Herausforderungen, denen sich Process Mining stellen muss?

Die größte Herausforderung liegt noch immer in der Daten-Transformation, also der Vorverarbeitung von Daten für das Process Mining. Es ist wichtig, dass man versteht, dass die intelligente Vorverarbeitung von Daten über den Erfolg der Einführung von Process-Mining-Methoden im Unternehmen entscheidet. Dieses Problem löst man am besten, wenn man mit einem Lösungspartner zusammenarbeitet, der weiß, wie Zeitstempel und komplexe verknüpfte Zusammenhänge der Tabellen und Felder in Datenbanken für das Process Mining korrekt zu interpretieren sind. Nur so lassen sich Zeit und Kosten für die manuelle Identifikation von Daten in Datenbeständen reduzieren sowie Aufwendungen bei der Verknüpfung von Daten aus unterschiedlichen Datenquellen senken.

Wo erzielt Process Mining den größten Mehrwert?

Wir sprechen im Process-Mining-Kontext alle gerne schnell von „Enterprise", „Big Data" oder „Real-Time". Den größten Mehrwert erzielt Process Mining jedoch eindeutig als Tool-basierte Methode im Rahmen konkreter Projekte. Weil Process Mining eine hoch attraktive Digitaltechnik für die Analyse von Geschäftsprozessen ist, bietet sie sich als unterstützende Methode in Transformations- und Change-Projekten genauso an wie für Optimierungsprojekte etwa im Kontext von Robotic Process Automation (RPA) oder IoT/Industrie 4.0. Besonders wichtig erachten wir den Einsatz von Process Mining zudem im Rahmen der Migration von Systemen und Datenbanken. Darüber hinaus kann man aus Process-Mining-Verfahren sehr großen Nutzen ziehen, wenn man Standard-Geschäftsprozesse wie Purchase-to-Pay oder Order-to-Cash analysieren möchte. Erste Hersteller bieten für diese End-to-End-Prozesse vorkonfigurierte Content-Pakete beziehungsweise Schnelltests an. Fachabteilungen können sich beispielsweise mit einem Order-to-Cash-Schnelltest einen faktenbasierten Überblick über die Effizienz in der Abwicklung von Kundenaufträgen verschaffen und Optimierungspotenziale innerhalb von weniger als 24 Stunden erkennen.

Welche Trends wird es in den nächsten Jahren geben?

Prof. Dr. Max Mühlhäuser, TU Darmstadt, hat anlässlich der Process Mining Online Konferenz PMOK17 gesagt: „Ziel von Process Mining muss es sein, ein intelligentes System bereitzustellen, das auf Basis der Historie autonom agiert". Das wird uns nur gelingen, wenn wir bereits integrierte BI- und Process-Mining-Verfahren mit künstlicher Intelligenz (Machine Learning/Deep Learning) kombinieren, um beispielsweise Prozess-Strukturen unbekannter operativer Daten besser zu verstehen und nachfolgend zu verbessern. Wir werden so in der Lage sein, Unternehmen dabei zu helfen, Produktivitätssteigerungen ganzer Wertschöpfungsketten noch sehr viel effektiver zu erzielen.

Wie verschaffe ich mir einen Überblick über meine Daten?

Dr. Nadine Schöne, ORACLE Deutschland B.V. & Co. KG

Um Daten zu analysieren, muss man zumindest ansatzweise verstehen, wie diese strukturiert sind und wie es um deren Qualität und Konsistenz bestellt ist. Daraus ergeben sich dann mögliche und nötige Aufbereitungsprozesse. Um einen Überblick über die Verteilung der Daten zu bekommen, kann man dann unterschiedliche Maßzahlen berechnen. Dieser Artikel liefert eine knappe Darstellung der nötigen Schritte für die Beurteilung, Bereinigung und Auswertung der Daten vor einer statistischen Analyse.

Vorab ein Hinweis: Es gibt eine Vielzahl von Tools und Algorithmen, die einem dabei helfen, einen ersten Überblick über einen Datensatz zu bekommen. Wir beschränken uns hier auf R (Open Source) und die kommerzielle Lösung Oracle Data Visualization. Zudem geht es in diesem Artikel darum, dass es bereits ein Datenschema gibt und dass die Daten in Tabellenform vorliegen.

Der Klassiker: Mittelwert und Standardabweichung

Meist ist die allererste Idee, zuerst den Durchschnitt einer Datenreihe („Mittelwert", „mean", „arithmetisches Mittel") zu berechnen – man zählt alle Werte zusammen und teilt diese Summe durch die Anzahl der Werte. Der Mittelwert liefert einen Eindruck davon, um welches Zentrum sich die Daten verteilen. Er sagt aber leider nichts darüber aus, wie groß die Schwankungsbreite der

	Messwerte								mean	sd
Messreihe 1	1	2	3	4	4	5	6	7	4	2
Messreihe 2	3	3	4	4	4	4	5	5	4	0,76

Tabelle 1: Messreihen mit gleichem Mittelwert, aber unterschiedlicher Streuung

Messwert	1,2	1,3	1,4	1,5	1,7	2,0	2,3	2,4	2,8	2,9
Häufigkeit	1	2	2	2	1	2	2	1	1	1
Klasse	>1,0 – 1,5				>1,5 – 2,0		>2,0 – 2,5		>2,5 – 3,0	
Klassenhäufigkeit	5				3		3		2	

Tabelle 2: Beispiel für Klassenbildung – nach dem Zusammenfassen von Werten in Klassen lässt sich die häufigste Klasse bestimmen

Daten ist. Beispiel: Jemand mit den Füßen beim Ofen und mit dem Kopf im Kühlschrank ist im Durchschnitt wohltemperiert …

Man benötigt also weitere Maßzahlen für die Schwankung oder Streuung der Daten um den Mittelwert. Häufig werden hier die Varianz und die Standardabweichung („standard deviation", „sd") berechnet. Kombiniert man den Mittelwert mit einem Streuungsmaß, bekommt man einen besseren Eindruck von der Verteilung der Werte. Sind die Daten normalverteilt, weiß man sogar, dass gut 68 Prozent der Werte im Abstand von maximal einer Standardabweichung vom Mittelwert entfernt liegen.

Dazu ein Beispiel: *Tabelle 1* enthält zwei Messreihen mit jeweils acht Messwerten. In *Abbildung 1* sind die Messreihen in einem Histogramm dargestellt. Man erkennt sofort, dass die Werte beider Messreihen unterschiedlich verteilt sind. Der Mittelwert („mean") ist jedoch für beide Messreihen identisch. Gibt man als Maßzahl jeweils nur den Mittelwert an, scheint es, als gäbe es keinen Unterschied zwischen den Messreihen. Die Werte streuen jedoch unterschiedlich stark um den Mittelwert, die Standardabweichung („sd") der ersten Messreihe ist fast dreimal so groß wie die der zweiten Messreihe.

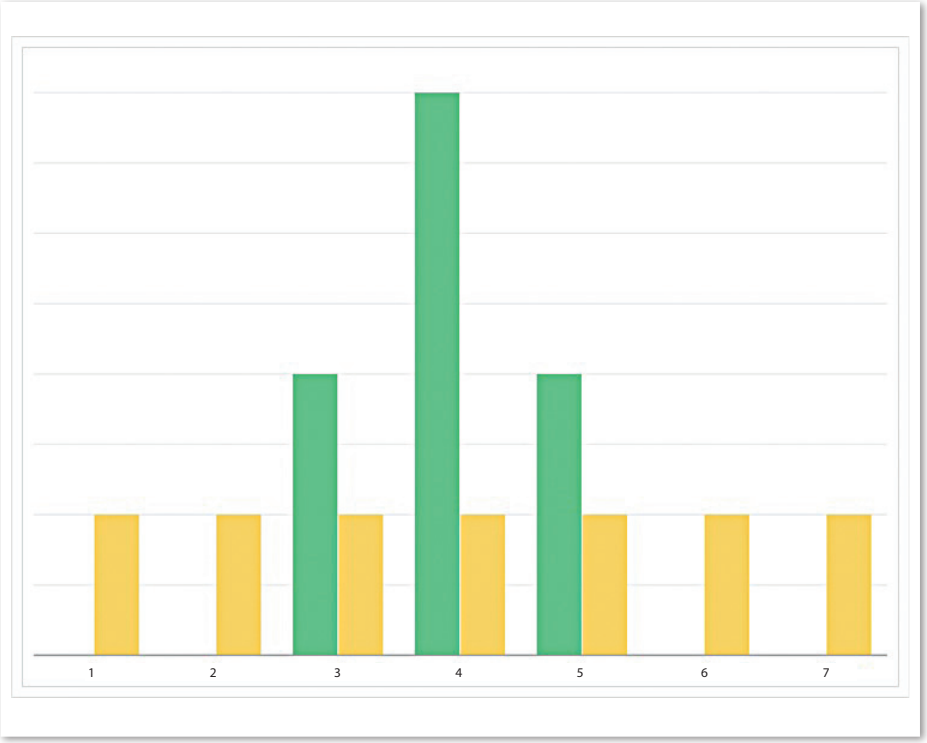


Abbildung 1: Grafische Darstellung der beiden Messreihen aus Tabelle 1 als Balkendiagramm – man erkennt auf einen Blick, dass sich die Verteilung der Messwerte unterscheidet

Der Modalwert – eine Maßzahl für alle Fälle

Was ist, wenn das Datenformat gar nicht zulässt, dass man den Mittelwert und die Streuung berechnet (etwa bei Kundennamen), oder wenn ein Mittelwert keinen Sinn ergibt (etwa bei Inventarnummern)? Welche Maßzahlen lassen sich dann berechnen?

Man kann durchzählen, welche Ausprägung wie häufig in den Daten vorkommt. Damit lassen sich „Schwerpunkte" in den Daten bestimmen. Die häufigste Ausprägung wird als Modalwert („dichtester Wert", „Modus", „mode") bezeichnet. Der Modalwert ist die einzige Maßzahl, die sich immer und für alle Daten berechnen lässt.

Wenn es im Datensatz kaum identische Ausprägungen gibt, sodass es keinen eindeutigen häufigsten Wert gibt, kann man die Daten in Klassen („Gruppen") zusammenfassen, bevor man den Modalwert berechnet. Auch dazu ein Beispiel: Die in *Tabelle 2* gelisteten Messwerte kommen alle nur ein oder zwei Mal vor, es gibt also keinen häufigsten Wert. Fasst man die Messwerte aber in Klassen mit Spannweite „0,5" zusammen, ist der Modalwert die Klasse „>1,0 – 1,5" mit einer Häufigkeit von „5".

Ordnung in den Daten

Wenn man die Daten auch ordnen kann („A ist größer als B", „C ist gleich D"), dann lassen sich unter anderem Minimum („min") und Maximum („max") bestimmen. Beispiele sind Bestellsummen oder ein Bewertungssystem mit den festen Werten „schlecht", „ok" und „gut".

Teilt man die geordneten Daten in Abschnitte mit gleicher Anzahl von Werten auf, nennt man die Grenzen dieser Intervalle „α-Quantile". Der Wert, der genau in der Mitte des Datensatzes liegt, der also die Daten in zwei gleich große Gruppen teilt, ist der Median (das 0,5-Quantil).

Oft viertelt man das Beobachtungsintervall. Die drei berechneten α-Quantile nennt man Quartile. Das erste Quartil ist größer oder gleich 25 Prozent der Werte im Datensatz. Das zweite Quartil ist gleich dem Median, teilt also den Datensatz in der Mitte. Das dritte Quartil ist größer oder gleich 75 Prozent der Werte im Datensatz.

Dazu ein Beispiel: Angenommen, die Messwerte sind 1, 5, 8, 2, 3, 3, 7 und 8. Sortiert man die Werte, erhält man 1, 2, 3, 3, 5, 7, 8 und 8. Das Minimum ist also 1, das Maximum ist 8.

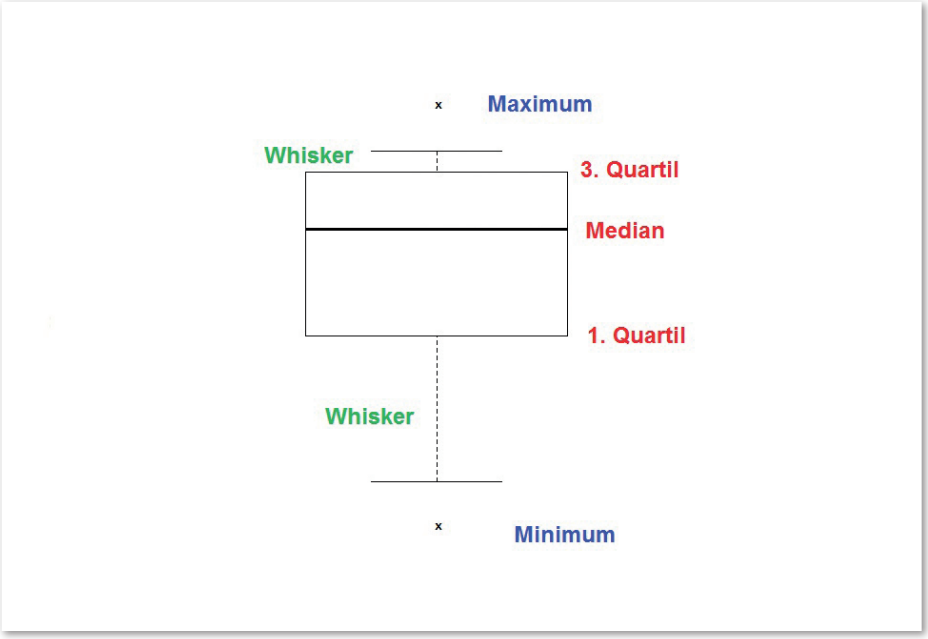


Abbildung 2: Grafische Darstellung der Quartile, Minimum und Maximum in einem Boxplot

```
> mean(meinedaten)
[1] NA
> mean(meinedaten, na.rm=TRUE)
[1] 311255.4
```

Listing 1

```
> summary(meinedaten$AssetCost)
   AssetCost
 Min.   :   7500
 1st Qu.:  41500
 Median :  89000
 Mean   : 311255
 3rd Qu.: 269000
 Max.   :1500000
 NA's   :5
```

Listing 2

Der Wert 3 teilt den Datensatz in zwei gleich große Teile (jeweils vier Messwerte). Damit ist 3 der Median. Die Hälfte der Werte ist kleiner oder gleich 3, die andere Hälfte ist größer als 3. Das erste Quartil ist gleich 2, da 25 Prozent der Werte (zwei Messwerte) kleiner oder gleich 2 sind. Das dritte Quartil ist 7, da 75 Prozent der Werte (sechs Messwerte) kleiner oder gleich 7 sind.

Teilt man das Beobachtungsintervall in 100 Teile, also in Prozente, erhält man die Perzentile. In der medizinischen Statistik werden in der Pädiatrie Perzentile von Körpergröße und Gewicht verwendet, um die Entwicklung von Kindern im Vergleich zu ihren Altersgenossen beurteilen zu können. Wenn also 70 Prozent der Bevölkerung kleiner oder genauso groß sind wie ich, bin ich auf der 70. Perzentile.

Abstände und Summen

Meist lassen sich auch Abstände zwischen Datenpunkten sowie Summen berechnen. Solche Daten nennt man „metrische Daten". Beispiele sind Temperatur, Bestellsumme oder Gewicht. Damit kann man ganz normal rechnen. Viele statistische Tests lassen sich nur auf metrische Daten anwenden.

Eine Maßzahl, die sich nur für metrische Daten bestimmen lässt, ist neben Mittelwert und Streuung auch die Spannweite als Differenz zwischen Minimum und Maximum eines Datensatzes. Zusammen mit Mittelwert und Standardabweichung liefert die Spannweite einen Eindruck von der Verteilung der Daten.

Grafische Darstellung der Quartile im Boxplot

Quartile lassen sich grafisch in einem Boxplot darstellen *(siehe Abbildung 2)*. Die Box stellt die Daten zwischen dem ersten und dritten Quartil dar und zeigt damit, wie die mittleren 50 Prozent der Daten verteilt sind. Der Median ist als Band in der Box eingezeichnet und ist der Wert, der den Datensatz in zwei gleich große Teile teilt.

Die Länge der „Whiskers" ist nicht eindeutig definiert – je nach Tool reichen sie bis zum Minimum und Maximum, oder Ihre Länge wird abhängig von der Spannweite berechnet. Im letzteren Fall werden metrische Daten vorausgesetzt.

Generell vereinfacht eine grafische Darstellung das Erfassen der Struktur eines Datensatzes (siehe Beispiel in *Tabelle 1* und *Abbildung 1)*. Deshalb sind Visualisierungstools besonders geeignet, um einen ersten Überblick über einen Datensatz zu bekommen.

Datenqualität und Datenkonsistenz

Bevor man die erste Maßzahl berechnet, sollte man sich allerdings die Frage stellen, ob die Daten inhaltlich zur Aufgabenstellung passen. Wenn man Voraussagen über Kundensegment X treffen will, aber nur Daten zu Kundensegment Y hat, sind diese Daten ohne Zusatzinformationen höchstwahrscheinlich für diese Analyse nicht geeignet. Damit wäre eine Durchführung einer geplanten Analyse dieser Daten für die gegebene Fragestellung eine reine Zeitverschwendung.

Wenn die Daten inhaltlich für die Lösung der Aufgabenstellung geeignet sind, wirft man als Nächstes einen Blick auf die Datenkonsistenz, das heißt auf die Korrektheit der Daten:

- Wo kommen die Daten her, welche Informationen beinhalten sie und wie wurden sie erfasst?
- In welcher Größenordnung liegt ein eventueller Messfehler? Wenn die Messgenauigkeit bei einer Nachkommastelle liegt, ist es nicht sinnvoll, Maßzahlen mit mehr als einer Stelle nach dem Komma anzugeben.
- Liegen eventuell unterschiedliche Schreibweisen oder auch Tippfehler vor, etwa wenn Ortsangaben aus einer Freitexteingabe stammen? Wenn solche Daten nicht bereinigt werden, kann schon die einfachste Maßzahl – der Modalwert – nicht korrekt berechnet werden.
- In welchem Format beziehungsweise als welcher Datentyp liegen die Daten vor? Werden Nachkommastellen mit einem Punkt oder einem Komma abgetrennt? Je nachdem, welches Tool man verwendet, werden eventuell einige Datentypen nicht unterstützt. Das kann unter anderem dazu führen, dass Zeitangaben nicht erkannt und als Text importiert werden. Die Unterstützung kann auch von Fileformat zu Fileformat variieren, so kann es sein, dass in „.csv"-Files andere Datentypen unterstützt werden als in „.xml"-Files.

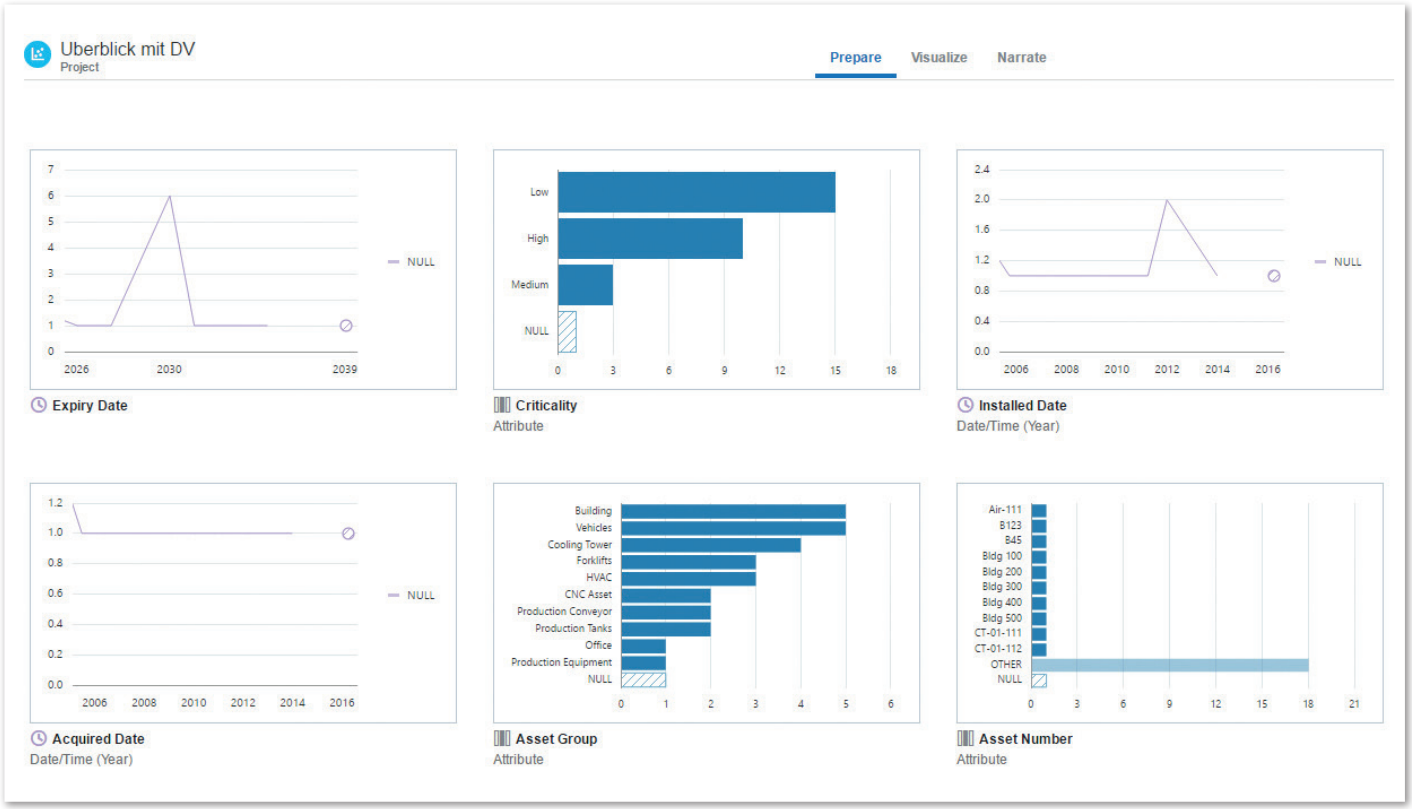


Abbildung 3: Grafischer Überblick inklusive Anzeige von fehlenden Werten in der Datenpräparation („Prepare“) in Oracle Data Visualization

- Gibt es doppelte oder fehlende Werte? Verschiedene Tools und Algorithmen gehen unterschiedlich mit fehlenden oder doppelten Werten um. R gibt beispielsweise bei fehlenden Werten (NAs) im Datensatz für den Mittelwert ein NA zurück. Dies kann man umgehen, indem man NAs explizit von der Berechnung ausschließt *(siehe Listing 1)*. Andere Tools zählen als Default die fehlenden Werte als den Wert 0 und verfälschen somit das Ergebnis.

Beispiele mit Oracle Data Visualization und R

Wie geht man nun konkret vor, um erste Maßzahlen zu berechnen und einen visuellen Überblick zu bekommen? Einen ersten grafischen Überblick bekommt man in der Datenpräparation („Prepare“) in Oracle Data Visualization. Hier sind die Daten aus allen Spalten einzeln grafisch dargestellt – inklusive der Anzahl fehlender Werte *(siehe Abbildung 3)*. Für jedes Datenformat wird automatisch eine passende Darstellung ausgewählt.

In R liefert die Funktion „summary()“ eine Auswahl von Kennzahlen passend zum Datenformat, in diesem Fall Minimum und Maximum, Quartile inklusive Median sowie die Anzahl fehlender Werte (NAs, *siehe Listing 2)*.

Fazit

Die folgenden sieben Punkte liefern ein Grundgerüst für die Beurteilung, Bereinigung und Auswertung der Daten vor einer statistischen Analyse:

1. Passen die Daten inhaltlich zur Aufgabe, kann die im Raum stehende Fragestellung also überhaupt mit den vorliegenden Daten gelöst werden?
2. Bereinigung der Daten von Tipp- und Messfehlern, Überführen in das richtige Datenformat für das gewählte Tool.
3. Gibt es doppelte oder fehlende Werte?
4. Berechnung des Modalwertes, eventuell nach Zusammenfassen der Werte in Klassen.
5. Lassen sich die Daten ordnen? Wenn ja: Berechnung von Minimum und Maximum, des Medians und anderer Quartile (etwa Quantile oder Perzentile)
6. Für metrische Daten: Berechnung von Spannweite, Mittelwert und Standardabweichung.
7. Visualisierung der Maßzahlen, zum Beispiel in einem Boxplot.

Themenverwandte Vorträge auf der DOAG 2017 Konferenz

Auf der DOAG 2017 Konferenz vom 21. bis 24. November 2017 in Nürnberg wird es einen Vortrag zum selben Thema geben, der dann auch noch mehr auf die geeigneten Tools eingeht. Außerdem werden in einem weiteren Vortrag „Einführung in statistische Analysen“ die Auswahl, Durchführung und Interpretation statistischer Tests behandelt.

Dr. Nadine Schöne
nadine.schoene@oracle.com

Daten als Chance

Sigrid Keydana, Trivadis AG

Wenn Daten heute ein Dschungel sind, sind Algorithmen es noch mehr. Dieser Artikel zieht Pfade durch den Algorithmen-Dschungel, die helfen sollen, für individuelle Anforderungen und Bedürfnisse relevante Methoden zu finden.

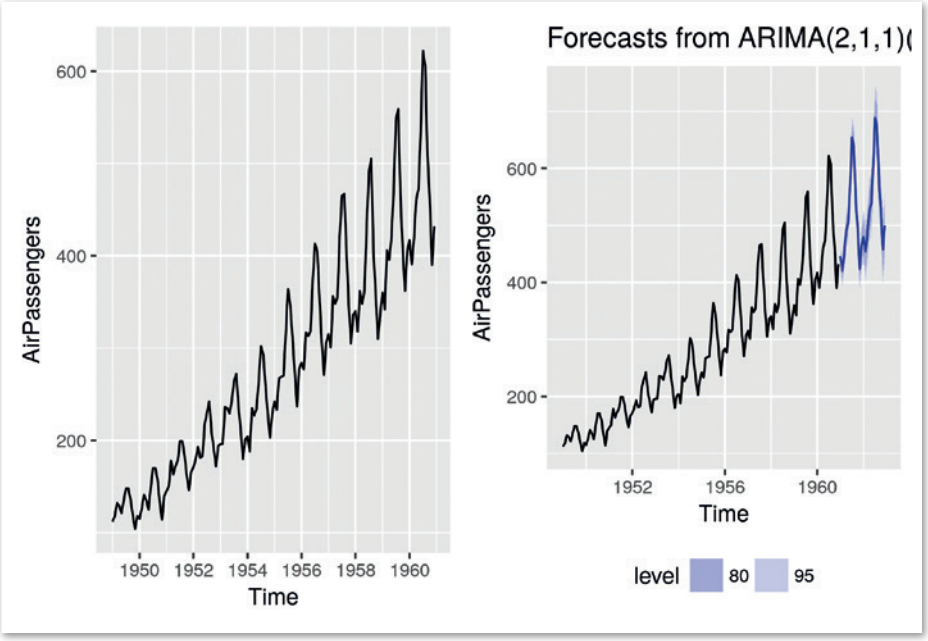


Abbildung 1: Eine univariate Zeitreihe samt Vorhersage mittels ARIMA

Hinweis: Die wissenschaftlichen Grundlagen für diesen Artikel sind als PDF unter *http://www.doag.org/go/1703BusinessNewsKeydana* herunterladbar. Mit diesem nötigen Rüstzeug, um uns im Dschungel zurechtzufinden, können wir uns endlich auf Schatzsuche begeben: Was für Erkenntnisse können wir aus den Daten ziehen, was für Methoden setzen wir ein?

Welche Methoden wir anwenden, hängt von unserer Fragestellung und der Art der Daten ab. Haben wir gewohnte, strukturierte Daten wie Verkaufszahlen, Ausgaben, Kundendaten, Business-Metriken etc., befinden wir uns im Bereich klassischer Methoden wie Regression, Decision Trees oder Cluster-Analyse. Ob man das „Data Science" nennt, „Machine Learning", „Data Mining" oder „Advanced Analytics", ist eine Frage der Vorlieben, der Richtung, aus der man kommt, oder des Umfelds, in dem man arbeitet – die Algorithmen dahinter sind dieselben. Auch die Übergänge zur Statistik sind fließend. Nicht umsonst wird im Literaturteil ein Buch mit dem Titel „Introduction to Statistical Learning" empfohlen. Im Folgenden ist von „Data Science" die Rede.

Wenn das (selbst unter der Hype-Bezeichnung „Data Science") ein klassisches Vorgehen ist, ist das sogenannte „Deep Learning" die Alternative, mit dem in Bereichen wie Bilderkennung, maschineller Übersetzung oder Robotik zurzeit jedes Jahr neue Rekorde gebrochen werden.

Auch wenn Deep Learning prinzipiell auf klassische Daten anwendbar ist, liegen seine Stärken doch vor allem in den genannten Aufgabenstellungen. Deep Learning stellt die gewohnten Denkweisen auf den Kopf: Es geht nicht darum, das zu lösen, was für uns Menschen schwer und zeitraubend ist (zum Beispiel komplizierte Kalkulationen), sondern das, was uns leicht fällt: sprechen, verstehen, sehen, gehen. Traditionell sind es genau diese Dinge, die für Maschinen schwer sind – das ist es, was sich mit Deep Learning nun zunehmend ändert. Bevor wir uns das genauer anschauen, wollen wir sehen, wie man Data Science macht, was damit geht und wobei uns das hilft.

Goodness of fit

Die Diskussion darüber, was wir mit Data Science machen können, steht unter dem Motto „Vorhersagen und Erklären" oder auch „Vorhersagen vs. Erklären". Um Missverständnisse zu vermeiden, müssen wir eine Sache vorab klären: Wie messe ich, wie gut ein Modell ist? Das Allerwichtigste: Immer an neuen Daten; niemals an denen, mit denen ich das Modell angepasst habe. Egal, mit welcher Methode man arbeitet, als Erstes teilt man immer die Daten in zwei, gegebenenfalls drei Sets – ein Training Set, ein Validation Set (optional) und ein Test Set.

Das Training Set wird benutzt, um ein Modell zu fitten – sagen wir, ein Modell zur Bestimmung von Kundenzufriedenheit aus Faktoren wie Produktkategorie, Preiskategorie, Schnelligkeit der Lieferung etc. Während des gesamten Modell-Fittings lasse ich mein Test Set unberührt. Das Test Set dient ausschließlich dazu, die Performance des Modells abschließend zu prüfen.

Viele Modelle aber haben Parameter, die man variieren kann (und sollte!), um die bestmöglichen Einstellungen zu finden. Wie kann man wissen, welches Setting schlussendlich das beste Ergebnis liefern wird, wenn man das Test Set nicht konsultieren soll? Zu diesem Zweck dient möglicherweise das Validation Set. In diesem Fall würde man es benutzen, um die Performance unter verschiedenen Permutationen von Parameter Settings zu testen. Häufiger ist der Einsatz von Kreuzvalidierung („cross validation").

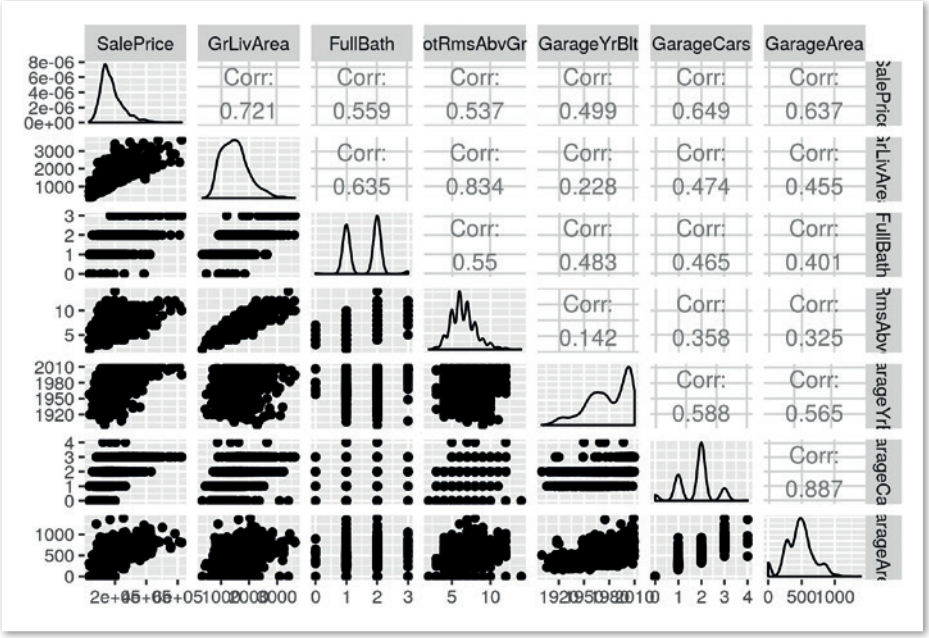


Abbildung 2: Der Verkaufspreis von Anwesen und sechs Einflussvariablen

Dabei benutzt man das Training Set, um die besten Settings zu bestimmen.

Das Vorgehen: Man unterteilt das Training Set in n Subsets (meist 5 oder 10). Nehmen wir 5 als Beispiel. Jetzt iteriert man über die Subsets. Im ersten Schritt nimmt man die Subsets 1 bis 4, legt sie zusammen und trainiert auf dieser Datenbasis die verschiedenen Modelle. (Der Einfachheit halber ist hier von „Modellen" die Rede, es kann sich aber durchaus um ein und denselben Algorithmus handeln, jeweils mit verschiedenen Parameter-Settings.) Die Performance dieser Modelle testet man nun für das Fitten am nicht benutzten Subset 5. Dann wiederholt man den Vorgang mit den Subsets 2 bis 5 fürs Fitten und Subset 1 zum Testen; dann die Subsets 3,4,5,1 fürs Fitten und Subset 2 zum Testen und so weiter, bis jedes Subset einmal zum Testen verwendet wurde. Die Ergebnisse für die verschiedenen Modelle werden gemittelt und man besitzt eine valide Entscheidungsgrundlage für die Modellauswahl. Die Performance auf dem Test Set wird dann separat bestimmt.

Die Trennung in Training Set und Test Set funktioniert im Normalfall einfach als Zufallsauswahl. Anders verhält es sich bei Zeitreihen, also Daten, die intrinsisch zeitlich geordnet sind. Hier würde man jeweils zusammenhängende Zeiträume für das Training Set und für das Test Set wählen.

Auf die Gefahr hin, die Wiederholung zu übertreiben: Die Performance eines Modells wird abschließend immer nur am Test Set evaluiert. Wenn nachfolgend der Kürze halber zu einer Methode nur ein Datensatz gezeigt wird, ist immer das Training Set gemeint, das Set, auf dem wir das Modell fitten.

Vorhersagen vs. Erklären

Abbildung 1 zeigt links ein klassisches Beispiel für sogenannte „Zeitreihendaten" („time series"), die monatlichen Zahlen von Fluggästen. Zeitreihen können auch multivariat sein, also mehrere Variablen umfassen, die man dann aufeinander beziehen kann. Häufig liegen einzelne Zeitreihen vor, um Aussagen über den weiteren Verlauf zu machen. Das Ziel ist ganz klar: Wir möchten eine möglichst korrekte Vorhersage. Damit haben wir ein ganz einfaches Entscheidungskriterium dafür, welche Methode wir wählen sollen: Wir wählen die, die die beste Vorhersage liefert. Für die Bestimmung der besten Vorhersage gilt, wie gerade besprochen: Model Fitting am Training Set und abschließende Beurteilung am Test Set. Im abgebildeten Beispiel kam ARIMA zum Einsatz, die wahrscheinlich meistverwendete Methode zur isolierten Betrachtung einzelner Zeitreihen.

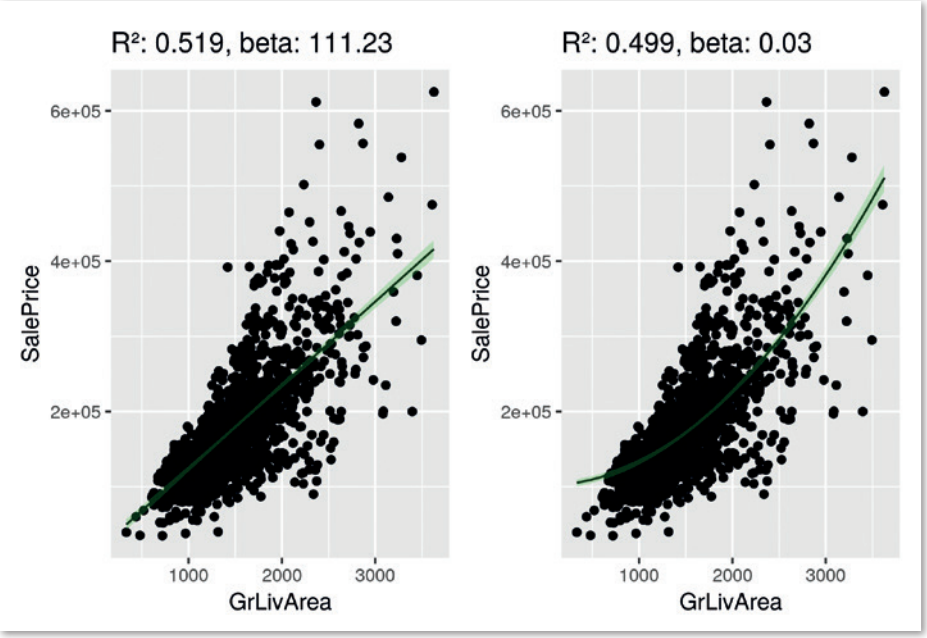


Abbildung 3: Regression des Verkaufspreises auf Wohnfläche, links ist der Prädiktor die Wohnfläche, rechts das Quadrat der Wohnfläche

In *Abbildung 2* geht es darum, den Verkaufspreis eines Anwesens aus Variablen wie Wohnraum, Größe der Garage etc. vorherzusagen [16]. Wie bei den Zeitreihen kann es auch hier einfach das Ziel sein, die bestmögliche Vorhersage zu machen, in diesem Fall, einen optimalen Verkaufspreis, der weder potenzielle Käufer verschreckt noch zu einem Verkauf unter Wert führt.

Das Ziel könnte aber auch sein, etwas über die Welt herauszufinden, in diesem Fall: Was bestimmt eigentlich den Verkaufspreis? Das mag bei der Beschränkung auf Quadratmeterangaben und dergleichen wenig spannend wirken, wird aber schon anders, wenn wir sozioökonomische, geografische und politische Fakten hinzunehmen. Damit hier keine Missverständnisse aufkommen: Natürlich bedeutet der Fokus auf Erklären/Verstehen keinesfalls, dass wir hier notwendig ein wissenschaftliches Interesse verfolgen. Das kann sein, muss aber nicht. Es kann auch einfach darum gehen, durch Abstraktion die Entscheidungsfindung zu erleichtern. Je nachdem, was uns primär interessiert – die Genauigkeit der Vorhersage oder ein erklärendes Modell –, werden wir den Schwerpunkt auf unterschiedliche Methoden legen.

Erklären

Wenn es primär ums Verstehen geht, sind lineare Modelle optimal – vorausgesetzt, die Daten lassen sich mit einem linearen Modell hinreichend gut abbilden. Wenn ein linearer Zusammenhang vorliegt, lässt sich die Schlussfolgerung einfach ziehen: Erhöhung von x um eine Einheit führt zur Erhöhung von y um eine Einheit (siehe *Abbildung 3*, links).

Hier wurde eine einfache lineare Regression des Verkaufspreises auf die Wohnfläche gerechnet. Die Gerade zeigt das gefittete Modell: Mit jedem zusätzlichen Quadratfuß Wohnfläche steigt der Verkaufspreis in der Stichprobe um etwa 111 Dollar („beta" in der Überschrift der Grafik). „R^2" im Titel sagt uns, wie gut das Modell auf diese Daten (Trainingsdaten!) passt: Hier wurden 50 Prozent der Varianz in den Verkaufspreisen durch die Varianz in der Wohnfläche erklärt.

Wenn wir auf *Abbildung 3*, links, schauen, können wir uns fragen, ob hier eine Gerade wirklich am besten die Daten beschreibt. Es sieht aus, als müsste eine Parabel besser passen. Wir können, ohne das lineare Modell zu verlassen, eine Parabel fitten, indem wir die Prädiktor-Variable Wohnfläche quadrieren, wie in *Abbildung 3*, rechts, geschehen. Optisch sieht das besser aus, der „Goodness of fit"-Wert R^2 ist aber tatsächlich niedriger.

In dem Modell in *Abbildung 3*, rechts, steigt der Verkaufspreis mit dem Quadrat der Wohnfläche. Das ist immer noch eine stark abstrahierende Aussage. Wenn wir beide Grafiken vergleichen, können wir auf den Gedanken kommen, dass im unteren Bereich der Wohnflächenwerte der lineare Fit besser

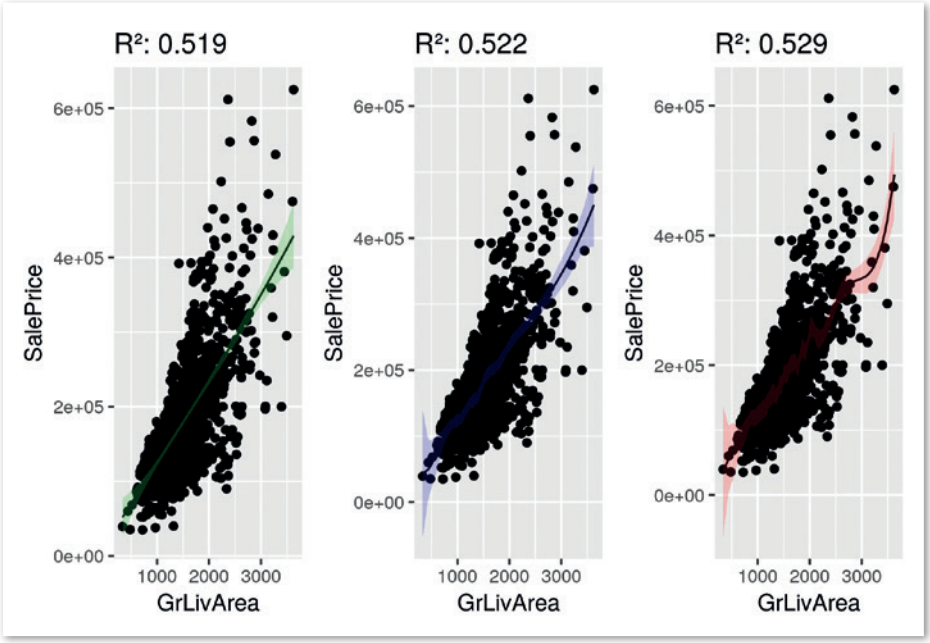


Abbildung 4: Regression Splines mit unterschiedlicher Flexibilität

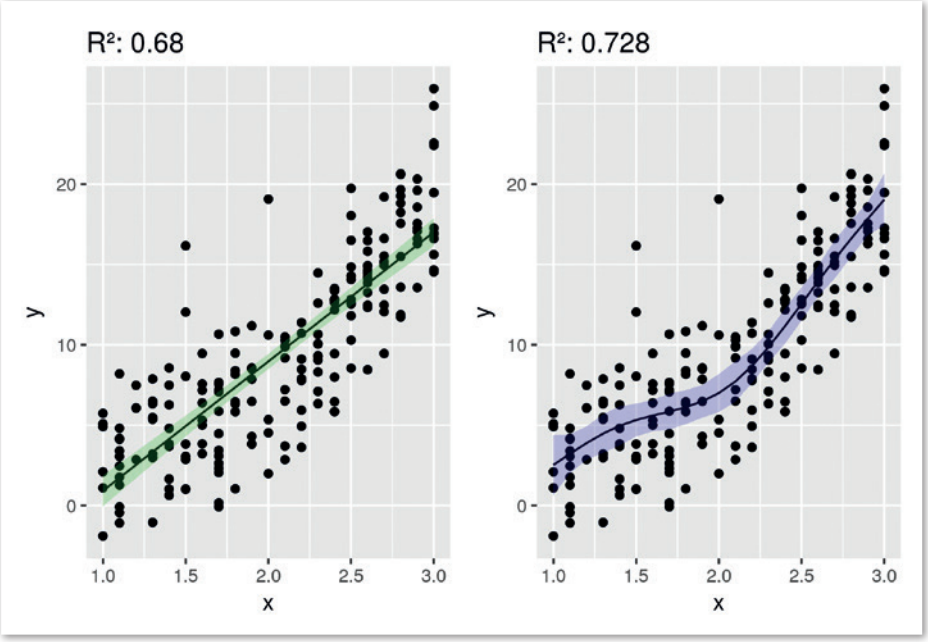


Abbildung 5: Lineare Regression vs. Regression Splines bei einem intrinsisch nichtlinearen Datensatz

ist, im oberen der quadratische. Wir können zwei getrennte Regressionen rechnen und kämen zu einem Ergebnis der Art: „Für niedrigere Wohnflächen ist der Zusammenhang mit dem Verkaufspreis eher linear, für größere eher quadratisch". Vielleicht gehen wir aber noch weiter und zerlegen den Wertebereich des Prädiktors in 5, 10 oder 20 Teile?

Vorhersagen

Wenn der zu beschreibende Zusammenhang intrinsisch nichtlinear ist, ist es sinnvoll, die Grenzen des verwendeten Frameworks zu erweitern. Wir können viele lokale Regressionen rechnen (die ihrerseits noch linear sind) oder wir wählen direkt nichtlineare Methoden wie „Regression Splines". *Abbildung 4* zeigt drei Regression Splines, die sich durch ihren Grad an Flexibilität unterscheiden.

Wir sehen, dass mit zunehmender Flexibilität der „Goodness of fit" auf den Trainingsdaten zunimmt. Wie immer ist es jedoch der Fit auf den Testdaten, auf den es ankommt. Nehmen wir einmal an, das Modell in der Mitte hätte sich in der Kreuzvalidierung als das beste erwiesen, weshalb wir davon ausgehen, dass es auch für die Testdaten die beste Performance zeigen wird. Dann ist die Frage: Ist uns der verbesserte Fit gegenüber der linearen Regression wichtig genug, um auf den explanatorischen Wert der Regression zu verzichten?

In diesem Beispiel ist der Performance-Gewinn wahrscheinlich nicht deutlich genug, als dass wir uns diese Frage wirklich stellen würden – der Zusammenhang ist offenbar linear genug für lineare Regression.

Abbildung 5 zeigt ein solches Beispiel, wenn die wahren Daten definitiv nichtlinear sind. Nehmen wir wieder an, wir sähen jeweils pro Modell den besten Fit, ermittelt durch zehnfache Kreuzvalidierung, links für lineare Regression, rechts für die Familie der Regression Splines. Für welches Modell entscheide ich mich? Die Zahlen sind eine Sache; jetzt kommt es wirklich darauf an, wo die Prioritäten liegen.

Modellselektion

Wir haben bisher nur jeweils einen einzigen Prädiktor betrachtet. In der Regel wird ein Datensatz allerdings mehrere potenzielle Prädiktoren enthalten – in der Zeit von Big Data gegebenenfalls extrem viele. Wie wählt man die wichtigsten aus? Hier wird der Gegensatz zwischen Erklären und Vorhersagen vielleicht noch augenfälliger. Das Gleiche gilt für die Notwendigkeit, die finale Performance auf dem Test Set zu evaluieren. Bei genügend Prädiktoren kann man alle Daten perfekt fitten, ganz nach dem berühmten John-von-Neumann-Zitat „With four parameters I can fit an elephant, and with five I can make him wiggle his trunk".

Kehren wir zurück zu unserem Immobilien-Beispiel. Wir können den Verkaufspreis mit einem Entscheidungsbaum vorhersagen. Dabei beschränken wir uns auf das Subset von Prädiktoren, das auch in *Abbildung 2* gezeigt wird. Wenn ich auf dem Training Set mittels Kreuzvalidierung die besten Parametersettings bestimme, erhalte ich den in *Abbildung 6*, links, gezeigten Baum. Hier werden vier von sechs möglichen Variablen zur Vorhersage verwendet. Der resultierende Split ist der, bei dem wir aufgrund der Kreuzvalidierung den niedrigsten Fehler auf dem Test Set erwarten.

Was passiert, wenn wir den Algorithmus anweisen, keine Rücksicht auf Generalisierbarkeit zu nehmen, sondern die gegebenen Trainingsdaten optimal vorherzusagen? Der resultierende Baum (*Abbildung 6*, rechts) lässt sich schon nicht einmal mehr beschriften, da er komplett unlesbar wäre. Er hat 1.129 Endknoten, bei insgesamt 1.456 Items im Datensatz. Das Training Set wird auf diese

Weise hervorragend beschrieben, es besteht aber keine Chance, dass ein derart verästelter Baum auch das Test Set adäquat beschreibt. Was wir hier sehen, ist „Overfitting", das große „Don't do it" der Data Scientists – don't overfit your training data.

Zurück zu unserem durch Kreuzvalidierung gewählten Baum. Wie erklärend ist das Modell? Entscheidungsbäume werden im Allgemeinen als gut kommunizierbar und leidlich explanatorisch angesehen. Das gilt aber umso weniger, je mehr Splits der Baum hat und je mehr ein- und dieselbe Variable für verschiedene Splits wiederverwendet wird. Wir können den Vorhersagefehler gegen die Anzahl der Blätter plotten und uns für den Baum mit dem besten Trade-off entscheiden. Im vorliegenden Beispiel etwa für den Baum mit fünf Endknoten *(siehe Abbildung 7)*, der gleichermaßen Einsicht vermittelt und die Daten gut beschreibt.

Was ist, wenn ich nicht erklären will, sondern tatsächlich einfach die bestmögliche Vorhersage brauche? Dann werde ich dezidiert eine der Methoden nehmen, die erfahrungsgemäß den Shootout der Algorithmen gewinnen (etwa bei Data Science Competitions wie auf Kaggle). Am erfolgreichsten sind in der Regel Ensemble-Methoden, also Methoden, die einen Algorithmus mehrfach anwenden und das Ergebnis mitteln. Im Bereich der Entscheidungsbäume sind das „Random Forests" und „Boosting".

Gemeinsam ist beiden Ansätzen, dass sie die guten Ergebnisse gerade dadurch erzielen, dass man es sich schwerer macht, um dadurch ein robusteres, von der Zufälligkeit der Trainingsdaten unabhängigeres Ergebnis zu erzielen. Im Falle von Random Forests geschieht das dadurch, dass nicht jeder vorhandene Prädiktor für einen anstehenden Split in Erwägung gezogen wird – die Kandidaten werden jeweils zufällig ausgewählt, woraus sich der Name „Random Forest" herleitet. Im Falle von Boosting ist es eine Kombination „schwacher Lerner", die in Synergie ein sehr gutes Ergebnis erzielen.

Es sich schwerer machen, um besser zu werden: Das ist ein Motto, unter das man viele erfolgreiche Methoden des Machine Learning stellen könnte. Das gilt auch für das schon mehrmals angekündigte, nachfolgend beschriebene Deep Learning.

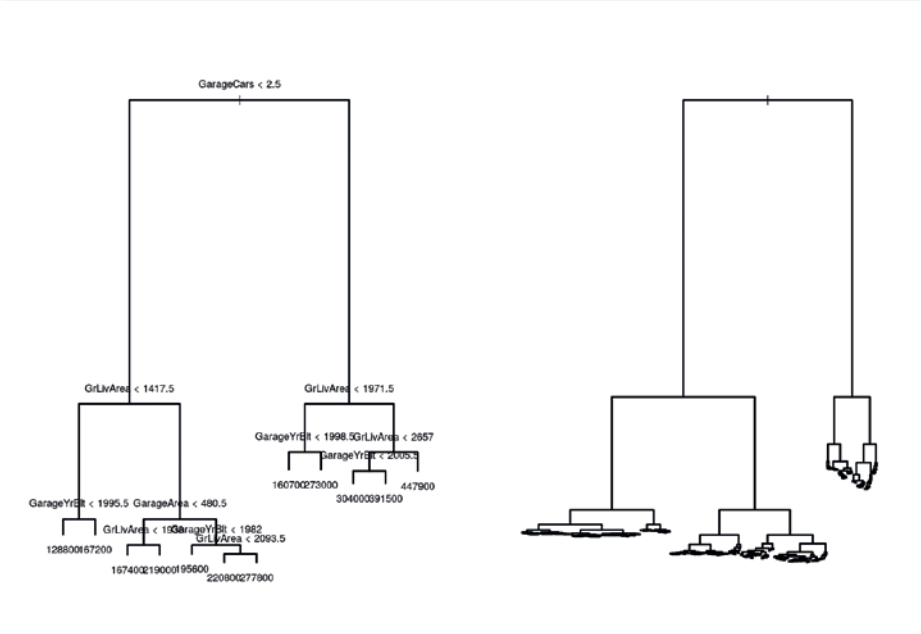


Abbildung 6: Entscheidungsbäume, links der durch Kreuzvalidierung ermittelte Baum, rechts ein Baum mit 1.129 Endknoten, der die Trainingsdaten nahezu perfekt erklärt, aber kläglich an den Testdaten scheitern wird

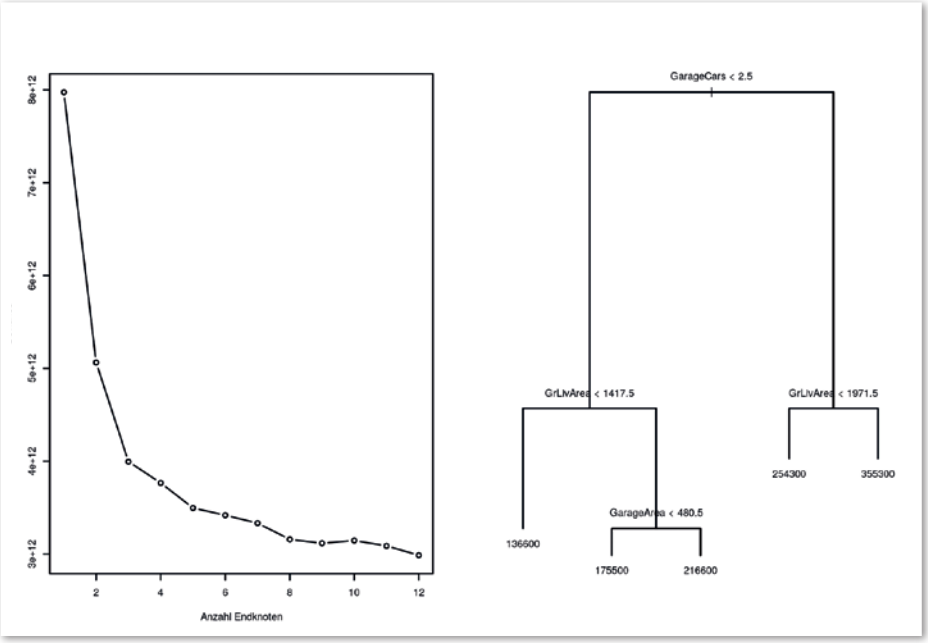


Abbildung 7: Der Baum mit fünf Endknoten ist ein guter Kompromiss zwischen Minimierung des Vorhersagefehlers und explanatorischem Wert, links die Vorhersagefehler, abhängig von der Anzahl der Endknoten, rechts der Baum mit fünf Knoten

Heute „magic", morgen „mainstream": Deep Learning

Wenn wir von „Data Science Competitions" sprechen, dabei den Begriff etwas lockerer fassen und auf „Machine Learning" bis zu „künstliche Intelligenz" ausweiten, kommen wir an Deep Learning nicht vorbei. Deep Learning findet heute bei Objekt- und Sprach-Erkennung, maschineller Übersetzung, Sprach- und Musik-Generierung sowie Generierung von Produktempfehlungen bis hin zur Generierung von „fake evidence", also Bildern, Audios, Videos, die es nie gegeben hat [17], am meisten Anwendung und ist in der Regel dort die Methode der Wahl.

Deep Learning ist nichts als ein neuer Begriff für etwas, das es seit der Mitte des letzten Jahrhunderts gibt und dessen wichtigste Konzepte in den 1980er- und 1990er-Jahren entwickelt wurden: künstliche neuronale Netze, vage an der Architektur biologischer Neuronen orientiert, mit einem Input und einem Output Layer und dazwischen mindestens einem, oft mehreren Hidden Layern. Diese sind es, die die Netze beim Deep Learning „deep" machen *(siehe Abbildung 8)*.

Strukturen durch Feature-Generierung erkennen

Deep Learning macht nicht die bloße Aneinanderreihung von Layern so erfolgreich. In den Hidden Layern bilden sich Strukturen, die wichtige Eigenschaften des Input-Objekts abbilden und deren Synthese wiederum neue Strukturen generiert. Deswegen sind die Netze auch keine amorphe Masse von „Computing Units", sondern haben jeweils eine der Aufgabenstellung angemessene Architektur.

Im Bereich „Objekt-Erkennung und -Klassifizierung" werden beispielsweise Convolutional Neural Networks (CNN) eingesetzt *(siehe Abbildung 9)*. Das Netz soll sagen, ob das Bild im Input eine Person, ein Tier, ein Auto etc. ist. Der Input sind aber lediglich Pixel. Wie kann das funktionieren? Die sukzessive Extraktion komplexerer Strukturen ist die Aufgabe der Hidden Layer. Zum Beispiel so: Im ersten Layer werden Kanten identifiziert, im zweiten Ecken und Konturen, im dritten Teile von Objekten (eine Nase, ein Arm etc.) und so weiter, bis am Ende die Klassifikation erfolgt: „Das ist eine Person". Dabei werden also sukzessive komplexere Features erkannt, ohne dass der Programmierer diese von außen vorgegeben hätte.

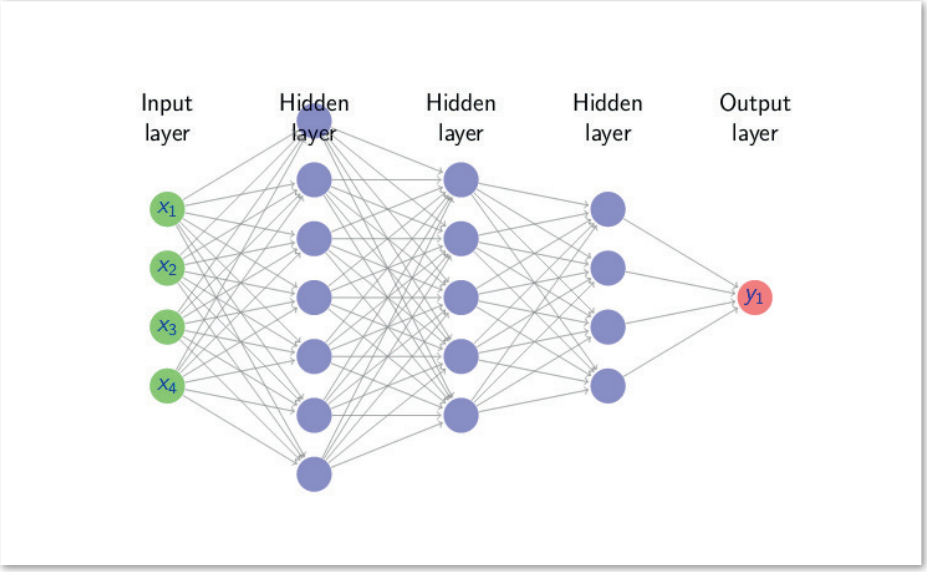


Abbildung 8: Schematische Struktur eines Deep Neural Network; die Anzahl der Hidden Layer ist variabel [15]

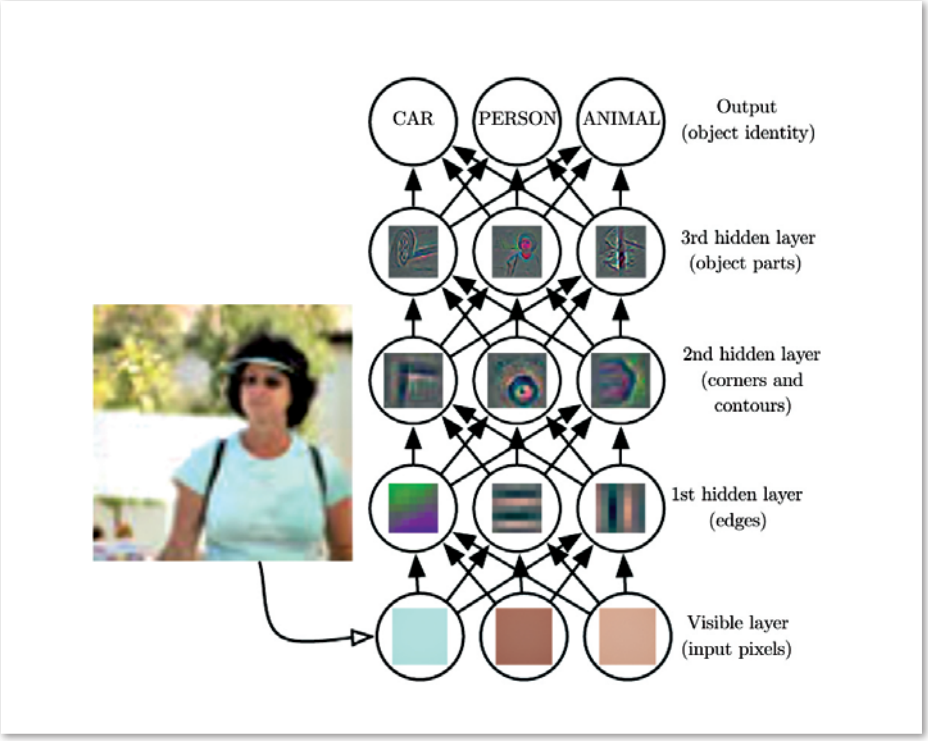


Abbildung 9: Beispielhafte Darstellung der Funktion der Hidden Layer bei einem Convolutional Neural Network [12]

Sequenzen: Sprache & Co.

Was ist mit Daten, die eine inhärente sequenzielle Struktur haben, also Sprache, Zeitreihen, Musik etc. Das ist die Domäne der Recurrent Neural Networks (RNN). Diese haben zusätzlich zu der Tiefenstruktur noch eine weitere verborgene Struktur, den Hidden State. Dieser transportiert die Information über vorherige Inputs. Im Bereich der RNNs ist der Model Zoo, also die Vielfalt existenter Architekturen, besonders groß. Die aktuell am meisten eingesetzte Architektur ist wahrscheinlich das Long Short Term Memory Network (LSTM).

Eine sehr praxisrelevante Anwendung, Google Translate, ist eine maschinelle Übersetzung. Hier sind nicht nur die Eingabedaten sequenziell, sondern auch die Ausgabedaten, was das Problem noch weiter kompliziert. Es kommen sogenannte „seq-2seq-Architekturen" zum Einsatz, mit je einem RNN für den Input und den Output *(siehe Abbildung 10)*. In komplexeren Modellen als dem abgebildeten gibt es zudem einen Aufmerksamkeitsmechanismus („attention"), der dem Netz sagt, welcher Teil des Inputs für welchen Teil des Outputs relevant ist.

Deep Learning in der Praxis

Eine Vielfalt von Aufgabenstellungen, Architekturen, Hyper-Parametern und Frameworks – ist das nicht viel zu komplex, um im Alltag anwendbar zu sein? Ganz zu schweigen von der Rechenpower, die es braucht, um die optimalen Hyperparameter-Settings zu finden, und der Menge an Trainingsdaten, mit der das Netz gefüttert werden muss?

Erstaunlicherweise ist Deep Learning sehr viel anwendbarer, als man meinen könnte. Passable Performance lässt sich oft auch schon mit geringeren Datenmengen und überschaubarem Trainingsaufwand erreichen. Um das Letzte an Performance herauszukitzeln, verwendet man vortrainierte Modelle, die frei heruntergeladen werden können. Auch konzeptuell ist vieles aus dem traditionellen Machine Learning auf das Deep Learning übertragbar, etwa das erwähnte Prinzip des „Es-sich-schwerer-Machens", das im Deep Learning als „drop out"-Hyper-Parameter auftritt und das Ausmaß

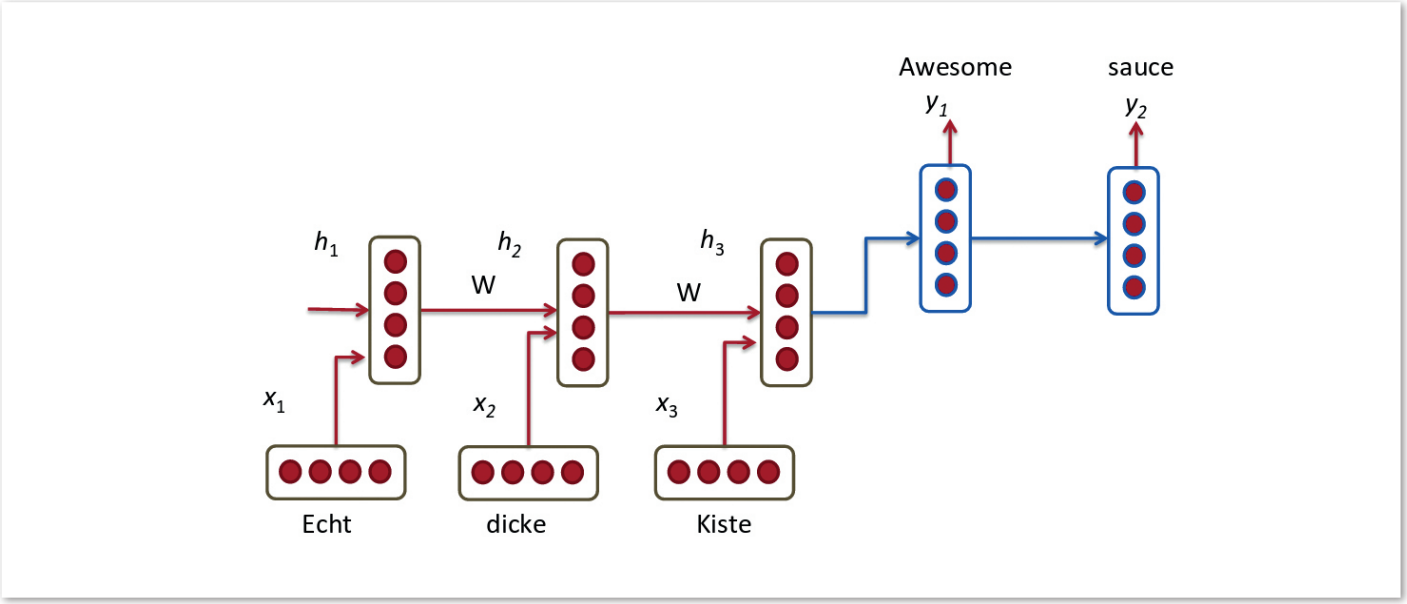


Abbildung 10: „seq2seq"-Architektur für maschinelle Übersetzung [13]

bezeichnet, in dem Neuronen im Training zufällig ausgeschaltet werden. Auf diese Weise wird im Training „noise" erzeugt, der zu einer besseren Generalisierung und damit auch zu besserer Performance auf neuen Daten führt.

Statistik, Machine Learning und Deep Learning haben, so unterschiedlich sie auf den ersten Blick wirken, doch ganz grundlegende konzeptuelle Gemeinsamkeiten, die ihren synergetischen Einsatz ganz natürlich erscheinen lassen.

Literatur

Natürlich gibt es eine Reihe hervorragender Einführungen in die Statistik, auf jedem gewünschten Niveau. Spannend sind vor allem neue Sichtweisen, die die letzten Jahre gebracht haben, unter anderem als Folge der Replikationskrise in der Psychologie und der zunehmenden praktischen Anwendbarkeit Bayes'scher Statistik. Empfohlen seien hier die hervorragenden MOOCs „Statistical Inference" der Johns Hopkins Universität [5] und „Improving your statistical inferences" der TU Eindhoven [6].

Welche Erkenntnisse sich aus der visuellen Aufbereitung von Daten ziehen lassen, zeigen sehr schön die Bücher des berühmten Statistikers John Tukey und seiner Mitautoren über explorative Datenanalyse. Als Beispiel ist hier „Graphical methods for data analysis" empfohlen [7].

Im Übergangsbereich zwischen Datenexploration und Grafikdesign liegen die auch optisch sehr ansprechenden Bücher von Edward Tufte, etwa das jüngste mit dem programmatischen Namen „Beautiful evidence" [1]. Der augenöffnende Essay über den „Cognitive Style of Powerpoint" ist frei als PDF verfügbar [2].

Auch zum Thema „Data Science" gibt es einige hervorragende MOOCs, unter anderem auch wieder von der Johns Hopkins University. Zur Abwechslung sei hier aber ein (noch dazu relativ dünnes) Buch empfohlen: das hervorragend konzipierte und geschriebene „Introduction to Statistical Learning" von Hastie et al. Zusätzlich zu den wunderbar klaren Erläuterungen enthält das Buch praktischen R-Code, mit dem die Leser gleich praktische Erfahrungen sammeln können. Wer mehr mathematische Vorkenntnisse mitbringt, für den lohnt sich die Lektüre von Kevin Murphys „Machine Learning: A probabilistic perspective" [9].

Das wichtigste Buch zu Deep Learning ist derzeit das gleichnamige MIT-Buch von Goodfellow et al. [12]. Ansonsten sind es vor allem eine Reihe auf YouTube verfügbarer Vorlesungen, die Einblicke in das rasante Geschehen auf dem Gebiet geben, so zum Beispiel „Natural Language Processing with Deep Learning" aus Stanford [13] oder „Deep Learning for self-driving cars" vom MIT [14]. Eine gleichzeitig gut zugängliche und gute Einführung in Deep Learning ist die Vorlesung von Ali Ghodsi der University of Waterloo [15].

Quellen und Literatur

[1] Edward Tufte, Beautiful Evidence, 2006
[2] *https://www.inf.ed.ac.uk/teaching/courses/pi/2016_2017/phil/tufte-powerpoint.pdf*
[3] *http://charliepark.org/slopegraphs*
[4] *https://www.edwardtufte.com/bboard/q-and-a-fetch-msg?msg_id=0001OR&topic_id=1*
[5] Johns Hopkins University, Statistical Inference: *https://www.coursera.org/learn/statistical-inference*
[6] TU Eindhoven, Improving your statistical inferences: *https://www.coursera.org/learn/statistical-inferences*
[7] Chambers et al., Graphical methods for data analysis, 1983
[8] James et al., Introduction to statistical learning, 2013: *http://www-bcf.usc.edu/~gareth/ISL*
[9] Kevin Murphy, Machine Learning: A probabilistic perspective, 2012
[10] *https://github.com/stephlocke/datasauRus*
[12] Goodfellow et al., Deep Learning, 2016: *http://www.deeplearningbook.org*
[13] Stanford University, Natural Language Processing with Deep Learning: *http://web.stanford.edu/class/cs224n*
[14] MIT, Deep Learning for self-driving cars: *http://selfdrivingcars.mit.edu*
[15] University of Waterloo, Deep Learning: *https://uwaterloo.ca/data-science/deep-learning*
[16] Ames Housing Dataset: *http://www.amstat.org/publications/jse/v19n3/decock.pdf*
[17] *http://www.economist.com/news/science-and-technology/21724370-fake-news-you-aint-seen-nothing-yet-generating-convincing-audio-and-video-fake*

Sigrid Keydana
sigrid.keydana@trivadis.com

Predictive Machine Learning – Analysen und Massendaten

Alfred Schlaucher, ORACLE Deutschland B.V. & Co. KG

Der Artikel beschreibt anhand von drei Beispielen (Kunden-, Maschinen- und Text-Daten) die Herausforderungen bei der Anwendung von Machine-Learning-Verfahren auf große Datenmengen in der kommerziellen IT. Ein überschaubares Analyse-Setup, beispielsweise mit der Statistiksprache R auf einem Laptop, muss auch mit Massendaten zurechtkommen, die in unternehmensweit angelegten Oracle-Datenbanken vorliegen.

Viele Prinzipien des Machine Learning lassen sich an einem einfachen Beispiel erklären. Es geht letztlich um die Bestimmung einer Wahrscheinlichkeit, dass etwas passiert. Eine persönliche Erfahrung mit dieser Wahrscheinlichkeit kennen wir schon seit Kindertagen: den altbekannten Würfel mit seinen sechs Augen. Die Wahrscheinlichkeit, dass wir eine „6" würfeln, liegt bei 1/6*100, also etwa 16 Prozent. Trotzdem weiß jeder, der zwanzig Mal würfelt, dass nur eine „6" dabeisein kann. Aber die berechnete 16-Prozent-Wahrscheinlichkeit stimmt, wenn wir nicht zwanzig Mal, sondern zehntausend Mal würfeln – dann haben wir in 16 Prozent aller Versuche einen 6er-Erfolg *(siehe Abbildung 1)*.

Was zeigt uns das? Die Aussage über die Wahrscheinlichkeit des Eintretens eines Ereignisses wird glaubwürdiger, je größer der Umfang der Ereignisdaten oder Beobachtungen ist. Also beschaffen wir möglichst viele Daten zu Ereignissen, damit Analysen zu Wahrscheinlichkeiten und Vorhersagen stimmiger werden. Ein altes Gesetz aus der Statistik: „Masse sticht, je mehr desto genauer". Unternehmen mit großen Datenbeständen können also glücklich sein.

Die frühen Statistiker in der Medizin und Soziologie hatten es fast immer mit unbekannten Gesamtmengen zu tun. Unbekannt waren „die Menge aller Kranken", „die Menge aller Bewohner einer Region" etc. Die Grundmenge entzog sich einer Analyse als Ganzes. Es war unmöglich, alle kranken Menschen in einem Land als Ganzes zu analysieren. Doch man konnte sich helfen, entnahm Stichproben aus den unbekannten Gesamtmengen – und um die Analysen dieser Stichproben zu verifizieren, entnahm man noch eine Stichprobe und dann noch eine. Letztlich konnte man das Problem der fehlenden Gesamtmengen durch eine gewisse Anzahl von Stichproben lösen. Das Schöne daran: Je mehr Stichproben man nahm, desto wahrscheinlicher wurde ein richtiges Ergebnis.

Wechseln wir in die kommerzielle IT der Unternehmen, dann benötigen wir allerdings keine Stichproben mehr. Wir verfügen meist über Gesamtmengen. Das Anwenden von Machine-Learning-Verfahren auf ein Datenvolumen von Excel-Tabellen ist ohne Vorkehrungen praktikabel. Sensor-, Bon-, Log- und Bewegungs-Daten sowie Textmassen oder auch nur die Fülle der Kundenmerk-

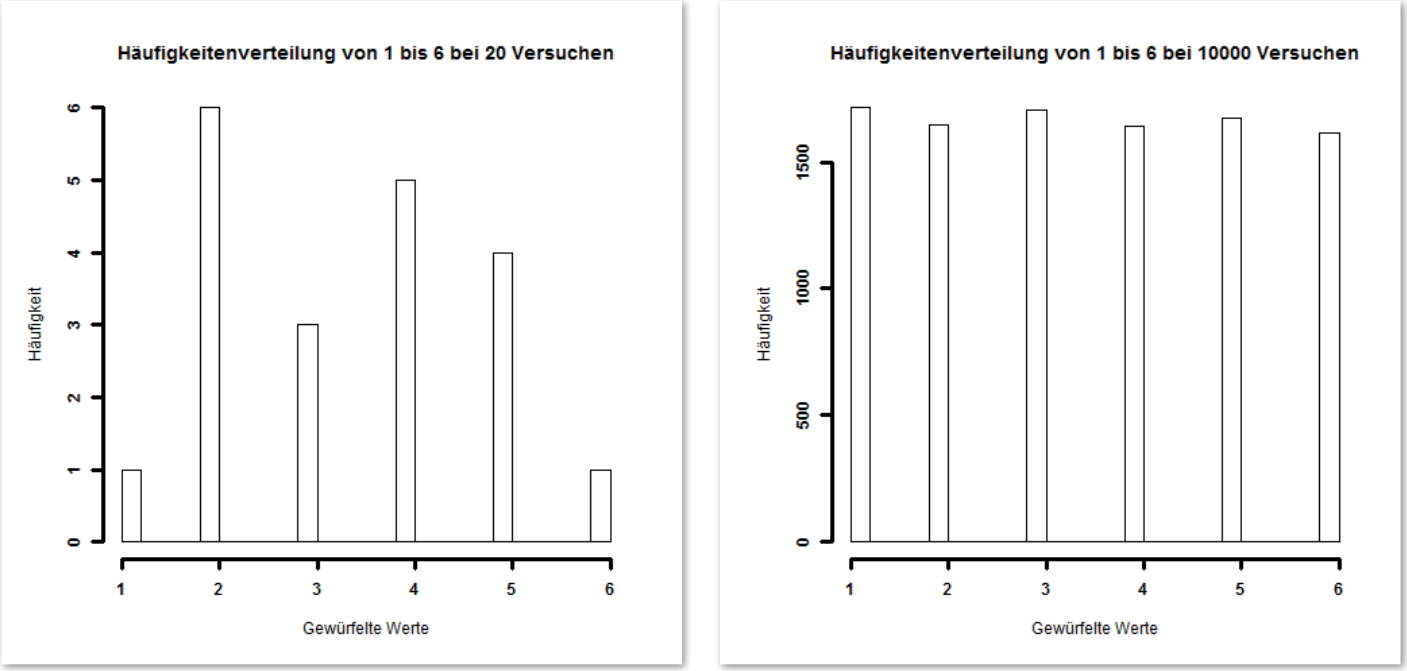


Abbildung 1: Die Verlässlichkeit einer Vorhersage steigt mit der Menge der Erfahrungswerte

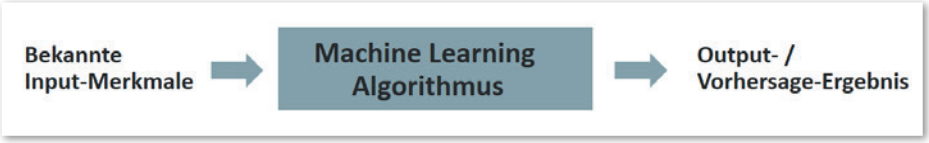


Abbildung 2: Anwendungsschema Machine-Learning-Algorithmus

male in den Stammdaten sprengen dagegen jeden Analyse-Prozess zum Finden von Abhängigkeiten oder zum Definieren von Modellen mit herkömmlicher Technik trotz funktionierender Algorithmen.

Predictive Analytics: Um welche Analyse-Schritte es geht

Um Ereignisse vorherzusagen, benötigen wir etwas, das wir mit bekannten Daten füttern können und das uns dann eine Information über künftige Entwicklungen verrät. So etwas nennt man Modell. Ein Modell ist letztlich eine irgendwie geartete Funktion (mathematische Formel, sprachlich formulierte Regeln, Select-/SQL-Statement etc.), bei der man eine Reihe bekannter Merkmale von etwas (Parameter oder Variablen) festlegt und die dann von diesen Merkmalen abgeleitet eine Zielinformation (Vorhersage-Ergebnis) liefert *(siehe Abbildung 2)*.

Man benutzt ein Modell, etwa um den Typus eines Kunden zu bestimmen, der Niedrigpreis-Produkte präferiert, oder – um ein Beispiel aus der Fertigungsindustrie zu nennen – man nutzt Modelle, um den Zeitpunkt vorherzusagen, an dem ein Elektromotor durch sein automatisches Ausschalten eine Produktionsstraße stillstehen lässt, nachdem ein Temperaturschwellwert überschritten wurde. In beiden Beispielen sucht man nach Konstellationen von bekannten Merkmalswerten, um ein Verhalten in der Zukunft vorherzusagen *(siehe Abbildung 3)*.

Auch hier gilt: Je mehr Informationen man in der Vergangenheit sammeln konnte, desto verlässlicher wird eine Vorhersage sein. Dieses „Mehr an Informationen" bezieht sich dabei sowohl auf die absolute Anzahl von Beobachtungen (Anzahl Kunden, Anzahl Messwerte) als auch auf die Anzahl der verschiedenen Merkmale. Wir können die Herausforderungen schon ahnen, denn gerade die Menge der Messwerte von Maschinen kann in die Milliarden gehen. Man misst solche Werte unter Umständen im Sekunden- oder Minuten-Abstand und die Maschinen können über Monate ohne Unterbrechung laufen. Auch bei Kundendaten sammeln sich eventuell Hunderte von Einzel-Informationen an, die als Merkmal zum Verhalten eines Kunden beitragen können.

Abbildung 3: Die determinierenden Merkmale für einen „Niedrig-Preis-Kunden" (links) und die Merkmalswerte für das Aussetzen der Maschine (rechts)

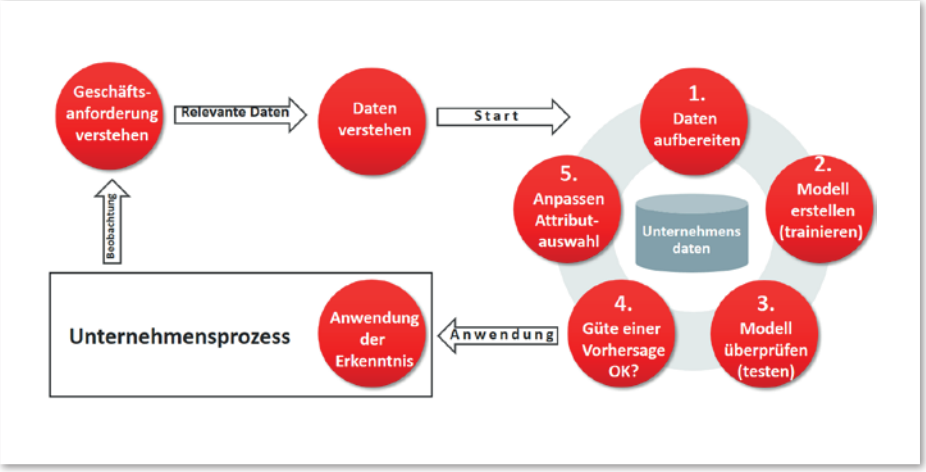


Abbildung 4: Phasen von Vorhersage-Analysen

Die Phasen einer Vorhersage-Analyse lassen sich schematisch wie in *Abbildung 4* zusammenfassen.

Wichtigste Erkenntnis aus diesem Bild ist: Das Entwickeln eines Modells ist ein iterativer Vorgang. Ein Vorhersagemodell wird nie so genau sein, dass es die Realität zu 100 Prozent abbilden kann. Es wird immer nur eine Näherung sein und es beschreibt eine Realität nur mehr oder weniger genau. Dabei startet man mit der Masse aller bekannten Merkmale (Attribute), in der Hoffnung, damit kein relevantes Input-Merkmal vergessen zu haben. Schließlich entfernt man nach und nach einzelne Attribute aus der Analyse (Schritt 5) und prüft, wie weit sich eine Vorhersage durch das Modell auf bekannte Ergebnisse (Testdaten) verschlechtert (Schritt 3). Am Ende des Vorgangs steht ein Kompromiss zwischen der Anzahl der für eine Vorhersage am meisten relevanten Attribute und der weniger hohen (aber akzeptierten) Genauigkeit der Vorhersage (Schritt 4). Ziel ist ein einfaches, robustes und vor allem leicht handhabbares Modell, denn ein solches Modell muss schließlich in einem operativen Unternehmensprozess, etwa in der Fertigung, permanent, eventuell in Realzeit, eingesetzt werden können.

Nicht nur ein Mengen- sondern auch ein Komplexitätsproblem

Hier offenbart sich die zweite Herausforderung von Predictive Analysen bei Unternehmensdaten: Starten wir den Analyse-Prozess mit der maximalen Anzahl der zur Verfügung stehenden Merkmale, so haben wir zumindest zu dessen Beginn nicht nur ein Datenmengen- sondern auch ein Komplexitätsproblem, denn Analyse-Algorithmen müssen oft paarweise die wechselseitige Abhängigkeit der Merkmale untereinander prüfen. So sollen etwa bei einem linearen Regressionsmodell die Input-Merkmals-Variablen unabhängig voneinander sein, also keine Kovarianzen haben (das genannte Maschinendaten-Beispiel erfüllt diese Anforderung zum Teil nicht). Bei einem Sensordatensatz mit mehreren Hundert Messwerten pro Zeiteinheit erfordert dies besonders hohe Rechenleistungen.

Wann kauft jemand „Niedrig-Preis-Produkte“

Verfeinern wir die Betrachtung der Analyse-Schritte noch etwas mehr, um die Klippen, die durch große Datenmengen und Attribut-Komplexität entstehen, deutlicher aufzeigen zu können, und wählen als erstes Beispiel die Verhaltensvorhersage eines Kunden, wonach wir aufgrund bekannter Merkmale einen Kunden einer Kategorie „Niedrig_Preis_Kunde“ („Kauft gerne günstige Produkte“, „ist affin für Sonderangebote“) zuordnen können. Wir nutzen dafür einen Naive-Bayes-Algorithmus. Dieser lebt davon, dass wir das Verhalten von einem Teil unserer Kunden schon kennen. Für einen Teil der Kunden wissen wir also schon, dass eine Affinität für Niedrigpreis-Produkte vorliegt, und wir kennen auch die Werte der übrigen Kundenmerkmale. Das wird unser sogenannter „Trainingsdatenbestand“ werden. Der Algorithmus berechnet für jeden dieser bekannten Werte die Wahrscheinlichkeit des Auftretens (Prozentwert) und markiert das Ergebnis mit der Information „Kunde ist affin für Niedrig-Preis-Produkte“ (ja oder nein beziehungsweise 1 oder 0). Dadurch entsteht ein sogenannter „Classifier“. Ein Classifier (-Modell) ist letztlich eine Tabelle, in der die Vorkommens-Häufigkeiten aller Werte aller betrachteten, bekannten Kundenmerkmale gemeinsam mit dem erwarteten Ergebnis einer Zielvariablen (0 oder 1 für „ist affin für Niedrig-Preis-Produkte“) aufgelistet sind *(siehe Abbildung 5)*.

Ein solcher Classifier ist jetzt das Hilfsmittel, mit dem neue Kunden mit bislang unbekanntem Kaufverhalten kategorisiert werden können – das Kundenverhalten wird vorhersehbar. Neben der Unterstützung von individualisierten Angebotskampagnen kann man mit einem solchen Classifier auch einem Kunden mit bis dahin unbekannten Eigenschaften, noch während er eine Kaufaktivität vornimmt, spontan neue Sonderangebote präsentieren.

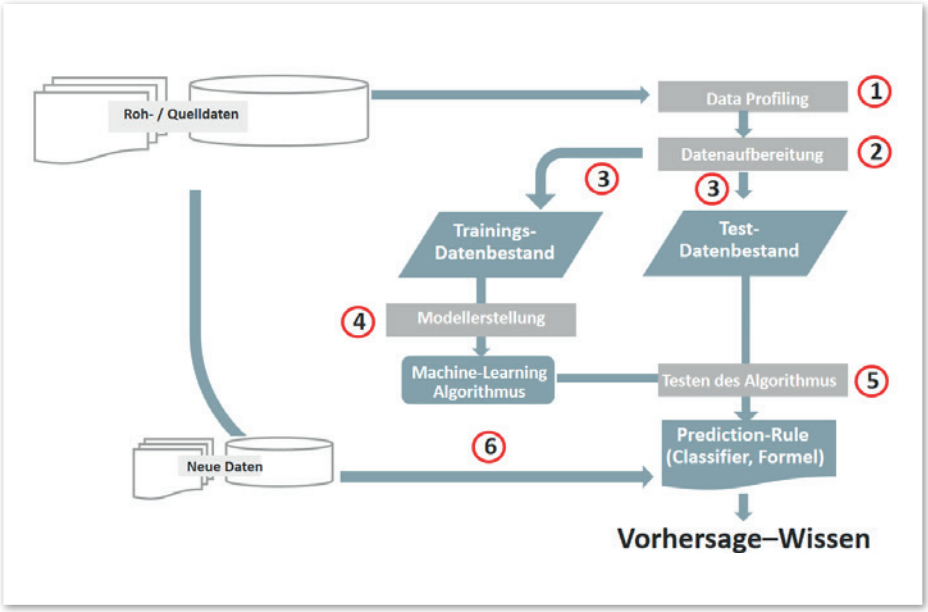


Abbildung 5: Schrittfolge Analyse-Prozess zur Erstellung eines Classifier

Artikelstamm (Anzahl Variablen)	Kundenstamm (Anzahl Variablen)	Kauftransaktionen (Anzahl Variablen)	Analyse-Objekt (Join zwischen Kunde, Artikel und Umsatz, Objektgröße in GB)
15	43	22	1,7
30	43	22	1,9
30	88	22	2,4
30	88	32	2,6

Tabelle 1

Zur Prüfung der Vorhersagegüte des Classifier teilt man vor der Modellerstellung den bekannten Datenbestand in einen Trainings- und Testdatenbestand. Mit den Daten des Trainingsdatenbestands erstellt man den Classifier. Über den Testdatenbestand überprüft man die Vorhersagefähigkeit des Classifier gegenüber den eigentlich schon bekannten Daten und kennt die Wahrscheinlichkeit, mit der ein Vorhersageergebnis richtig ist.

Auf dem Weg zu einem brauchbaren Classifier sind jedoch ein paar Hürden zu überwinden. Die erste Aufgabe ist das Kennenlernen der Daten mithilfe des sogenannten „Data Profiling". Hier erstellt man für alle Attribute der Ausgangsdaten (Kundendatensatz) ein Profil mit Wertebereichen, Struktur der Werte, fehlenden Werten etc. Über ein solches Profil kann man bereits unbrauchbare Attribute aussortieren und den Aufwand in den Folgeschritten minimieren. Das kann ein sehr umfangreiches Unterfangen sein. In unserem Beispiel betrachten wir das Kaufverhalten von Kunden anhand getätigter Kaufentscheidungen. Die Datenbasis der Analyse ist also nicht ein überschaubar großer Kundenstammdatensatz, sondern die Masse der Kauftransaktionen der letzten Monate, gekoppelt mit den bekannten Merkmalen der Kunden in dem Kundenstammsatz.

In den Data Marts heutiger Analyse-Datenbanken sind das die Join-Produkte aller Werte der dort enthaltenen Star-Schemen. Zur Erstellung der Ausgangsdaten für die Naive-Bayes-Analyse müssen wir also zunächst eine Join-Verbindung zwischen Stamm- und Transaktionsdaten herstellen. Es entsteht ein neues, wesentlich größeres Datenobjekt. Um das noch einmal hervorzuheben: Während in einer Analyse-Datenbank die Daten noch in getrennten Tabellen liegen, benötigt der Algorithmus ein einziges Verbundobjekt, das aufgrund der Join-Mechanik naturgegeben wesentlich größer ist als die einzelnen Datentabellen. Das veranschaulicht die folgende Tabelle aus einem R-Analyse-Szenario auf den drei Datenbank-Tabellen *(siehe Tabelle 1)* „ARTIKEL" (600 Sätze, 15 Attribute), „KUNDE" (3000 Sätze, 43 Attribute) und „Umsatz" (4 Millionen Sätze, 22 Attribute).

Das Volumen der Umsatztabelle verdoppelt sich. Zu bemerken ist, dass die Anzahl der Attribute bei den Artikeln und Kunden sowie die Anzahl der Sätze in diesem Szenario relativ klein sind. Der Sinn dieser Zahlenwerte liegt darin, ein Gefühl für die Größe zu bekommen. Höhere Werte erhält man durch Multiplikation. Wenn wir also anstatt 4 Millionen Kauftransaktionen 40 Millionen nehmen, wächst die Analyse-Grundmenge auf 26 GB. Wenn wir anstatt 15 Artikelspalten einen realistischeren Wert von 50 nehmen, dann steigt das Volumen auf 50 GB, und wenn wir dann noch die Anzahl der Kundenattribute von 43 auf 100 erhöhen, ist das Analyse-Objekt schon etwa 80 GB groß. Diese Vorstellung wird dann dramatisch, wenn man weiß, dass ein Naive-Bayes-Algorithmus beispielsweise mit der Analysesprache R die Daten zunächst komplett in den Hauptspeicher der jeweiligen Rechner laden muss.

Man erkennt, dass dies keine Aktivität mehr für einen Arbeitsplatz-PC ist und dass das Data Profiling und die Berechnung eines Classifier auf einen größeren Server gehören. Für die Statistiksprache R bedeutet dies: R muss auf einem großen Server und am besten in der Nähe der Unternehmensdaten betrieben werden. Ein Transport solcher Daten auf einen PC ist fast ausgeschlossen.

Modell und Daten zusammenbringen

Ist der Classifier (das Modell) einmal erstellt, hat man das Größenproblem hinter sich gebracht, denn die nun folgende Anwendung des Classifier findet jetzt nur noch auf einzelnen neuen Kundensätzen statt. Aber es kommt eine weitere Herausforderung auf uns zu: Um neue Kundensätze zu bewerten („scoren"), bringen wir am einfachsten den Classifier dorthin, wo sich neue Daten befinden. Das ist in der Regel eine Transaktions-Datenbank, in der die operativen Daten des Unternehmens anfallen.

Wenn wir mit der Statistiksprache R arbeiten, dann ist der Classifier aus technischer Sicht ein R-Objekt vom Typ „Naive Bayes", das man mithilfe einer sogenannten „Predict-Funktion" aufruft und zur Anwendung auf neue Daten bringt. Will man dazu die Kaufdaten nicht aus der Datenbank herauslösen, sondern dort scoren, wo sie anfallen und gelagert werden, muss man diesen Predict-Vorgang, also den Vorgang der Modell-Anwendung, mit SQL innerhalb der Datenhaltung initiieren. Das ist beispielsweise über die R-Integration in einer Oracle-Datenbank leicht möglich. Problem gelöst.

Zweites Beispiel: „Wann brennt die Hütte"

Das vorgenannte Handels-Kundenbeispiel beschäftigte sich mit kategorialen Daten, also meist fixen Merkmalswerten zu Kunden (Wohnart, Familienstand etc.). Das war auch ein Grund für die Wahl des Naive-Bayes-Algorithmus. Das zweite Beispiel zu Maschinen-Messdaten nutzt überwiegend metrische Daten, also Zahlenwerte zu Temperaturen, Umdrehungen, Kräften etc. Hier versucht ein Regressionsalgorithmus, einen mathematisch beschreibbaren Zusammenhang zwischen den physischen Merkmalen eines Elektromotors zu finden, der sich beim Überschreiten eines Temperaturlimits innerhalb seiner Feldwicklung abschaltet. Um das Abschalten und damit Produktionsausfälle zu verhindern, sucht man nach den Rahmenbedingungen mit dem größten Einfluss auf den Temperaturanstieg innerhalb dieser Drahtwicklung *(siehe Abbildung 6)*.

Die Parameter folgen dabei durchaus unterschiedlichen Gesetzen und verändern sich nicht immer linear. Steigt das Drehmoment, weil die Maschine mehr leisten muss oder durch äußere Einflüsse gebremst wird, nimmt auch die Stromaufnahme zu, das wäre ein multiplikativer Zusammenhang („ax"). Gleichzeit läuft die Maschine jedoch langsamer, hinsichtlich der Drehzahl müssen wir eine Division in eine mögliche Beschreibungsformel übernehmen ($\frac{1}{b}x$). Das Gleiche gilt für die Kühlung, die über ein Flügelblatt

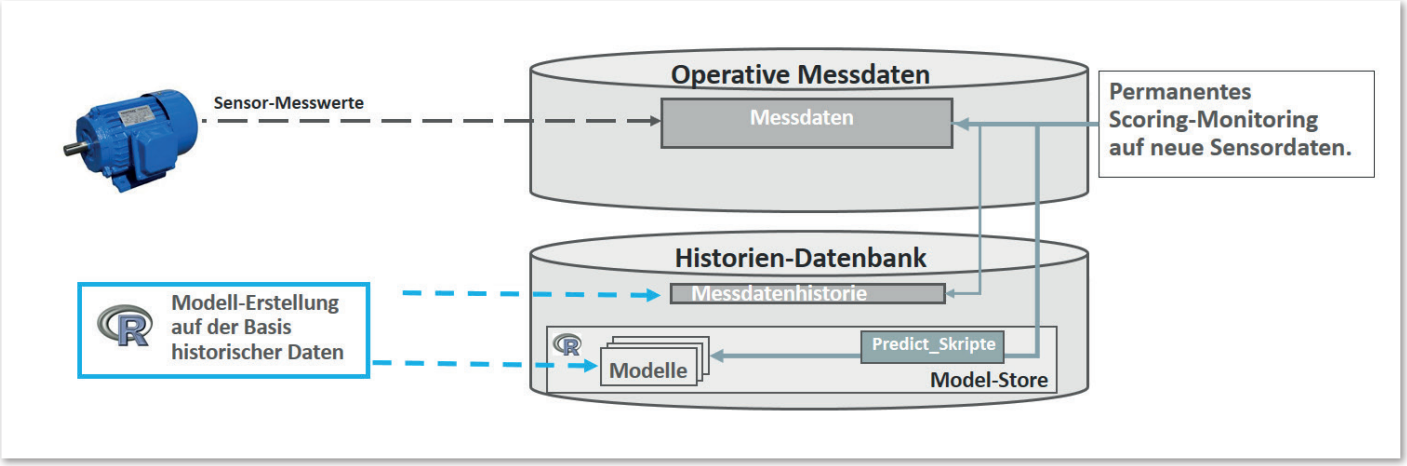


Abbildung 6: Messdaten-Szenario – Verwendung von Modellen zum permanenten Monitoring von Maschinen-Laufzeitdaten

auf dem Rotor erzeugt wird. Dreht sich die Maschine langsamer, wird wegen der starren Verbindung weniger gekühlt. Die Temperatur steigt dadurch jedoch und erfährt einen zusätzlichen positiven Schub.

Kommt die Maschine durch eine Blockade an einem Fertigungslaufband zum Stillstand, steigt die Stromaufnahme sogar exponentiell nach oben (x^n), denn es fällt der durch den drehenden Rotor entstehende Gegenstrom weg, der bei einer drehenden Maschine einen elektrischen Widerstand zur Stromaufnahme darstellt. Das wirkt fast wie ein Kurzschluss. Wenn die Maschine nicht selbstständig durch einen Wärmeschutzmechanismus abschaltet, kann sie sogar nach wenigen Sekunden Feuer fangen *(siehe Abbildung 7)*.

Hier gibt es für einen Algorithmus also eine Menge zu erkunden. Ein Regressions-Algorithmus nimmt die Veränderungen aller Werte über eine Zeitdistanz wahr und versucht aufgrund der Varianzen dieser Werte eine Formel zu bilden, die sich idealisiert als Linie oder Kurve in einem Koordinaten-System darstellen lässt. Das Ergebnis ist eine Formel, mit der man eine mögliche kritische Temperatur in dem Motor berechnen kann, wenn Drehmomentkräfte, Umgebungstemperaturen, Differenz der Zu- und Abluft etc. bekannt sind. Rechtzeitiges Reagieren ist möglich, ohne dass die Maschine sich selbstständig abschaltet oder sogar wegen eines Schadens repariert werden muss.

Die Veränderungen der Merkmale erfolgen dabei im Sekundenbereich – wenn man bedenkt, dass solche Maschinen normalerweise über Monate problemlos ohne Störungen laufen, dann ist die riesige Datenmenge erklärbar, die bei solchen Sensordaten auftritt. Multiplizieren wir die Daten einer Maschine mit der großen Anzahl der Maschinen in einem Maschinenpark, so wird das Ausmaß noch deutlicher.

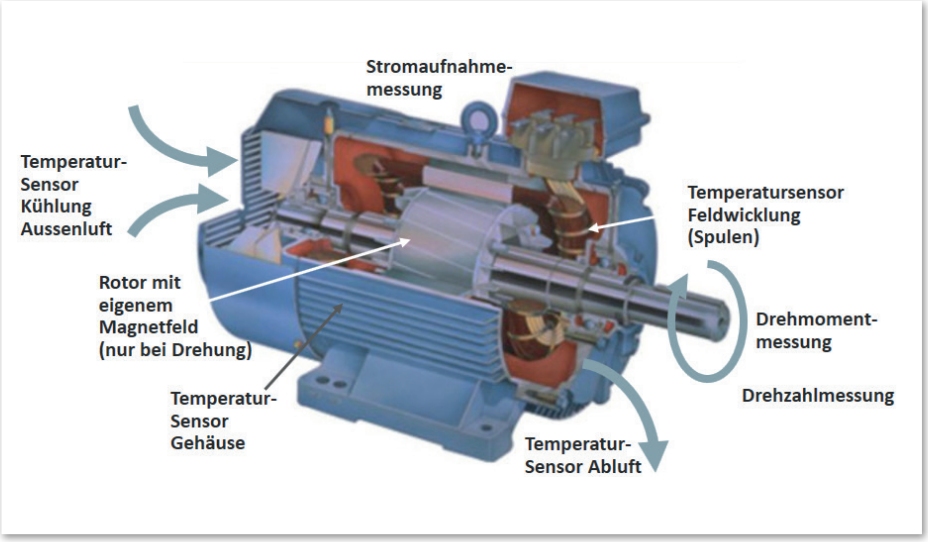


Abbildung 7: Die Sensorik eines Elektromotors

Auch hier gibt es ein Zahlen- und Ressourcen-Szenario. Für das beschriebene Beispiel liegt ein Datensatz mit überschaubaren 17 Variablen vor, für den ein Regressionsmodell in R erstellt werden soll. Wir wiederholen den Test in einer lokalen R-Analyse-Umgebung und in einer Datenbank-Umgebung, die die R-Befehle verarbeiten kann. Wir erhöhen nach und nach die Anzahl der Sensordatensätze bis auf 50 Millionen. In der Praxis könnte das sicherlich ein Vielfaches davon werden und auch die Anzahl der Messwerte pro Datensatz liegen im dreistelligen Bereich.

Das Beispiel untersucht folgenden Zusammenhang mit einer Ziel- und drei determinierenden Variablen: TEMP_ROTOR ~ AMPERE + DREHMOMENT + TEMP_ZULUFT + AMPERE:DREHMOMENT. Zwischen Ampere und Drehmoment liegt eine Abhängigkeit vor. Durch den Analyselauf in R entsteht zusätzlich ein Modellobjekt, das R im Hauptspeicher des Rechners platziert *(siehe Tabelle 2 und 3)*.

Diese Zahlenwerte sind alle auf einem Laptop ermittelt. Die Modell-Erstellung wird sinnvollerweise auf einer größeren Server-Maschine stattfinden und die Anzahl der Sensordaten auf das Zehnfache, wenn nicht sogar auf das Hundertfache steigen. Die Parallelisierung wird beispielsweise auf 64 gesetzt und der In-Memory Column-Store wird 500 GB und mehr betragen. Man kann aufgrund dieser Zahlenwerte davon ausgehen, dass etwa 5 Milliarden Messwertsätze (bei diesem Beispiel mit 17 Variablen) in weniger als zwei Minuten berechnet werden können. Diese Tests bringen drei Erkenntnisse:

- Die Erstellung eines Regressionsmodells erfolgt in der Datenbank effizienter, also

Lokale R-Umgebung ohne Datenbanknutzung			
Anzahl Sätze Sensordatensatz (bei 17 metrische Variablen)	**Größe des Sensordaten-satzes (GB)**	**Modellobjekt im Hauptspeicher (GB)**	**Laufzeit**
6,4 Mill.	0,8384 GB	1,5 GB	6 Minuten
12,9 Mill.	1,6 GB	2,9 GB	14 Minuten
25,8 Mill.	3,3 GB	5,9 GB	30 Minuten
51,7 Mill.	6,5 GB	Abbruch, weil zu wenig Haupt-speicher	---

Tabelle 2

R-Umgebung in Verbindung mit Datenbank / DB-Algorithmen werden R von genutzt					
Anzahl Sätze Sensordatensatz (bei 17 metrische Variablen)	**Größe der Sensor-Tabelle in DB**	**Größe der Sensor-Tabelle in Memory (Column-St)**	**Laufzeit Parallel 1 Ohne In-Memory**	**Laufzeit Parallel 4 Ohne In-Memory**	**Laufzeit Parallel 4 mit In-Memory**
6,4 Mill.	1,2 GB	0,12 GB	40 Sek	21 Sek	8 Sek
12,9 Mill.	2,6 GB	0,35 GB	53 Sek	38 Sek	13 Sek
25,8 Mill.	4,7 GB	0,7 GB	1,7 Min	58 Sek	27 Sek
51,7 Mill.	9,5 GB	1,5 GB	2,9 Min	1,6 Min.	50 Sek

Tabelle 3

ohne dass zusätzliche Ergebnis-Objekte im Hauptspeicher entstehen. Das hängt mit der SQL-basierten Art der Berechnung zusammen.

- Die Regressionsanalyse ist in der Datenbank hoch parallelisierbar. Eine lokale R-Umgebung ist nur „single-thread".
- Sensordaten in einem In-Memory Column-Store können noch schneller verarbeitet werden als Sensordaten im Hauptspeicher einer R-Umgebung. Das hängt mit der Optimierung des Column-Stores zusammen.

Attribute-Importance

Die letzten beiden Beispiele werden nur kurz dargestellt. Wir haben gesehen, dass gerade zu Beginn der Modell-Entwicklung möglichst viele Attribute in die Betrachtung miteinbezogen werden sollten. Nachvollziehbar ist sicher, dass nicht alle Attribute in gleichem Maße Einfluss auf eine Zielvariable ausüben. Um den Vorgang der Wahl der „besten Attribute" zu beschleunigen, verwendet man in einer Datenbank mit eingebetteter Data-Mining-Funktionalität das Verfahren des „Attribute Importance". Dieses arbeitet mit „Minimum Description Length" (MDL) und geht davon aus, dass die knappste und kompakteste Form einer Beschreibung am besten eine Information liefert.

Übertragen auf die Suche nach den „besten Attributen" bedeutet das: Das Verfahren prüft alle potenziell determinierenden Attribute einzeln auf die Wirkung gegenüber der Zielvariablen. Diejenigen Attribute mit dem geringsten Beschreibungsaufwand werden am höchsten bewertet. Ergebnis ist eine nach Einflussstärke sortierte Rangliste aller beteiligten Attribute. Ein solcher Vorgang ist aufgrund der Masse an Prüfungen bei einer Attributanzahl im dreistelligen Bereich und Millionen von Sätzen nur innerhalb einer Datenbank sinnvoll und auch hier nur auf größeren Maschinen.

Ein Test auf das oben schon besprochene Kunden-Beispiel „Affinität für Niedrigpreis-Produkte" liefert Folgendes: Analysiert wird ein zusammenhängendes (Join)-Analyseobjekt, bestehend aus Kunden-, Artikel- und Umsatzdaten mit 134 Variablen und steigender Anzahl von Sätzen (alle Werte gemessen auf einem Vier-Core-Laptop, siehe *Tabelle 4*).

Das Beispiel zeigt: Keine Angst vor großen und komplexen Datenmengen. Man beachte, dass hier jedes Attribut einmal angefasst und dessen Relevanz gegenüber der Zielvariablen gemessen wird. Nebenbei zeigt das Beispiel die Effizienz der In-Memory-Technik in der Datenbank. Die Testwerte sind lediglich auf einem Laptop erzeugt worden. Die Attribute-Importance-Funktion wurde aus einer R-Umgebung in die auf diesem Laptop laufende Datenbank abgesetzt, in der auch die fast 60 Millionen Sätze umfassende Umsatz-/Kunden-Artikeltabelle mit ihren 132 Attributen lag. Die Laufzeitdaten auf einer großen Maschine kann man sich ausmalen. Mit einer Parallelisierung von etwa 64 würde man für dieses Beispiel auf unter zehn Sekunden kommen.

Machine-Learning mit Text

Das letzte Beispiel stammt aus dem Bereich der Text-Analyse. Mit Machine-Learning-Verfahren versuchen wir, auch die Flut an textlicher Information automatisiert zu erfassen. Interessante Beispiele sind unter anderem:

- Die Kategorisierung eingehender Mails, um sie über eine Vorverarbeitung entsprechenden Arbeitsplätzen zuzuordnen
- Klassifizierung von Textdokumenten in Unternehmensarchiven, um diese als Wissensbasis den Mitarbeitern leichter zugänglich zu machen und um Inhaltsgestützte Glossare zu erstellen

Anzahl Sätze in Millionen	Größe des Objektes in der Datenbank	Größe Objekt In-Memory	Attribute-Importance-Laufzeit (Sekunden)
1,7	0,9 GB	0,1 GB	8
3,6	1,6 GB	0,15 GB	10
7,2	3,9 GB	0,3 GB	17
14,4	8,0 GB	0,4 GB	24

Tabelle 4

- Automatisiertes Erfassen von Medienberichten zur Kategorisierung von relevanten Nachrichten
- Kategorisierung von Gesprächsmitschnitten (Kundengespräche, Mitarbeitergespräche etc.)

Das Gemeinsame an diesen Beispielen ist die Zuordnung von Textmaterial zu Kategorien. Diese Kategorien können fachgebietsbezogen sein oder auch einfach nur eine Art von Sentiment ausdrücken, etwa positiv oder negativ. Zur Lösung der Aufgabe nutzt man auch hier einen Naive-Bayes-Algorithmus und das Verfahren gleicht dem, wie wir es bereits oben bei dem Kundenbeispiel beschrieben haben. Nur sind hier in dem Classifier-Objekt Wörter gemeinsam mit ihrer relativen Verwendungshäufigkeit und einer Art Bewertung oder Zuordnung zu einer Kategorie enthalten. Vereinfacht gesprochen besteht der Classifier aus einer gewichteten Wörterliste mit einem Kommentar darüber, ob ein Wort zum Beispiel positiv oder negativ eingestuft wird. Diesen Classifier nutzt man jetzt zur Bewertung von neu eingehendem Textmaterial.

Was hat das jetzt mit großen Datenmengen zu tun? Der Classifier muss natürlich zunächst erstellt werden. Man hat eine Lernphase, in der bereits bekannte und bewertete Texte als Grundlage genommen werden. Um Texte zu bewerten und zu kategorisieren, muss man sie zunächst standardisieren. Dazu durchläuft jeder Text ein Transformationsprozedere, bei dem Groß- in Kleinbuchstaben gewandelt, Zahlen, Leerstellen und die Stoppwörter (inhaltslose Wörter wie der, die, das, und etc.) entfernt werden, man führt das Stemming durch, also das Reduzieren der Wörter auf den sinntragenden Stamm, und man vergleicht Wörter mit einem eigenen Thesaurus, um die Synonymen-/Homonymen-Problematik zu lösen. Diese Textaufbereitung erfolgt einmal, bevor man den Classifier erstellt, und danach auch für jedes weitere neue Dokument, das durch den Classifier bewertet werden soll. Denn alle Dokumente, sowohl die Lern-Dokumente als auch die regelmäßig im Unternehmen neu eingehenden Textmaterialien, sind nach den gleichen Verfahren zu standardisieren, damit man sie überhaupt vergleichen kann. Diesen gesamten Prozess wird man in die Datenbank legen, weil es dort Mechanismen der Parallelisierung gibt. Wenn 100.000 Textdokumente zu bearbeiten sind, dann erfolgt dies bei „parallel 4“ in einem Viertel der Zeit, bei „parallel 16“ in einem Sechzehntel etc.

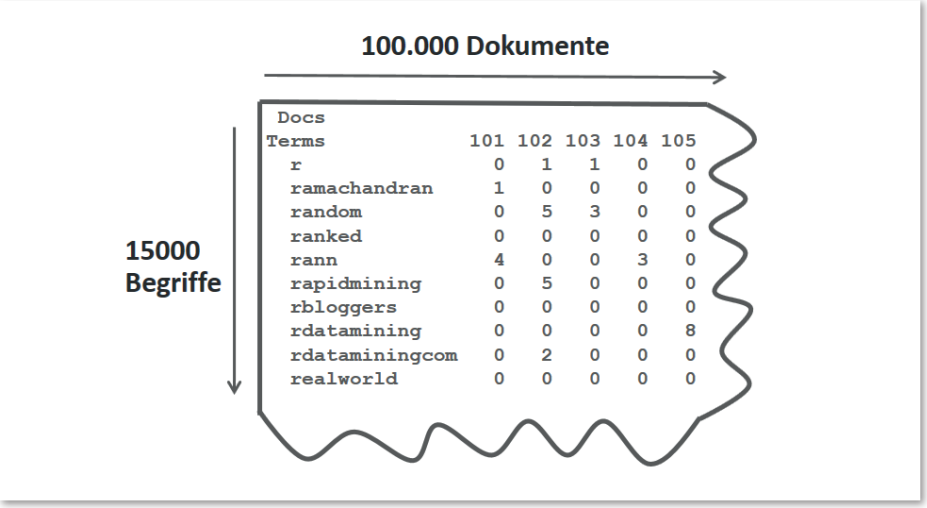


Abbildung 8: Text-Term-Document-Matrix-Objekt zum Herausfinden von gemeinsam besprochenen Themengebieten, tendenziell extrem hoher Speicherbedarf

Hat man kein Wissen über die Inhalte von Texten, also keinen Lerndatenbestand, muss man über die Worthäufigkeiten segmentieren. Hierfür nutzt man sogenannte „Text-Term-Document-Matrix-Objekte“. Das sind Matrizen mit einer Auflistung aller relevanten und sinntragenden Wörter einer Dokumentensammlung und den erfassten Dokumenten selbst. Über diese erfährt man, welche Wörter wie oft in welchen Texten vorkommen. Solche Matrizen können bei Tausenden von Dokumenten und Tausenden von Wörtern extrem groß werden, sodass man auch hier viel Rechenleistung und vor allem Hauptspeicher benötigt *(siehe Abbildung 8)*.

Fazit

Machine-Learning-Analyseprozesse in der kommerziellen IT bringen vor allem zwei Herausforderungen mit sich:

1. Die Größe der Datenbestände
2. Die Komplexität der Analyse-Daten aufgrund der vielen Attribute

Analyse-Umgebungen wie R mit einer Fülle an Machine-Learning-Methoden sind zwar sehr praktisch auf lokalen Laptops einsetzbar, bei den hier besprochenen Datenmengen sind diese jedoch an zentrale Server-Maschinen zu koppeln. Diese dürfen nicht nur einfach eine SQL-Datenbank bereitstellen, sondern sie müssen auch die Möglichkeit für einen Analyseablauf etwa mit R anbieten. Modelle sind dort zu platzieren, wo operative Daten anfallen, in der Datenbank.

Alfred Schlaucher
alfred.schlaucher@oracle.com

Elastisch und skalierbar – Data Lake in der Oracle Cloud

Harald Erb, ORACLE Deutschland B.V. & Co. KG

Vielerorts ist der Umgang mit Big Data und wie man neue Erkenntnisse aus dem Rohstoff Daten zieht, noch ein schwieriges Geschäft.

Zahlreiche neue Datenquellen sind in ihrer Vielfalt, Menge und zeitlichem Aufkommen adäquat entgegenzunehmen und zu einem vertretbaren Preis verlässlich aufzubewahren. Der Datenschatz soll dann aber nicht ungenutzt im Keller (Speicher) lagern, sondern schnell für die aktive Nutzung, also für vielfältigste Analysen bereitgestellt werden können. Das Analyse-Spektrum reicht dabei von der klassischen Self-service-Business-Intelligence-Abfrage bis hin zu Deep Learning, um dringende – vielleicht sogar existenzielle – Geschäftsfragen und Initiativen schnell genug beantworten beziehungsweise umsetzen zu können. Auf Basis der Oracle-Cloud-Plattform lässt sich für solche Zwecke das in diesem Artikel näher beschriebene „New Data Lake“ schnell aufbauen und abhängig vom Analyse- und Ressourcenbedarf anpassen.

Verglichen mit heute war Data Management noch eine überschaubare Aufgabe: Geschäftsanwendungen verarbeiteten in Echtzeit Transaktionen mithilfe hochoptimierter relationaler Datenbanken, deren reibungsloser Betrieb heute noch unternehmenskritisch ist. In das zentrale Enterprise

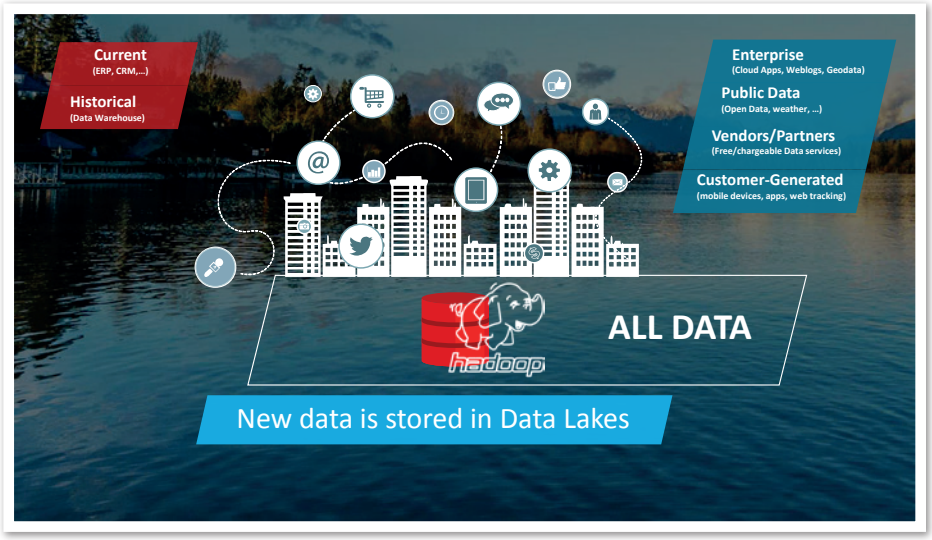


Abbildung 1: Wohin mit den neuen Daten?

Data Warehouse werden laufend ausgewählte Stamm- und Bewegungsdaten aus allen relevanten internen Informationssystemen geladen, die zuvor bereinigt, harmonisiert, angereichert und für den Langzeitdatenbestand historisiert wurden. Diese Vorgehensweise und die qualitativ sehr hochwertigen Daten sind für das Geschäft auch heute noch unverzichtbar (Finanzabschluss, Berichtspflichten, KPI-basierte Steuerung etc.), aber zunehmend nicht mehr ausreichend, wenn man die bisher zu wenig berücksichtigten Datenquellen im blauen Kasten von *Abbildung 1* betrachtet.

Unternehmen, die den Wert ihrer eigenen Daten erkannt haben, überwinden mittlerweile interne Hürden sowie Datensilos und stellen diese unternehmensweit für Experimente (Data Labs, Hackathons) beziehungsweise für die Umsetzung neuer Ideen (neue Services, Geschäftsmodelle) zur Verfügung. Besteht die Chance, bisher nicht realisierbare Projekte bewältigen zu können, kooperieren Unternehmen auch in Form von Joint Ventures miteinander und bringen dazu unter anderem eigene Daten mit. Allgemein bekannt und akzeptiert ist, dass es eine Vielzahl frei zugänglicher Datenquellen gibt (die wir mit Steuergeldern schon bezahlt haben), in Europa beispielsweise von der Europäischen Union, Open-Data-Initiativen bis hinunter auf Städte- und Kreisebene, Forschungsinstitutionen etc. Die Datenbeschaffung ist aufgrund gängiger Datenformate (CSV, XML, JSON) und Schnittstellen einfach, mit der Daten-Qualität und -Vollständigkeit muss man sich dann allerdings selbst helfen. Komfortabler ist es dagegen, auf freie oder kostenpflichtige Datenprovider zurückzugreifen, die „Data as a Service" (DaaS) anbieten, wie die Oracle-Data-Management-Plattform (basierend auf Oracle BlueKai).

Möchte man im Business-to-Consumer-Marketing für eine Kampagnenplanung Internetaktivitäten und digitale Spuren der Konsumenten berücksichtigen, die heute von mehreren Geräten (Computer, Smartphone, Tablet) generiert werden, dann ist die Sammlung, Konsolidierung (möglichst pro Nutzer-ID) und Anreicherung der Daten eine zu bewältigende Herausforderung. Erst danach sind diese externen Informationen in Kombinationen mit eigenen Unternehmensdaten für eine bessere Kunden-Segmentierung und -Ansprache auf den richtigen Kanälen effektiv nutzbar. Es gibt viele weitere Szenarien und Erfolgsstories, die datengetriebene Lösungen und Anwendungen zum Thema haben. Man kann sie grob in drei Kategorien einteilen:

- *Erweiterung bestehender und Erstellen brandneuer Anwendungen*
  Hier muss man nur auf sein eigenes Smartphone schauen oder an die intelligenten Geräte im Haushalt denken und hat schnell beliebig viele Anwendungsbeispiele parat, die insbesondere mit aktuellen Ort- und Zeitinformationen sowie Daten arbeiten, die ihre (im Idealfall identifizierten) Nutzer in großer Menge selbst generieren.

- *Verbesserung der Analytik*
  Hierüber liest man schon viel in Zeitungen oder Fachartikeln, etwa wie Unternehmen mit mehr Detail-Informationen zum Kaufverhalten und besseren mathematischen Modellen erfolgreicher Kaufabschlüsse tätigen, das Risiko der Kundenabwanderung minimieren etc.

- *Erfüllung regulatorischer Vorgaben*
  Einzuhaltende Aufbewahrungspflichten, etwa für Kassenbons durch elektronische Archivierung, werden heute eher so gelöst, dass die Daten einerseits revisionssicher vorliegen und andererseits produktiv genutzt werden können. Die besagten Kassenbons sind gleichzeitig auch die Datengrundlage für Warenkorb-Analysen und helfen, das (sich über die Zeit ändernde) Kaufverhalten der Filialkunden besser zu verstehen.

Das Data-Lake-Konzept

Der interessante Aspekt bei den oben skizzierten Beispielen ist die Fähigkeit, bei Bedarf in kurzer Zeit unternehmenseigene Daten mit den neuen Daten kombinieren, analysieren oder anderweitig produktiv verwerten zu können. Ein modernes Data-Lake-Konzept muss unter Berücksichtigung betrieblicher Rahmenbedingungen genau diese Anforderung unterstützen und nicht nur für die kostengünstige Ablage neuer Daten (im Sinne einer erweiterten Staging Area für das existierende Data Warehouse) zuständig sein. *Abbildung 2* hebt daher als wesentliche Funktionsbereiche die agile Datenaufbereitung („Prepare") und vielfältige Analysefähigkeiten („Analyze") samt Data-Lake-Anwender („Data Consumers") besonders hervor.

Im Vergleich zum klassischen Data Warehouse werden in einem Data Lake die eingehenden Daten nicht mit komplexen Datenqualitäts- und Integrationsverfahren in definierte Strukturen überführt, sondern direkt in ihrer Ursprungsform abgelegt. Damit können beliebige Daten schnell und einfach für Analysen nutzbar gemacht und beliebig verknüpft werden. Manchmal ist es aber auch erforderlich, Daten schon vor dem Speichern zu analysieren, etwa um Echtzeit-Anforderungen (Warnung bei bestimmten Zustands-Informationen) bis zum vollautomatischen Prozess umzusetzen oder wenn sich eine vollumfängliche Speicherung technologisch/wirtschaftlich nicht rechtfertigen lässt oder eine Vorverdichtung stattfinden soll.

Als Schlüsseltechnologie für Big Data wurde in den letzten Jahren üblicherweise das Open-Source-Software-Framework Hadoop angesehen, weil damit beliebige

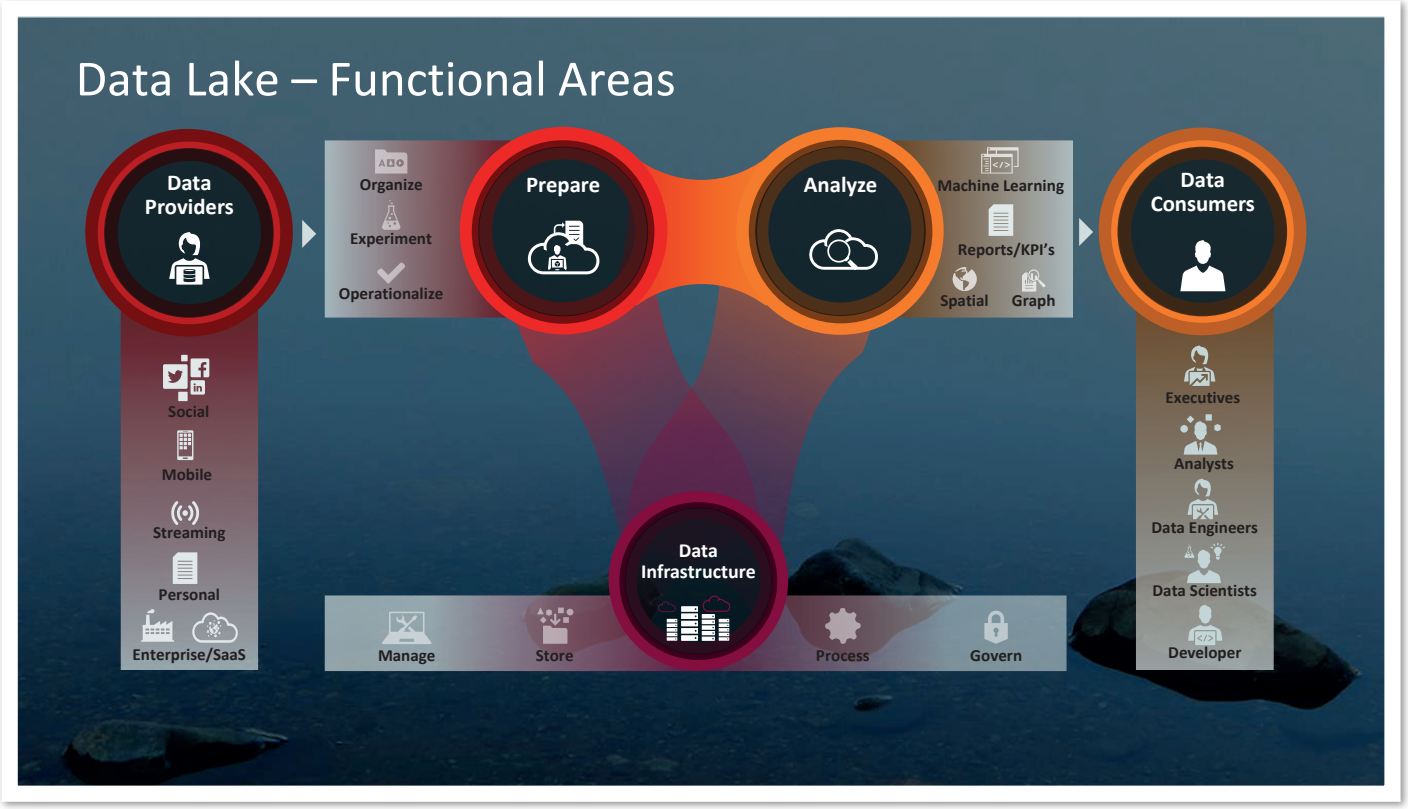


Abbildung 2: Funktionsbereiche eines Cloud-basierten Data Lake

Datenarten in großer Menge verarbeitet und Berechnungen über viele Knoten eines Clusters verteilt werden können. Die große Idee hinter Hadoop ist, die Daten mithilfe des Hadoop Distributed File System (HDFS) zu verteilen und die Analysen (etwa mit Map Reduce) direkt bei den Daten durchzuführen. Datenspeicherung („Storage") und Rechenleistung („Compute") wurden dabei zusammengeführt mit dem Vorteil, dass man bei Analysen mit sehr großen Datenmengen das Bewegen eben dieser Daten vermeidet. Aber selbst bei frühen Data-Lake-Konzepten erstreckte sich die Datenspeicherung durchaus über mehrere Data Stores („Repositories"), vor allem war aber von Hadoop, relationalen Technologien und NoSQL-Datenhaltungen die Rede. In diesem Kontext bietet Oracle seit dem Jahr 2011 mit der Oracle Big Data Appliance ein Engineered System an, das unter anderem die Cloudera Distribution of Hadoop (CDH) und weitere auf Oracle-Technologie abgestimmte Software-Komponenten enthält und zum Aufbau eines Data Lake eingesetzt werden kann.

Abbildung 3 zeigt das Konzept eines Data Lake. Entgegengenommene Rohdaten („Raw Data") werden dort teilweise aufbereitet, um Fachanwendern die Analysen zu erleichtern. Dabei kommen Verfahren zur Datenqualitätssicherung (Datenbereinigung, -harmonisierung, Abgleich mit/Verwendung von Referenzdaten) und Datenanreicherung (wie Geocodierung) zum Einsatz. Diese aufbereiteten Daten („Curated & Transformed Data") werden dann oft einer großen Gruppe von Anwendern bereitgestellt, die bevorzugt über eine SQL-Schnittstelle und Abfrage-Werkzeuge auf die neuen Datasets zugreifen. Data Lakes adressieren üblicherweise eher explorative Anwendungsfälle, bei denen neue Fragestellungen häufig in Discovery Labs (oder Data Labs) unter Einbeziehung bisher nicht genutzter Daten untersucht und im Erfolgsfall in die produktive IT-Infrastruktur überführt werden.

Der aufbereitete Teil des Data Lake ist von seinem Konzept her nicht weit von dem des Data Warehouse entfernt. Ziel ist vor allem aber die schnellere Umsetzung neuer Anforderungen, um auf dynamische Veränderungen im Geschäftsumfeld schnell reagieren zu können. Die so gewonnene Agilität ist allerdings eine große Herausforderung aus Governance-Sicht. Diese beschränkt sich dabei nicht nur auf die allgemeine Zugriffssicherheit, sondern umfasst auch Aspekte wie Nachvollziehbarkeit der Datenflüsse, Dokumentation der Dateninhalte und Interpretationen (Data Catalog) oder aber auch die Maskierung von Daten für bestimmte Benutzergruppen bis hin zur Einrichtung eines Zonenkonzepts für unterschiedlich eingestufte Datenklassen. Effektive Governance erfordert daher einen ganzheitlichen Ansatz über den gesamten Prozess und Technologiegrenzen hinweg, um ein komplettes Bild über den Datenschatz zu erhalten.

Aufbau eines Data Lake in der Oracle Cloud

In den letzten Jahren ist die Diversifizierung im Bereich der Technologien eher gestiegen. Beispiele dafür sind Graph-Datenbanken, um stark vernetzte Informationen abzuspeichern, darzustellen und analysieren zu können (Fragestellung: „Wer sind die Meinungsmacher zu aktuellen Themen in den sozialen Netzwerken?").

Ein erkennbarer Trend der letzten Jahre ist auch, dass Object Storage und Apache Spark Hadoop zunehmend Konkurrenz machen, da die feste Verbindung von Compute und Storage Einschänkungen in Bezug auf flexible Skalierung und Elastizität mit sich bringt. Dagegen würde eine Trennung von Storage und Compute das dynamische Hinzu¬fügen von Rechenkapazität für sehr komplexe Berechnungen ermöglichen. Genau diesen Aspekt adressiert Object Storage. Er ist sehr einfach administrierbar,

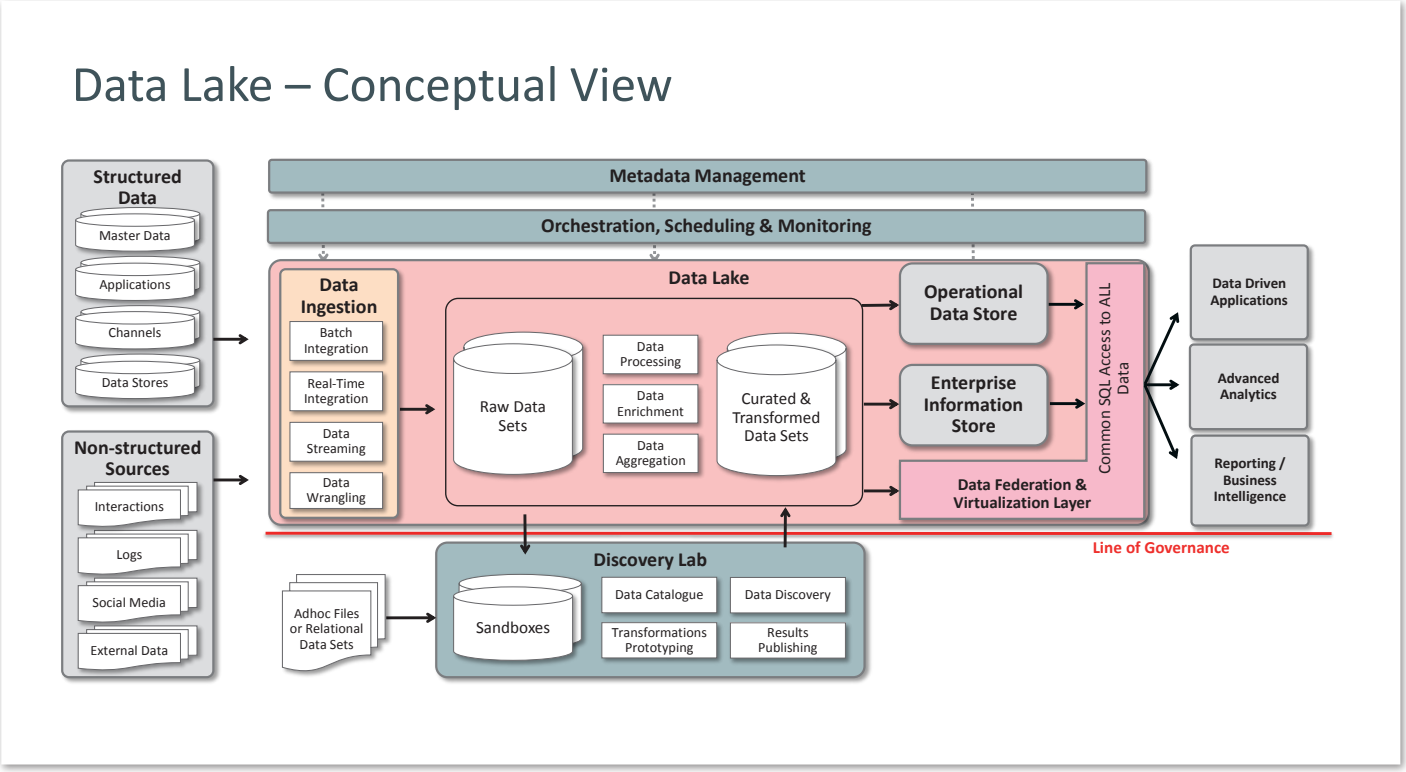


Abbildung 3: Data-Lake-Konzeptansicht

gleichzeitig sehr kostengünstig sowie skalierbar und ermöglicht das effiziente Speichern auch sehr großer Datenmengen. In Kombination mit Apache Spark, das als Framework für Cluster Computing für die Parallelisierung der Berechnung sorgt, lassen sich damit sehr gut komplexe analytische Berechnungen durchführen. Dabei müssen die Daten allerdings in die entsprechenden Spark-Compute-Knoten geladen und somit bewegt werden. Da aber häufig nur kleinere Datenausschnitte betrachtet werden, hat dies oftmals keine negativen Performance-Auswirkungen und wird durch die In-Memory-Verarbeitung von Spark überkompensiert.

Wer über Flexibilität bei der Nutzung von Speicherplatz, Rechenleistung oder die benötigte Anwendungssoftware für die Daten-Aufbereitung und -Analyse nachdenkt, wägt sicher auch den Aufbau einer eigenen IT-Infrastruktur gegen das Beziehen von Cloud Computing Services ab und sollte sich unbedingt das modulare Angebot in der Oracle

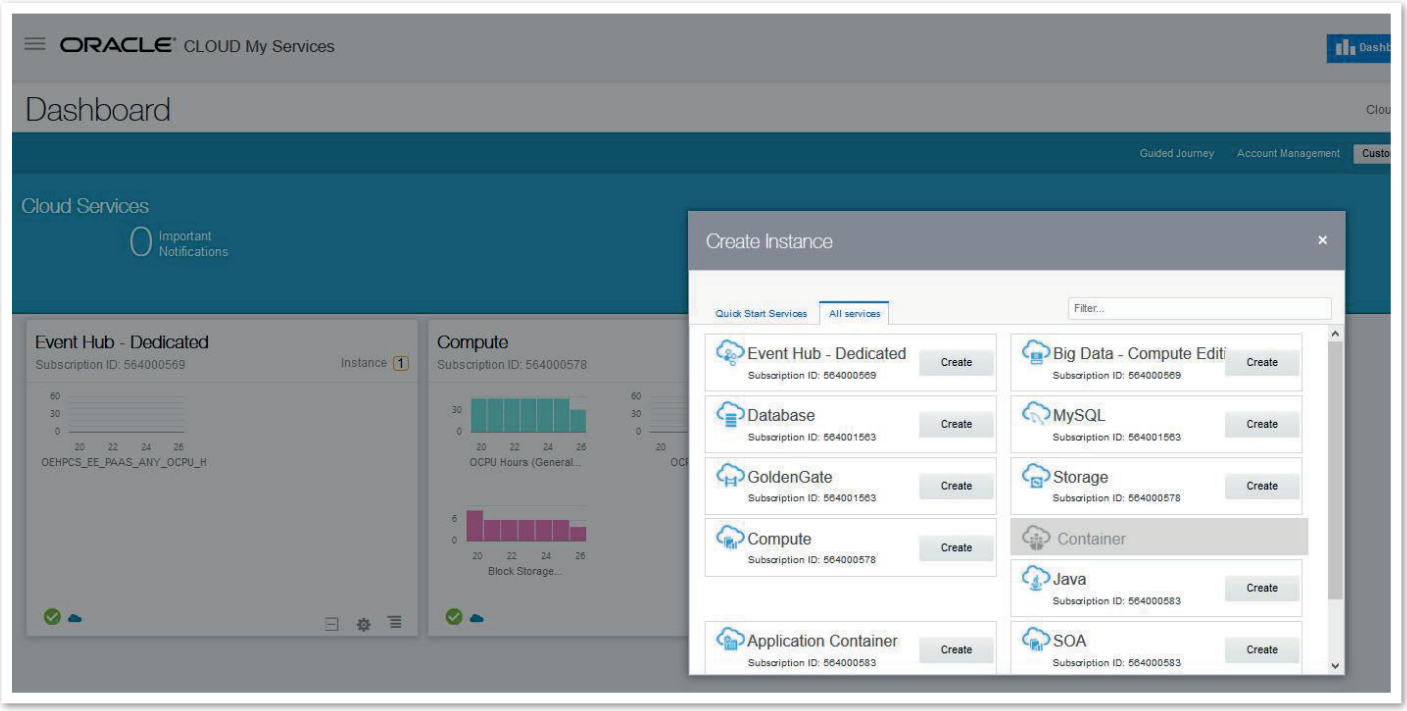


Abbildung 4: Service-Konsole zur Verwaltung der Oracle Cloud Services

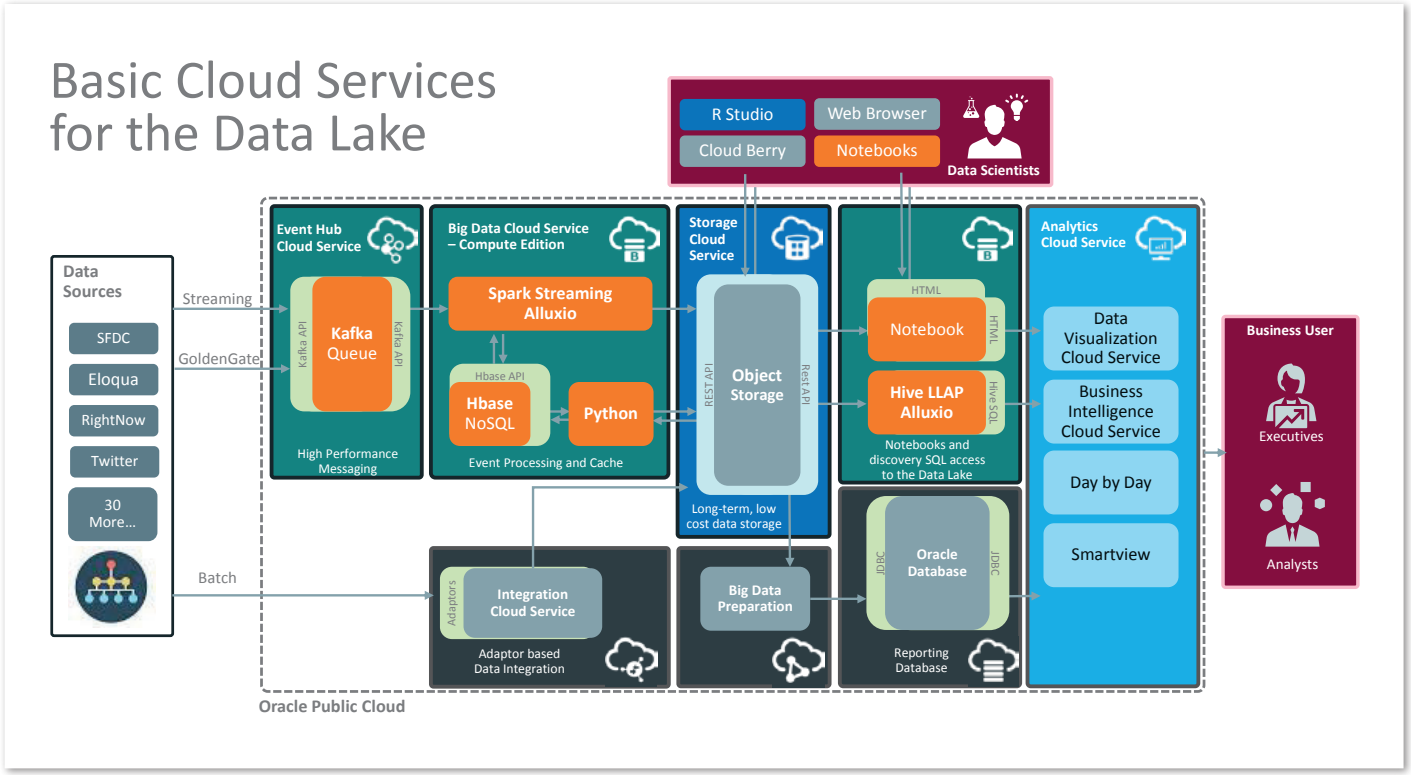


Abbildung 5: Event Hub, Big Data: Compute Edition und Storage Cloud Services bilden die Basis für den Aufbau eines Cloud Data Lake

Cloud näher ansehen. *Abbildung 4* zeigt, wie sich die erforderlichen Oracle Cloud Services für den Aufbau eines Data Lake auswählen lassen, um die eben besprochenen Big-Data-Schlüsseltechno¬logien Hadoop, Spark etc. zusammen mit den Vorteilen von Object Storage zu nutzen:

- *Oracle Big Data Cloud Service – Compute Edition*
  Mit diesem Service können Big Data Cluster innerhalb weniger Minuten bereitgestellt, bei Bedarf um weitere Rechenkapazität erweitert, aber auch wieder verkleinert und als Datendrehscheibe beziehungsweise Processing Engine mit anderen Cloud Services gekoppelt werden. Ausgefallene Cluster-Komponenten und -Knoten werden im Rahmen der kontinuierlichen Überwachung ohne menschliches Eingreifen automatisch korrigiert. Administrationsaufgaben sind variabel über grafische Benutzeroberflächen oder per Maschine-zu-Maschine-Kommunikation über REST-APIs durchführbar.

- *Oracle Event Hub Cloud Service*
  Eine von Oracle verwaltete Streaming-Plattform, die mit Apache Kafka eine weitere Big-Data-Schlüsseltechnologie verwendet. Vor allem, wenn es um die Verarbeitung von Protokolldaten geht, etwa basierend auf Aktivitäten in sozialen Netzwerken oder Sensordaten, werden Messaging-Systeme beim Sammeln, Analysieren und Verteilen dieser Datenströme vor große Herausforderungen gestellt. Apache Kafka besticht in diesem Feld vor allem durch sehr hohen Datendurchsatz (mehrere Tausend Nachrichten pro Sekunde). Kafka-Komponenten verbinden Arbeitsspeicher, Cache von Speichersystemen sowie die Speicherverwaltung des lokalen Betriebssystems miteinander und sind im Cluster-Betrieb auf mehrere Rechnerknoten verteilbar, sodass eine effiziente Verteilung der Rechen- und Speicheraufgaben ermöglicht wird.

- *Oracle Storage Cloud Service*
  Neben anderen Speicher-Services (etwa für Backup und Archivierung) bietet Oracle den besagten „Object Storage" für die sehr preiswerte und flexible Speicherung beliebiger Datasets, also für strukturierte und unstrukturierte Daten an. Im Gegensatz zu den bekannten Dateisystemen enthalten die Objekte zwar die Daten, sind allerdings nicht in einer Hierarchie organisiert. Jedes Objekt befindet sich auf der gleichen Ebene eines Adressraums, wird mithilfe seiner erweiterten Metadaten charakterisiert und bekommt einen einzigartigen Identifikator zugewiesen. Somit können Server oder Endanwender das Objekt beziehen und müssen den physischen Standort der Daten nicht kennen. Diese Herangehensweise ist für die Automatisierung und Rationalisierung der Datenspeicherung in Cloud-Computing-Umgebungen nützlich und darüber hinaus auch preisgünstiger.

Aus Anwendersicht haben Entwickler, Data Engineers und Data Scientists damit bereits die relevanten Cloud Services, die sie zum Arbeiten und Analysieren benötigen. Diese Personengruppen werden dabei feststellen, dass Oracle auch bei der Ausgestaltung seiner Cloud Services eine lange Tradition fortsetzt und weiterhin Open-Source-Technologien unterstützt beziehungsweise ergänzend nutzt. In *Abbildung 5* sind dafür stellvertretend einige Werkzeuge eingetragen: CloudBerry (ein komfortabler Windows-Client für Dateimanagement und -transfer); für Entwicklung und Analyse kommen webbasierte Notebooks (Jupyter, Zeppelin) mit einbindbaren Interpretern zum Einsatz, die zahlreiche Programmier- und Skriptsprachen unterstützen; RStudio für die Arbeit mit der Statistik-Programmiersprache R.

Statt leicht bedienbarer Benutzeroberflächen ist es für diese Klientel wichtiger,

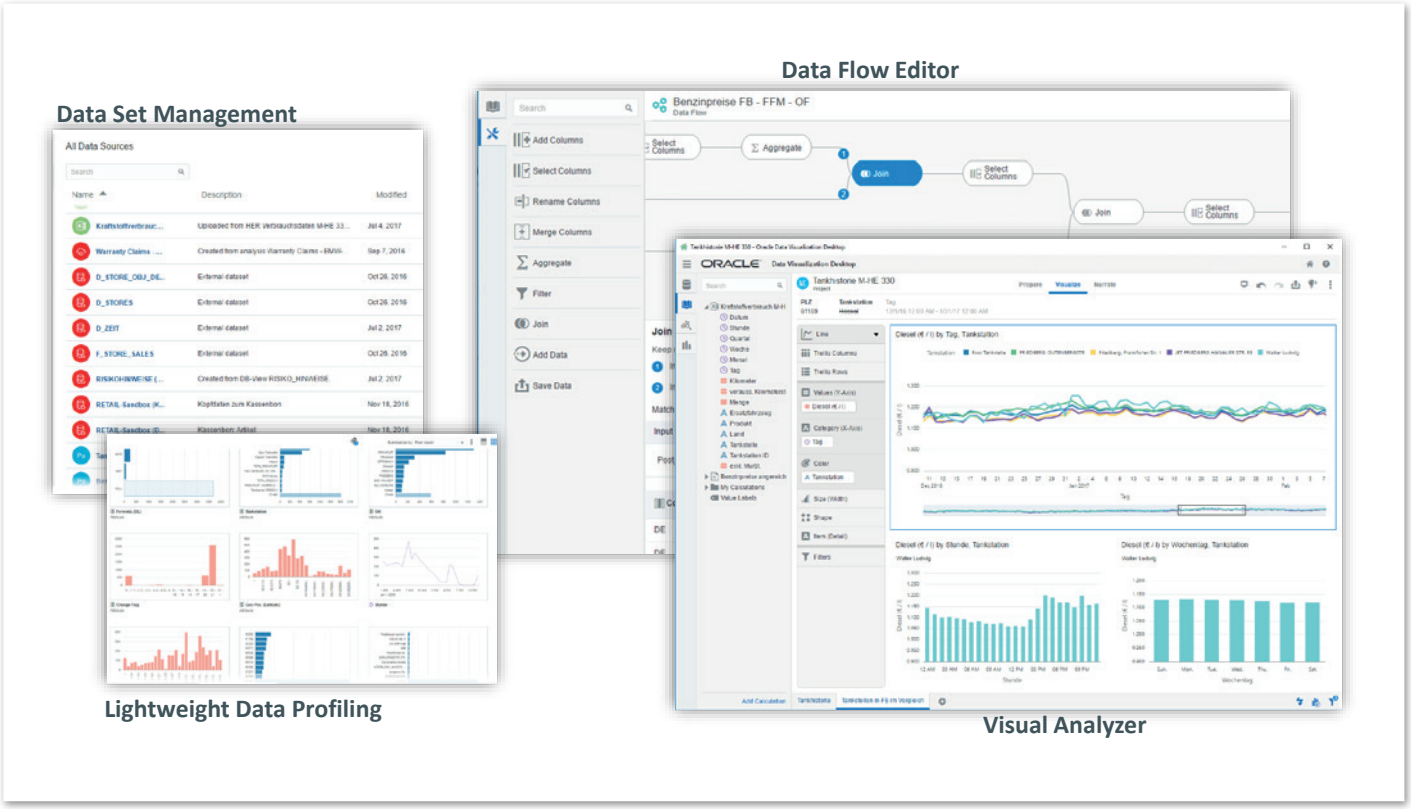


Abbildung 6: Daten-Exploration und visuelle Analyse in der Oracle Analytics Cloud

möglichst schnell neue Technologieversionen (Spark, Hive etc.), Programmpakete (insbesondere für Python und R) oder populäre Frameworks (wie TensorFlow für Deep Learning inklusive Grafikprozessor-Support) auf der Big-Data-Plattform einsetzen zu können.

Die heutigen Anforderungen an die Data-Lake-Anwender sind durchaus höher im Vergleich zu den früheren Business-Intelligence-Anwendern. Oftmals sind Programmier- und Skripting-Kenntnisse erforderlich, wenn bei Datenexperimenten im Discovery Lab oder bei der direkten Verwertung neuer Datenquellen nicht auf die formalen IT-Entwicklungsprozesse gewartet werden kann. Um dem zu begegnen, ist für die Fachanwender erfreulicherweise ein Trend zu benutzerfreundlichen Werkzeugen für alle Aspekte der Wertschöpfungskette vom Laden der Daten bis zum Analysieren erkennbar. Entsprechend bietet auch Oracle für die Arbeit mit dem Data Lake unter anderem Daten-Integration sowie Big Data Preparation Cloud Services an und hat Reporting-, Data-Discovery- sowie Analytics-Funktionen in einem Cloud Service gebündelt.

Analytics Cloud Service ist ein interaktiver Cloud Service für das Data Lake und die klassische rollenbasierte Informationsversorgung des Unternehmens durch Business-Intelligence-Dashboards, Reports, proaktive Alarme etc. Für die ambitionierten Fachanwender ist besonders das Werkzeug „Oracle Data Visualization" interessant, das mithilfe einer grafischen Oberfläche Daten aus einer Vielzahl von Quellsystemen laden beziehungsweise live abfragen kann. Da die Ursprungsdaten nicht immer 100-prozentig perfekt für die geplanten Analysen sind, fällt den Anwendern damit nun die eigenständige Durchführung der Datenaufbereitung zu. Mit den integrierten „Lightweight-ETL"-Funktionen ist dies jedoch ohne Programmierkenntnisse möglich. *Abbildung 6* gibt einen Eindruck davon, wie sich Datasets organisieren, vor der Analyse inspizieren („Lightwight Data Profiling") und in Data Flows kombinieren lassen. Durch die einfache Bedienung und enge Integration aller Funktionen können Fachbenutzer ihre Analysen innerhalb einer Anwendung von der Idee über mehrere Iterationen hinweg bis zum Endergebnis durchführen, Analyse-Schritte dokumentieren und für Live-Präsentationen aufbereiten.

Fazit

Ein Data Lake ist mehr als nur Hadoop und ein spannendes Thema – wie die rasanten neuen Möglichkeiten und technologischen Entwicklungen zeigen. Es wird das Data-Warehouse-Konzept nicht verdrängen, sondern in den meisten Fällen ergänzen. Mit Data Lakes können viele Anwendergruppen dynamisch und agil arbeiten, komplexe analytische Aufgaben-stellungen sind mit dieser Infrastruktur lösbar. Die Oracle-Cloud-Umgebungen führen im Big-Data-Analytics-Umfeld alle benötigten Fähigkeiten zusammen und bieten beste Bedingungen für neue Entwicklungen bei gleichzeitiger Integration in die bestehende Systemlandschaft.

Harald Erb
harald.erb@oracle.com

Jetzt **Ticket** sichern

mit eigenem Applications-Stream

PROGRAMM ONLINE

mit rund 450 Vorträgen

2017.doag.org

Eventpartner: AOUG Austrian Oracle User Group · SOUG swiss oracle user group · iJUG Verbund · ORACLE