RODM - Statistische Datenanalyse mit R und Oracle Data Mining

Prof. Dr. R. von Schwerin, F. Langenbruch, J. Bellan

Hochschule Ulm
Institut für Informatik
Fachgebiet: Betriebliche Informationssysteme

Technik
Informatik & Medien
Hochschule Ulm
University of
Applied Sciences

ORACLE ACADEMY

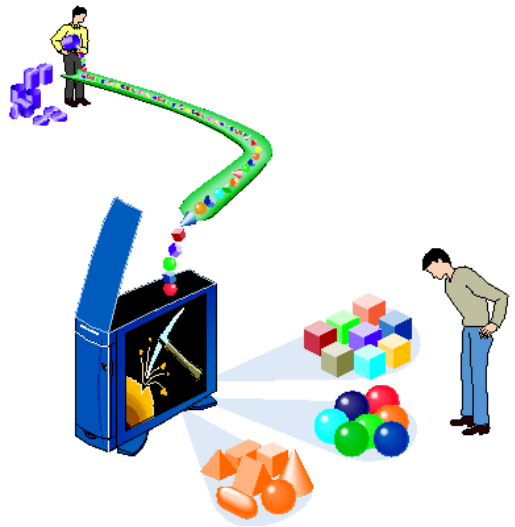
HS DOAG HOCHSCHUL-COMMUNITY

17. November 2010

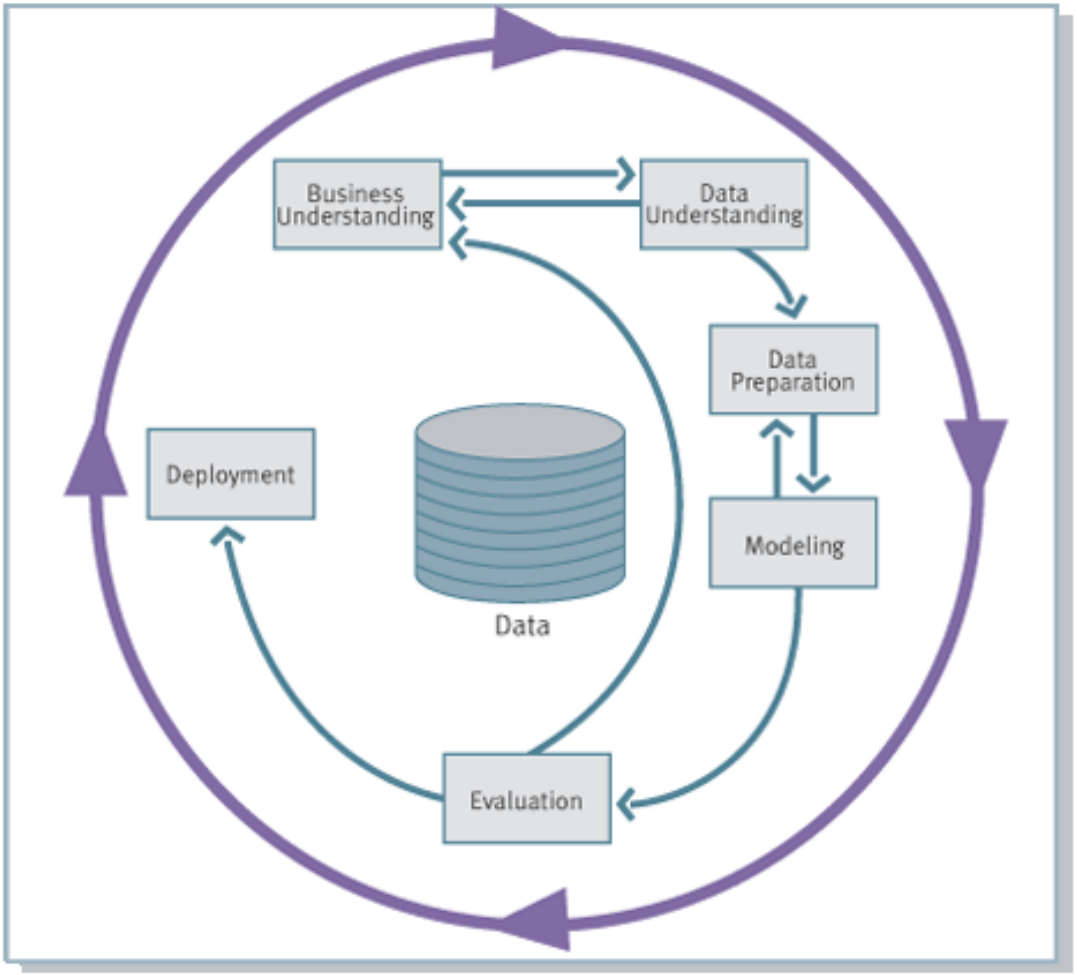
Einführung

Aufbau des Vortrags

- Phasen des *CRISP-DM* Vorgehensmodell
- Praxisnahes Beispiel: Aufgabe des **Data Mining Cups 2010** von der *prudsys AG*
  - Intelligentes Couponing
  - Umsatzsteigerung durch Kaufanreize mit Gutscheinen



CRoss-Industry Standard Process for Data-Mining



Quelle: http://www.crisp-dm.org/

Aufgabenstellung des DMC 2010

Business Understanding

- Umsatzsteigerung eines Online-Shops
- Schaffung eines Kaufanreizes durch Versenden eines Coupons
- Identifikation von Kunden, bei denen erneute Käufe innerhalb der nächsten 90 Tage nicht zu erwarten sind
- Erwartung: 10% der Kaufanreize führen zu einem Bestellwert von 20 Euro
- Vorsicht: Coupons, die von Stammkunden eingelöst werden, führen zu einem Verlust von 5 Euro

		Real	
		Non-repurchasers (0)	Repurchasers (1)
Forecast	No voucher (0)	0	0
	Voucher (1)	1.5	-5

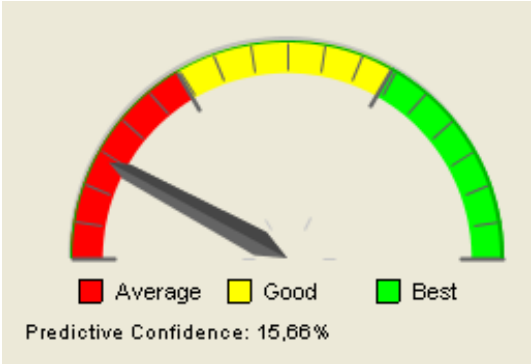
Aufgabenstellung des DMC 2010

Data Understanding

- Trainings- und Klassifizierungsdaten
- Daten repräsentieren Erstbestellungen der Kunden
- Target90 in den Trainingsdaten zeigt an, ob der Kunde innerhalb der folgenden 90 Tage erneut eingekauft hat

CUSTOMERNUMBER	FIRSTORDERDATE	SALUTATION	TITLE	DOMAIN	DATECREATED	NEWSLETTER	MODEL	PAYMENTTYPE	DELIVERYTYPE	...	TARGET90
28146	20.07.08	0	0	4	16.07.08	1	1	1	1	...	0
16549	11.10.08	1	0	12	11.10.08	0	1	0	0	...	0
26112	25.08.08	1	0	4	25.08.08	0	1	2	0	...	0
31311	03.07.08	0	0	2	03.07.08	0	2	1	1	...	0
7649	03.08.08	0	0	5	03.08.08	0	3	1	1	...	0
2044	20.11.08	0	0	11	20.11.08	1	3	2	0	...	0
44682	05.10.08	1	0	4	05.10.08	1	3	1	1	...	0
27123	27.10.08	1	0	12	27.10.08	0	1	2	0	...	0
7601	12.12.08	1	0	12	12.12.08	0	1	2	0	...	0
20662	20.07.08	1	0	12	20.07.08	0	1	3	0	...	1
47287	18.11.08	1	0	5	18.11.08	0	2	2	0	...	0
54338	13.10.08	0	0	12	13.10.08	1	1	3	0	...	0
38473	21.12.08	1	0	6	21.12.08	0	2	3	0	...	0

Erste Klassifizierungsversuche



Notwendigkeit einer Datenanalyse und -vorbereitung

- Qualität der ersten Modelle nicht zufriedenstellend
- Vermutung: Datenqualität noch zu schlecht, um gute Vorhersagen zu treffen
- Abhilfe: intensives Data Profiling (Phasen: Data Understanding und Preparation)

Einführung

Was ist R-ODM?

- Zusatzpaket für die freie Programmiersprache für Statistik “R”
- In R implementierte Befehle unterstützen den Benutzer bei statistischem Rechnen
- R-ODM bietet die Möglichkeit, Data Mining Funktionen der Oracle Datenbank zu verwenden
- Modellerstellung läuft in Datenbank ab, Analyse schließlich in einer R-Umgebung

Einführung

Installation und Verwendung von R-ODM

- R selbst wird vorausgesetzt
- Grafische Oberflächen wie z.B. Tinn-R können hilfreich sein
- Notwendige Pakete in R:
  - RODBC - Zugriff auf Datenbanken mit Hilfe von ODBC
  - RODM - R-Funktionen, mit denen direkt auf das Package DBMS_DATA_MINING in der Datenbank zugreifen
- Notwendige Vorbereitungen:
  - Oracle Instant Client installieren
  - Datei TNSNAMES.ORA erstellen und konfigurieren
  - ODBC Data Source Name (DSN) im Betriebssystem konfigurieren
  - Wichtig: 32-bit oder 64-bit beachten

Verbindung mit Oracle Datenbank herstellen

Notwendige Pakete laden

```
library(RODM)
library(RODBC)
# RODBC wird automatisch mit Paket RODM
    geladen
```

Verbindung herstellen und abspeichern

```
DB = RODM_open_dbms_connection( dsn="rodm",
    uid="<db_user>", pwd="<password>" )
```

Anfängliche Probleme

Spracheinstellungen der Datenbank

- R-Funktionen, die gebrochene Zahlen als Aufrufparameter erfordern, schlugen fehl
- **Grund:** deutsche Spracheinstellungen der Datenbank
  - Komma anstatt Punkt als Trennzeichen
- Im Internet finden sich ähnliche Berichte zur Java oder PL/SQL API
- **Lösung:** Parameter NLS_NUMERIC_CHARACTERS auf den Wert "." setzen
- Alternativ: NLS_LANGUAGE ändern

Datenanalyse

Besseres Datenverständnis mit R

- Beschreibung der Daten mit statistischen Lage- und Streuungsmaße
- Erforschung der Daten mit visuellen Darstellungsmethoden
- Untersuchung von Korrelationen innerhalb der Daten

Datenbereitstellung mit RODBC

Data Frames

- Daten werden in R in Data Frames abgelegt
- Abfrage von Daten aus der Datenbank mit Funktionen des RODBC-Pakets möglich
- Ablegen von Daten in Data Frames mit Zuweisungsoperator <-
- Darstellung der ersten Zeilen mit Spaltennamen mit HEAD()

sqlQuery Methode

```
query <- (select * from myTable)
result <- sqlQuery(DB,query)
head(result)
```

Beschreibung von Daten mit st. Methoden

Statistikfunktionen

- R bietet eine Vielzahl von Statistikfunktionen
- Auch mit DBMS_STAT_FUNCS der Oracle Datenbank möglich
- Erster Überblick über die Daten mit STR() und SUMMARY()

```
> str(result)
'data.frame':   32396 obs. of  45 variables:
 $ CANCEL_ALL           : int  0 0 0 0 0 0 0 0 0 0 ...
 $ W6                   : int  0 0 0 0 0 0 0 0 0 0 ...
 $ W3                   : int  0 0 0 0 0 0 0 0 0 0 ...
 $ W10                  : int  0 0 1 0 0 0 0 0 0 0 ...
 $ MODEL                : int  1 1 1 1 2 2 1 1 3 2 ...
 $ DELIVPOSTCODE        : Factor w/ 100 levels "0","00",
 $ W5                   : int  0 0 0 0 0 0 3 0 0 0 ...
 $ W9                   : int  0 0 0 0 0 0 0 2 0 0 ...
 $ ENTRY                : int  0 0 0 0 1 1 0 0 1 1 ...
 $ NUMBERITEMS          : int  1 2 2 2 1 1 3 2 2 1 ...
 $ PAYMENTTYPE          : int  0 2 1 0 0 2 0 0 1 1 ...
 $ SHIPPINGDAYS_PROMISED: int  2 1 1 3 2 2 1 13 1 2 ...
 $ SHIP_PROM_TO_REAL    : int  -1 0 0 -3 0 -1 0 10 0 -1
 $ ADVERTISINGCODE_CLASS: int  0 1 0 1 0 0 1 1 0 0 ...
 $ REMI                 : int  0 0 0 0 0 0 0 0 0 0 ...
```

```
> summary(result)
   CANCEL_ALL             W6
 Min.   :0.00000   Min.   : 0.00000
 1st Qu.:0.00000   1st Qu.: 0.00000
 Median :0.00000   Median : 0.00000
 Mean   :0.02954   Mean   : 0.02794
 3rd Qu.:0.00000   3rd Qu.: 0.00000
 Max.   :1.00000   Max.   :27.00000

 SHIP_PROM_TO_REAL  ADVERTISINGCODE
 Min.   :-276.000   Min.   :0.0000
 1st Qu.:  -2.000   1st Qu.:0.0000
 Median :  -1.000   Median :0.0000
 Mean   :  -2.411   Mean   :0.2013
 3rd Qu.:   0.000   3rd Qu.:0.0000
 Max.   : 343.000   Max.   :1.0000
 NA's   : 972.000
```

Beschreibung der Häufigkeit von Merkmalen mit R

Häufigkeitstabelle

```
absolut <- table(result$REMI)
```

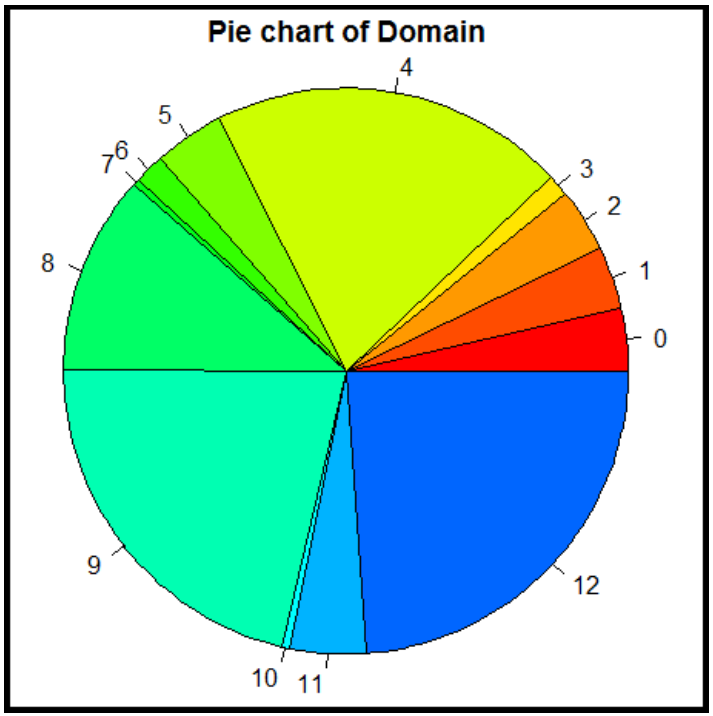
Relative Häufigkeit

```
relativ <- absolut / sum(absolut)
#relative Häufigkeit
prozent <- relativ * 100; round(prozent, 1)
#relative Häufigkeit gerundet
```

Visualisierung von Häufigkeiten in R

Piechart

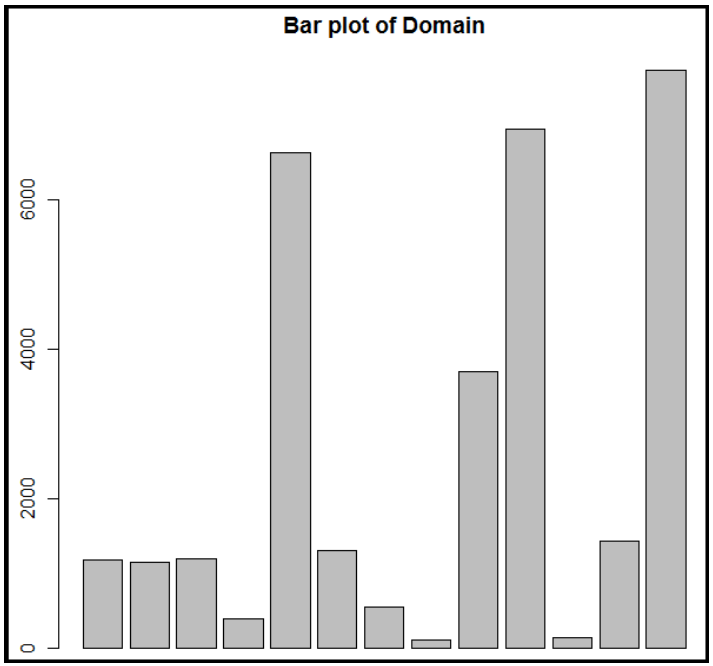
```
pie(table(result$DOMAIN),main="Häufigkeiten
    von Domain", col=rainbow(20))
```



Visualisierung von Häufigkeiten in R

Bar Plot

```
barplot(table(result$DOMAIN),
    main = "Häufigkeiten von Domain")
```



Beschreibung der Daten mit Streuungsmaße

Sortierung und Rang

```
sort(result$DOMAIN)
rank(result$DOMAIN)
```

Streuung

```
MA <- mean(abs(result$DOMAIN - median(data$
   DOMAIN)));#Abweichung vom Median
D <- mad(result$DOMAIN, const=1); #Median-
   Deviation mit mad()
```

Visualisierung von metrischen Daten

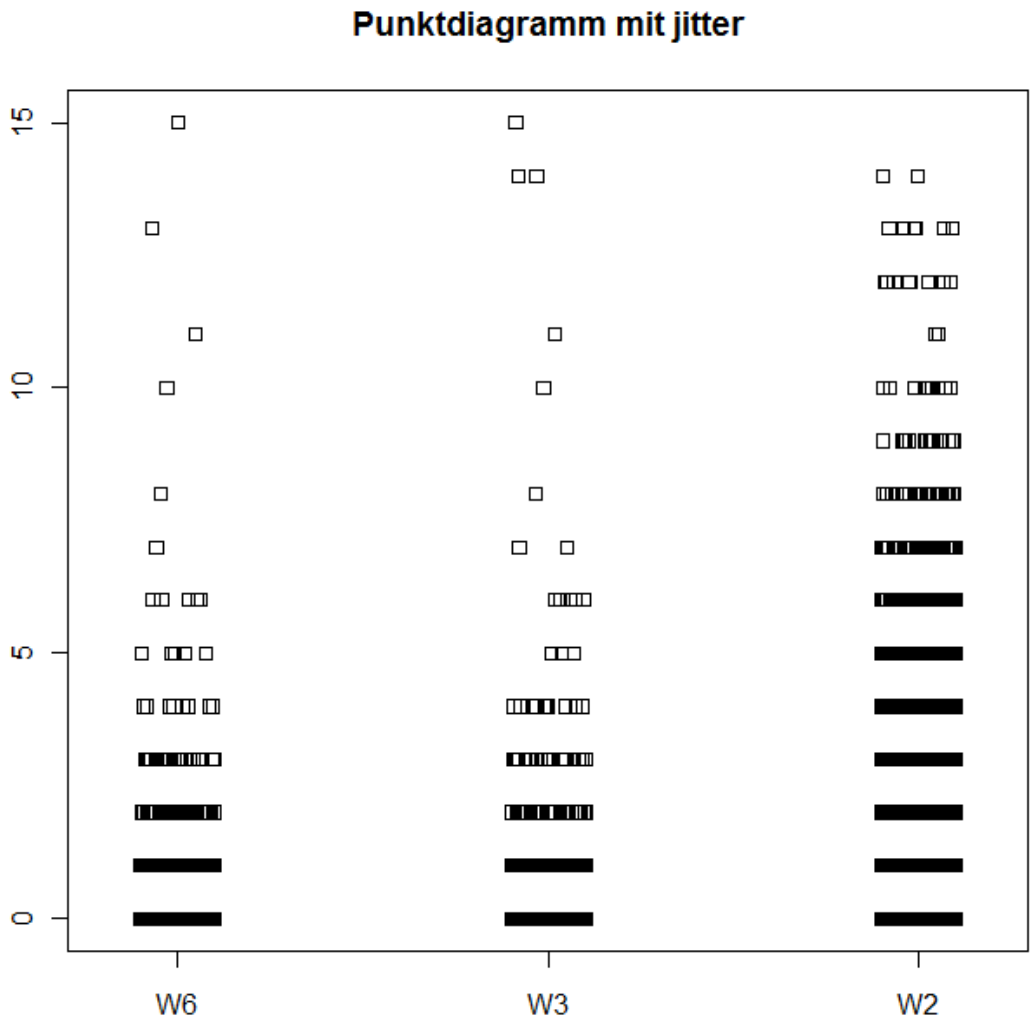
Punktdiagramm

- Darstellung von ordinal skalierten und metrischen Daten
- Y-Achse: Beobachtungen als Punkte
- X-Achse: Unterteilung von mehreren Merkmalen
- Darstellung der Häufigkeit mit „jitter“ (gleiche Werte werden auf gleicher Höhe nebeneinander angeordnet)

Dot-Plot mit „jitter“

```
stripchart(list(result$W6,result$W3,result$W2)
   ,vertical=TRUE,method="jitter",jitter=0.1,
   group.names=c("W6","W3","W2"),ylim=c(0,15),
   main="Punktdiagramm mit jitter")
```

Visualisierung von metrischen Daten

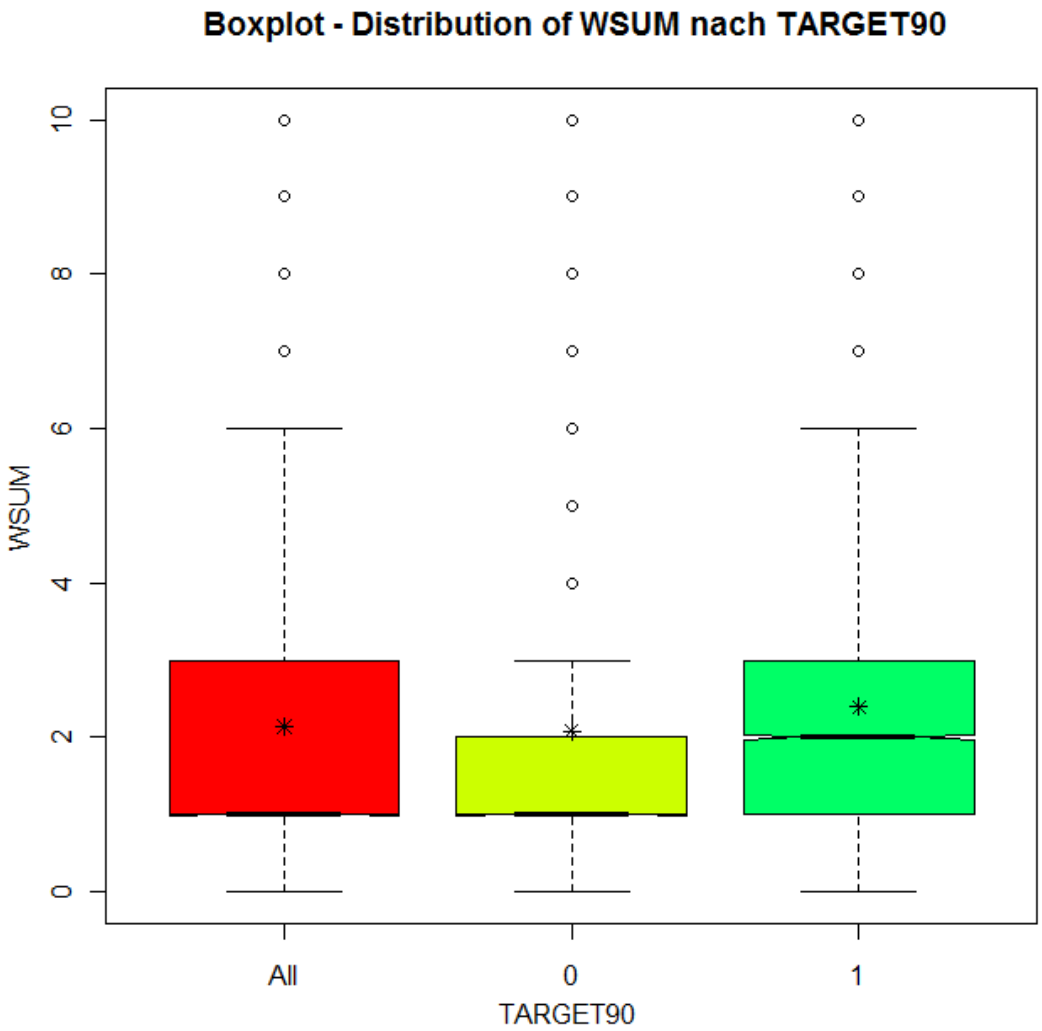


Visualisierung von metrischen Daten

Box-Plot mit Bezug zum Target

```
ds <- rbind(data.frame(dat=data[,"WSUM"],
   grp="All"), data.frame(dat=data[,]
   [data$TARGET90=="0","WSUM"], grp="0"),
   data.frame(dat=data[,] [data$TARGET90=="1",
   "WSUM"], grp="1"))
boxplot(formula=dat ~{} grp, data=ds, col=
   rainbow(5), xlab="TARGET90", ylab= "WSUM",
   ylim=c(0,10), notch=TRUE, main=" Boxplot -
   Distribution of WSUM nach TARGET90")
```

Visualisierung von metrischen Daten



Zusammenhänge in Daten

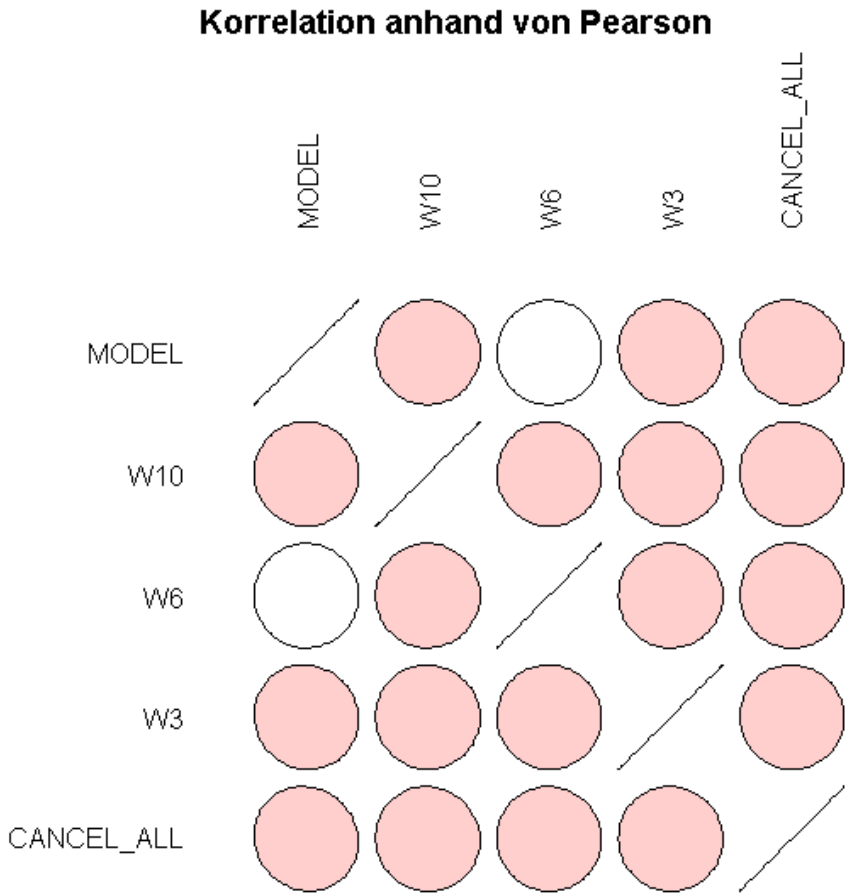
Korrelationsanalyse

- Einige DM-Algorithmen setzen unabhängige Variablen voraus
- Bessere DM-Ergebnisse mit vorheriger Korrelationsanalyse
- Funktion PLOTCORR() des Pakets ELLIPSE

Korrelationsanalyse für die ersten fünf Variablen

```
cor <- cor(result[,1:5],use="pairwise",method=
   "pearson")
ord <- order(cor[1,])
cor <- cor[ord,ord]
plotcorr(cor,col=colorRampPalette(c("red","
   white","blue"))(11)[5*cor + 6], main="
   Korrelation nach Pearson")
```

Zusammenhänge in Daten



Data Mining Modelle erstellen

Data Mining Modelle

- Sind nach wie vor in der Oracle Datenbank gespeichert
- Ergebnisse der Modellerstellung können nach R zurückgeliefert und weiterverarbeitet werden
- Aufruf von Data Mining Modellen erzeugt ein Ergebnis, welches in einem R-Objekt gespeichert wird
- Vorsicht: Übersicht über vorhandene Modelle kann schnell verloren gehen. Namenskonvention und gelegentliches Aufräumen ratsam.

Vorhandene Modelle anzeigen oder löschen

```
RODM_list_dbms_models(DB)
RODM_drop_model(DB,"MODELNAME")
```

Modellerstellung

Data Mining Modelle

- Entsprechende Methoden verfügbar, um einen Algorithmus aufzurufen
- Ähnlich wie bei der Verwendung der PL/SQL API
- Ergebnisse können auf Wunsch in den R-Workspace zurückgeliefert werden. (folgt auf nächster Folie)

Entscheidungsbaum erstellen

```
dt  <-  RODM_create_dt_model( database=DB,
   data_table_name="TRAINING_DATA",
   target_column_name="TARGET90",
   model_name="DTMODEL1" )
```

Ergebnis der Modellerstellung in R

R Console

Datei Bearbeiten Verschiedenes Pakete Windows Hilfe

```
> dt
$model.model_settings
            SETTING_NAME      SETTING_VALUE SETTING_TYPE
1              ALGO_NAME ALGO_DECISION_TREE        INPUT
2              PREP_AUTO                 ON        INPUT
3   TREE_IMPURITY_METRIC TREE_IMPURITY_GINI      DEFAULT
4    TREE_TERM_MAX_DEPTH                  7      DEFAULT
5  TREE_TERM_MINPCT_NODE                .05      DEFAULT
6 TREE_TERM_MINPCT_SPLIT                 .1      DEFAULT
7  TREE_TERM_MINREC_NODE                 10      DEFAULT
8 TREE_TERM_MINREC_SPLIT                 20      DEFAULT

$model.model_attributes
   ATTRIBUTE_NAME ATTRIBUTE_TYPE DATA_TYPE DATA_LENGTH DATA_PRECISION DATA_SCALE USAGE_TYPE TARGET
1    DELIVERYTYPE      NUMERICAL    NUMBER          22             NA          0     ACTIVE     NO
2   DIGITALBUYERS      NUMERICAL    NUMBER          22             NA          0     ACTIVE     NO
3           ENTRY      NUMERICAL    NUMBER          22             NA          0     ACTIVE     NO
4           MODEL      NUMERICAL    NUMBER          22             NA          0     ACTIVE     NO
5      NEWSLETTER      NUMERICAL    NUMBER          22             NA          0     ACTIVE     NO
6     NUMBERITEMS      NUMERICAL    NUMBER          22             NA          0     ACTIVE     NO
7     PAYMENTTYPE      NUMERICAL    NUMBER          22             NA          0     ACTIVE     NO
8            REMI      NUMERICAL    NUMBER          22             NA          0     ACTIVE     NO
9        REMITTER      NUMERICAL    NUMBER          22             NA          0     ACTIVE     NO
10  SHIPPINGCOSTS      NUMERICAL    NUMBER          22             NA          0     ACTIVE     NO
11       TARGET90    CATEGORICAL    NUMBER          22              0          0     ACTIVE    YES
12     USEDBUYERS      NUMERICAL    NUMBER          22             NA          0     ACTIVE     NO
13           WSUM      NUMERICAL    NUMBER          22             NA          0     ACTIVE     NO
14             W5      NUMERICAL    NUMBER          22             NA          0     ACTIVE     NO
15             W6      NUMERICAL    NUMBER          22             NA          0     ACTIVE     NO

$dt.tree_xml
[1] DETAILS
<0 Zeilen> (oder row.names mit Länge 0)

>
```

Anwendung des Modells

Data Mining Modelle anwenden: Voraussetzungen

- Modellname muss bekannt sein
- Zieldaten müssen in der Datenbank als Tabelle vorliegen und Berechtigungen für verbundenen Nutzer müssen erteilt sein

Modellliste

```
RODM_list_dbms_models(DB)
```

```
> modelList <- RODM_list_dbms_models(DB)
> modelList
             MODEL_NAME      MINING_FUNCTION                  ALGORITHM       CREATION_DATE BUILD_DURATION MODEL_SIZE COMMENTS
1             DMCTRAINAI ATTRIBUTE_IMPORTANCE MINIMUM_DESCRIPTION_LENGTH 2010-07-28 16:07:21              4     0.0267       NA
2            DMCTRAINAI2 ATTRIBUTE_IMPORTANCE MINIMUM_DESCRIPTION_LENGTH 2010-07-29 14:04:57              4     0.0269       NA
3             IRIS2263_SV       CLASSIFICATION    SUPPORT_VECTOR_MACHINES 2010-10-05 17:37:59              2     0.0830       NA
4           IRIS24349_CL           CLUSTERING                     KMEANS 2010-09-16 15:58:41              3     0.1316       NA
5           IRIS56751_NB       CLASSIFICATION                NAIVE_BAYES 2010-10-05 17:27:41              1     0.0487       NA
6           IRIS76613_DT       CLASSIFICATION              DECISION_TREE 2010-10-05 17:43:15              2     0.0817       NA
7           IRIS85863_SV       CLASSIFICATION    SUPPORT_VECTOR_MACHINES 2010-09-16 15:23:59              2     0.0882       NA
8           IRIS86310_DT       CLASSIFICATION              DECISION_TREE 2010-09-16 15:22:28              2     0.0817       NA
9           IRIS91230_AI ATTRIBUTE_IMPORTANCE MINIMUM_DESCRIPTION_LENGTH 2010-09-23 10:42:04              3     0.0212       NA
```

Anwendung des Modells

Data Mining Modelle anwenden: Befehl

```
dtapply <- RODM_apply_model(database=DB,
   data_table_name="TRAINING_DATA",
   model_name="DTMODEL1",
   supplemental_cols="TARGET90")
```

Wie sieht der Aufruf in der PL/SQL API aus?

```
dbms_data_mining.apply(
  model_name => 'DTMODEL1',
  data_table_name => 'TRAINING_DATA',
  case_id_column_name => 'TARGET90',
  result_table_name => 'dtapply');
```

Anwendung des Modells

Ergebnis verarbeiten

- Ergebnis der Anwendung des Modells ist eine Liste
- ! Befehle wie HEAD() schlagen fehl
- Weiterverarbeitung in R mit Hilfe von Data Frames jedoch flexibler
  - 1. Zeile: alle Zeilen und alle Spalten als Data Frame abspeichern
  - 2. Zeile: erste 20 Zeilen aber nur die Spalte TARGET90

Ergebnis in Data Frame speichern

```
myFrame <- appResult$model.apply.results[,]
myFrame2 <- appResult$model.apply.results
   [1:20,"TARGET90"]
```

Bewertung des angewendeten Modells

Confusion Matrix

- Gegenüberstellung von tatsächlichen und vorhergesagten Werten
- Auch mit der Funktion CMX() aus Paket PRESENCEABSENCE möglich

```
actual <- svm$model.apply.results[,"TARGET90"]
predicted <- svm$model.apply.results[,"
   PREDICTION"]
table(actual,predicted)
```

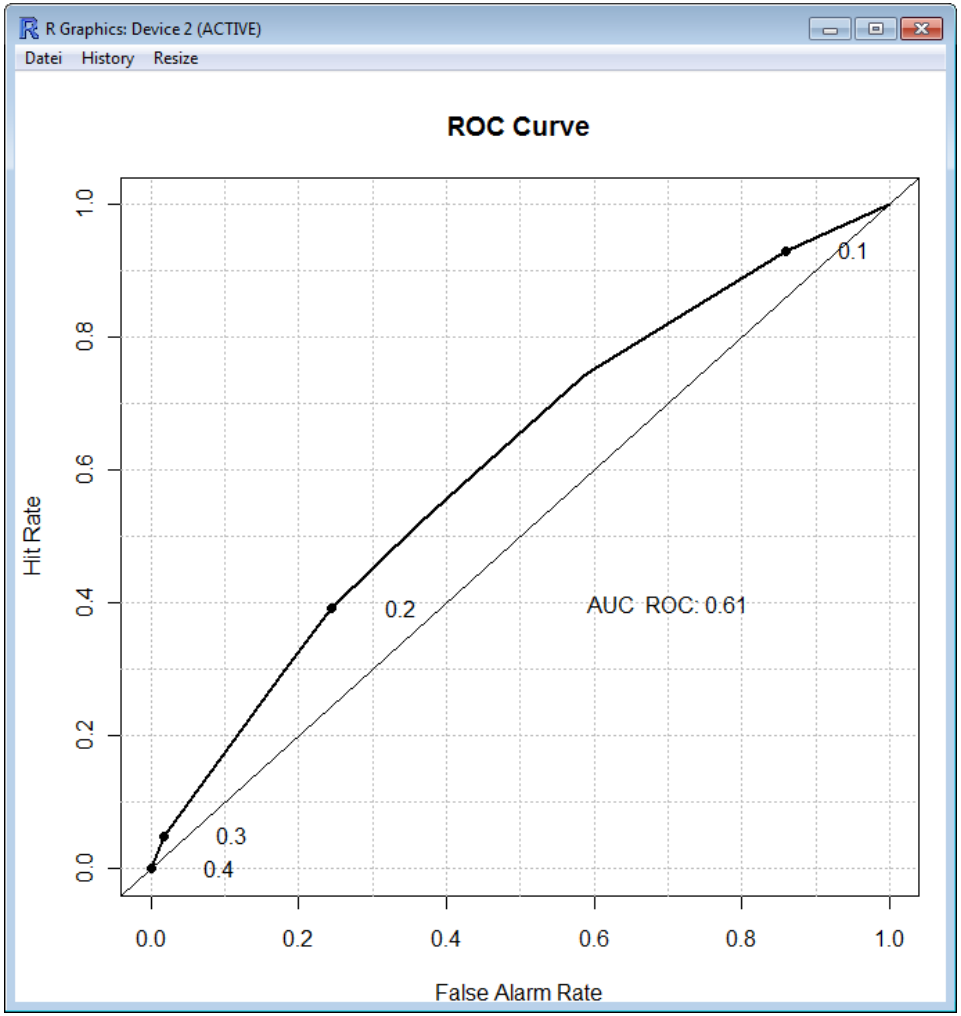
```
> table(actual,predicted)
      predicted
actual     0     1
     0 16796  9550
     1  2776  3274
```

Bewertung des angewendeten Modells

Receiver Operating Characteristic (ROC) Kurve

```
probs <- dta$model.apply.results[,"'1'"]
perf.auc <- roc.area(ifelse(actual=="1",1,0),
   probs)
roc.plot( actual, probs )
text(0.7,0.4,labels=paste("AUC ROC:",signif(
   perf.auc$A,digits=3)))
```

Bewertung des angewendeten Modells



Ausblick: Automatisierung mit R-Skripten

- Neue Datensätze werden hinzugefügt und ein neues Modell erstellt
- Historie der Vorhersagen somit ersichtlich

```
neu <- result$model.apply.results[,]
# Nummer des Iterationsprozesses speichern
neu[,5] <- 1
# neue Ergebnisse an bisherige anhängen
gesamt <- rbind(gesamt,neu)
# Ergebnisse sortieren zur Übersicht
gesamt <- gesamt[with(gesamt, order(gesamt$
   CUSTOMERNUMBER)),]
# Ergebnis in Datenbank speichern
RODM_SVMC_SIM <- gesamt
RODM_create_dbms_table(DB,"RODM_SVMC_SIM")
```

Fazit

Stärken und Möglichkeiten von R-ODM

- Sehr gut geeignet für Testzwecke
- Modularer Aufbau von R schafft flexible Umgebung und bietet für die meisten Anwendungsfälle entsprechende Lösungen
- Grafische Oberflächen vereinfachen die Arbeit weiter und somit sind weder tiefgreifende Datenbank- noch R-Kenntnisse notwendig
- Größte Stärke von R ist die Datenanalyse und die Visualisierung
  - besonders geeignet für die Data Understanding Phase

Fazit

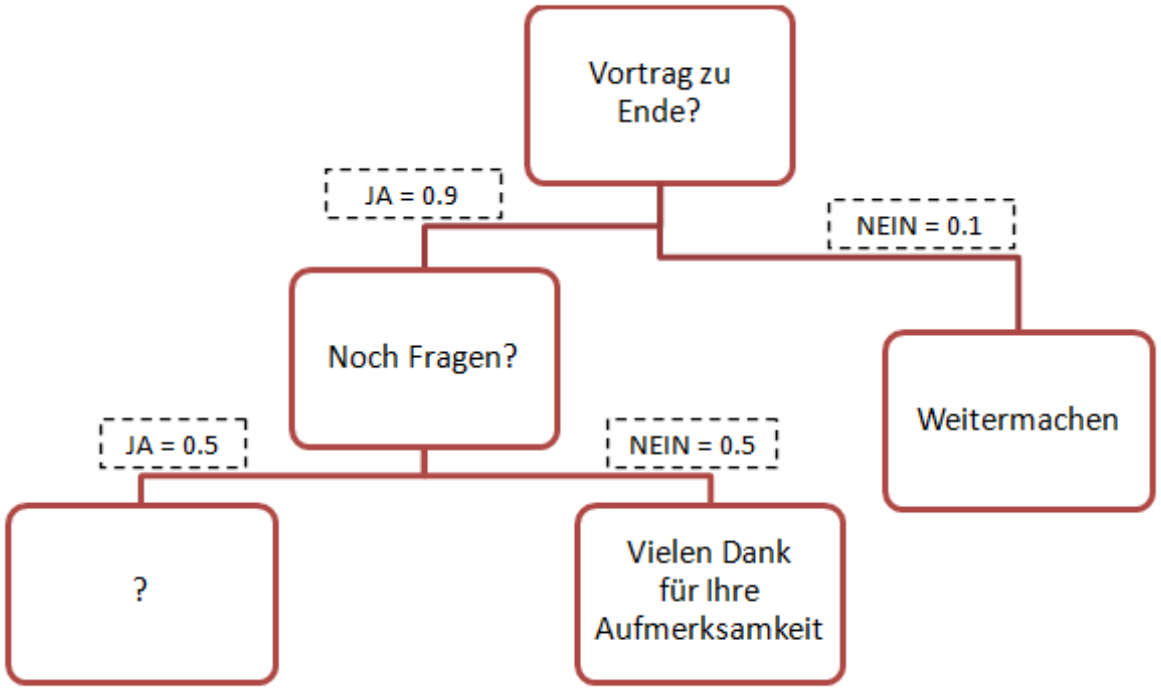
Schwächen von R-ODM

- Verarbeitung der Daten erfolgt im Hauptspeicher des Host Systems, was bei großen Datenmengen problematisch ist
  - auch hier stehen jedoch Erweiterungen zur Verfügung, die sich dem Problem widmen
  - durch die Verbindung mit Oracle Data Mining wird dieser Nachteil ebenfalls abgeschwächt
- Keine direkte Auswahl von Spalten einer Tabelle mit R möglich (vgl. Oracle Data Miner), hierzu muss eine neue Tabelle z.B. mit CTAS angelegt werden

Quellen und nützliche Links

- R Projekt
- RODM CRAN Webseite (Download und Handbuch)
- Beschreibung von RODM auf der Oracle Webseite
- Oracle Data Mining Blog
- Dokumentation von Oracle Data Mining

Noch Fragen?


Vortrag zu Ende?
JA = 0.9
NEIN = 0.1
Noch Fragen?
Weitermachen
JA = 0.5
NEIN = 0.5
?
Vielen Dank für Ihre Aufmerksamkeit