Multidimensionales Clustering mit Webanalysedaten

[R] Kenntnis-Tage, November 2016

Alexander Kruse, Data Analyst

etracker®

Inhalt

1. Über die Webanalyse
2. Use-Case
    1. Anforderungen des Kunden
    2. Herausforderungen
    3. Lösungsansatz
    4. Aufbereitung der Ergebnisse
3. Vorteile von R

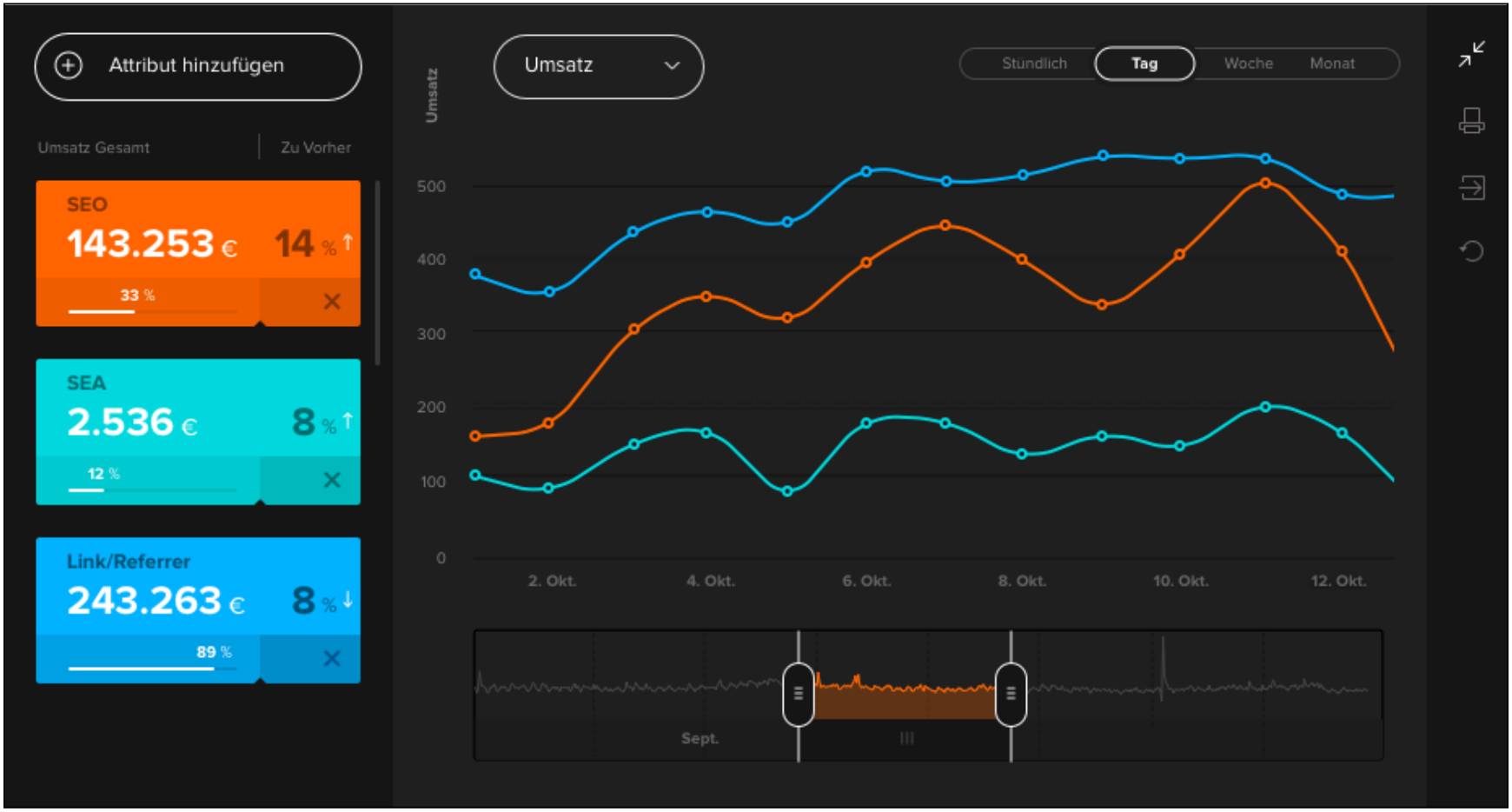
Über die Webanalyse

Über die Webanalyse

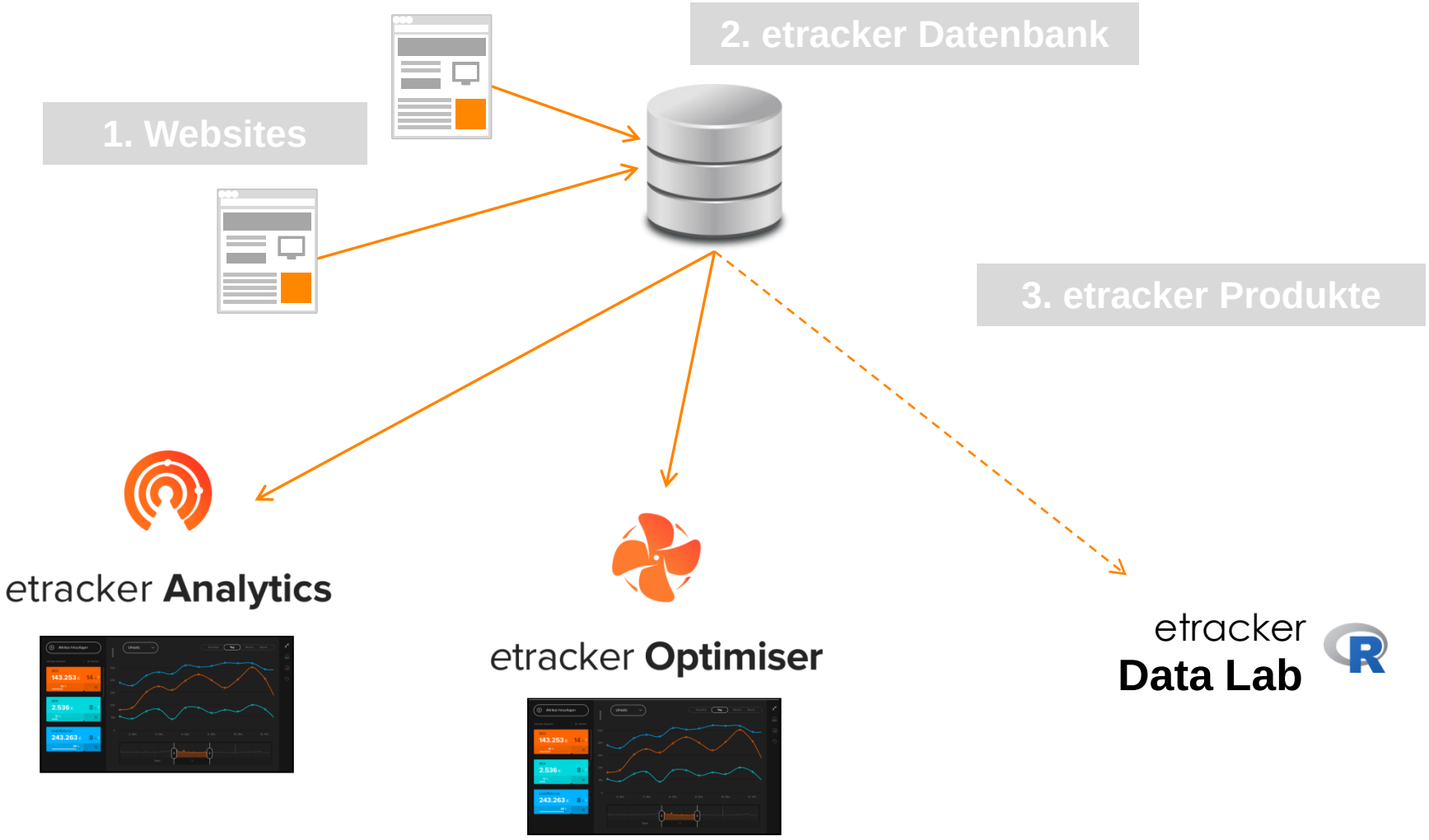
- Webanalyse ist die Sammlung von Daten und deren Auswertung bzgl. des Verhaltens von Besuchern auf Websites. Ein Webanalyse-Tool untersucht typischerweise, woher die Besucher kommen, welche Bereiche auf einer Website aufgerufen werden und wie oft und wie lange welche Unterseiten und Kategorien angesehen werden
- Wichtige Kennzahlen beziehen sich zum Beispiel auf:
  - die Anzahl der Besucher eines Onlineshops
  - den Anteil der Besucher, die etwas in den Warenkorb legen
  - den Anteil der Besucher, die den Kaufprozess abschließen
  - den durchschnittlichen Warenkorbwert
  - die Zeitspanne bis zum Kauf im Onlineshop
  - die Wirksamkeit einzelner Werbemittel (z. B. Banner, Newsletter)

Über die Webanalyse

- etracker ist ein kommerzielles Webanalyse-Tool



Über das etracker Data Lab



Webanalyse-Daten in Rohform

- Fein granulare Daten auf UUID-Ebene mit Zeitstempel mit bis zu 400.000 Datenpunkten pro Tag (pro Kunde)
- Ø um die zwanzig Features in den Rohdaten

userid	sessionid	timestamp	…
0001	0001	2016-07-28 17:45:34	…
0001	0001	2016-07-28 17:45:42	…
0001	0001	2016-07-28 17:46:03	…
0001	0002	2016-07-29 20:01:56	…
0001	0002	2016-07-29 20:02:13	…
0002	0001	2016-08-01 10:11:45	…
0002	0001	2016-08-01 10:11:59	…
…	…	…	…

Weitere Features
pagename
product_name
product_category
product_price
device
order_number
channel
…

Use-Case: Anforderungen des Kunden

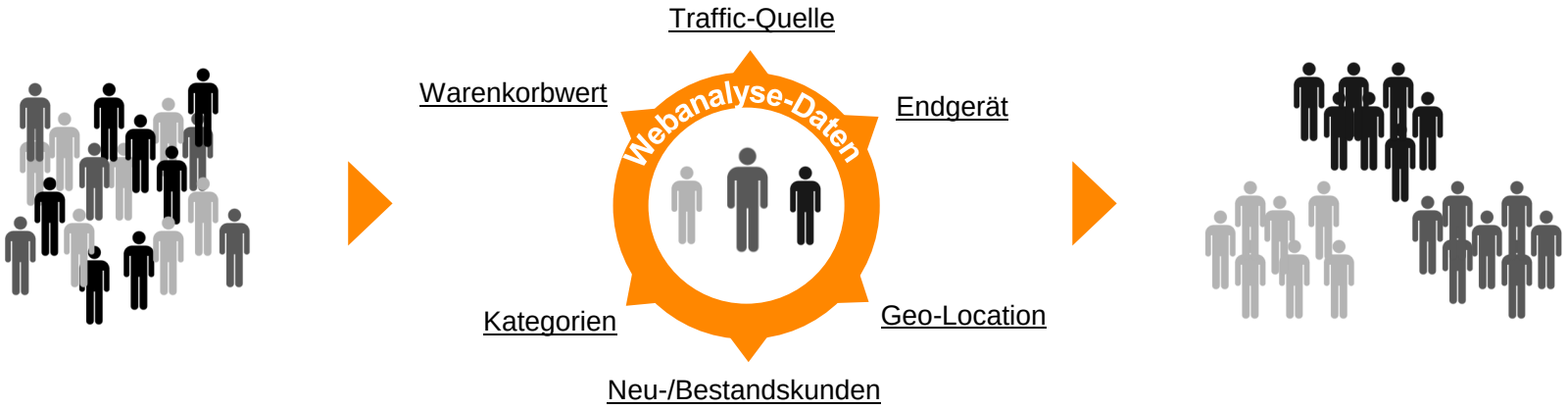
Beschreibung des Kunden

- Internationaler Onlineshop mit wöchentlich…
  - 60.000 Besuchern
  - 175.000 Sessions
  - 500.000 Produktaufrufen
  - 2.000.000 Seitenaufrufen
  - 5.000 Bestellungen
  - Ø 75,00 € pro Bestellung

- Weitere Infos zum Onlineshop:
  - Marketing über Newsletter, Retargeting, Preissuchmaschinen, Google, …
  - Verschiedenste Endgeräte, Betriebssysteme, Browser, …
  - Komplexe Bereichs- und Produkthierarchie

Anforderungen des Kunden

- Multidimensionale Segmentierung der Onlineshop-Besucher
  - Der Shopbetreiber weiß nicht, in welchen Eigenschaften sich seine Besucher besonders stark unterscheiden,
  - möchte herausfinden, wie viele Besuchertypen es eigentlich gibt, weil
  - der eindimensionale Ansatz keine verwertbaren Ergebnisse gebracht hat.



Use-Case: Herausforderungen

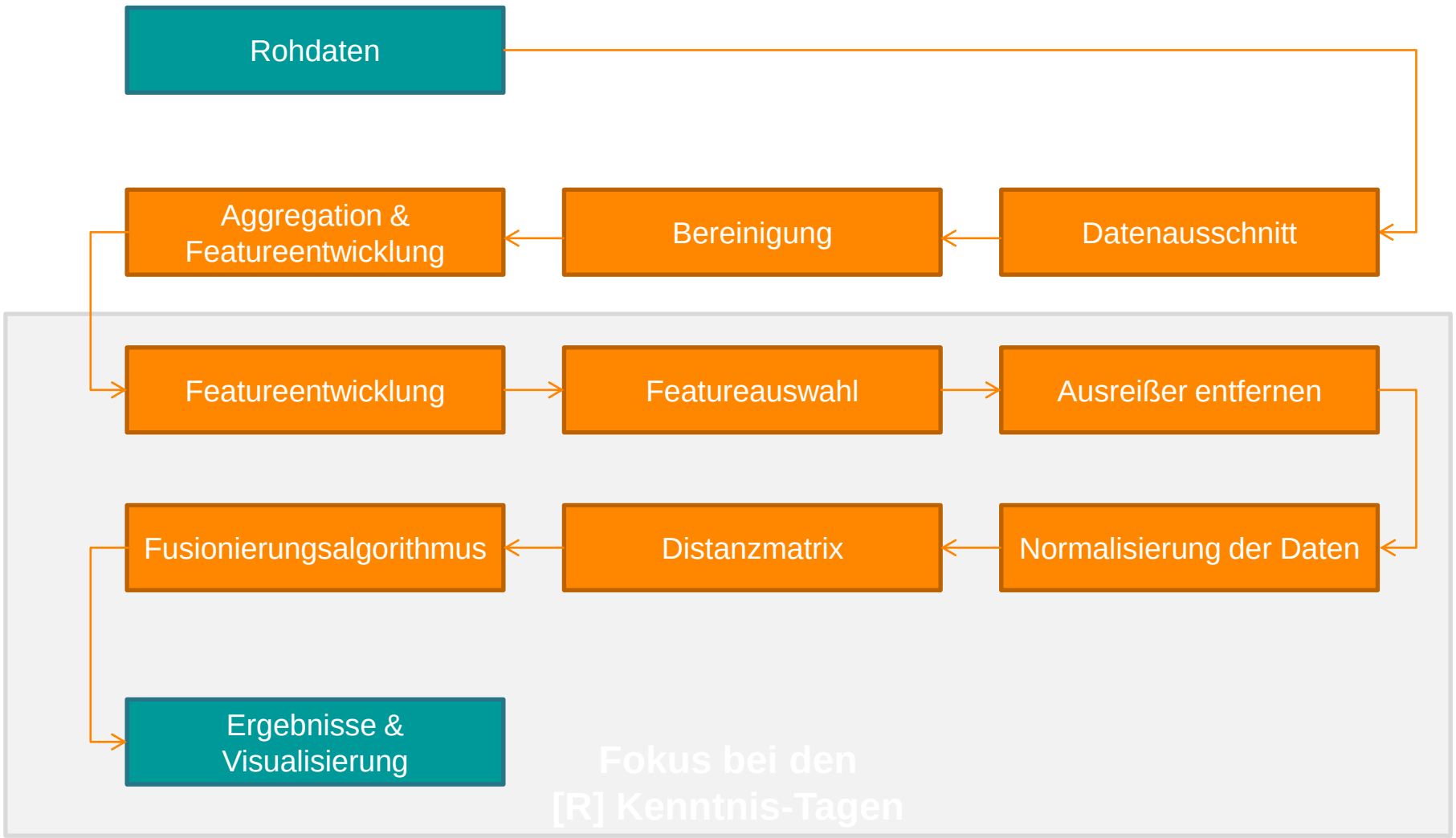
Herausforderungen

1. **Unsupervised Machine Learning:** Maschinelles Lernen ohne im Voraus bekannte Zielwerte sowie ohne Belohnung durch die Umwelt. Der Algorithmus versucht, in den Eingabedaten Muster zu erkennen, die vom strukturlosen Rauschen abweichen. Das Problem sind nicht immer eindeutige Ergebnisse wie beim überwachten Lernen.

2. **Die Gruppe der „Bouncer“:** Die Absprungrate (Bounce-Rate) umfasst per Definition alle Besucher mit nur einem Seitenaufruf. Die Gruppe ist in der Regel sehr groß aber irrelevant für die Besuchersegmentierung.

Use-Case: Lösungsansatz

Lösungsprozess



Datenfeatures

- Die Datenvorverarbeitung (Bereinigung, Featureentwicklung, Aggregation) wurde insb. mit data.table und dplyr durchgeführt
- Das Ergebnis sind folgende Datenfeatures:

Wochentag
Montag
Dienstag
Mittwoch
Donnerstag
Freitag
Samstag
Sonntag

Nutzung
Session_count
Session_duration_in_min
Mean_session_duration_in_min
Overall_duration_in_days
Product_views
Product_view_value

Kaufabschluss
Conversions
Conversion_value
Mean_conversion_value
Total_products

Endgerät
Smartphone
Tablet
Desktop

Herkunft
Affiliate
Link
Newsletter
PLA
PSM
Retargeting
SEA
SEO
SM
Social
TypeIn

- Input für die Clusteranalyse ist folgende Tabelle:

UserID	Session_count	Session_duration	…
…	…	…	…

Datenfeatures

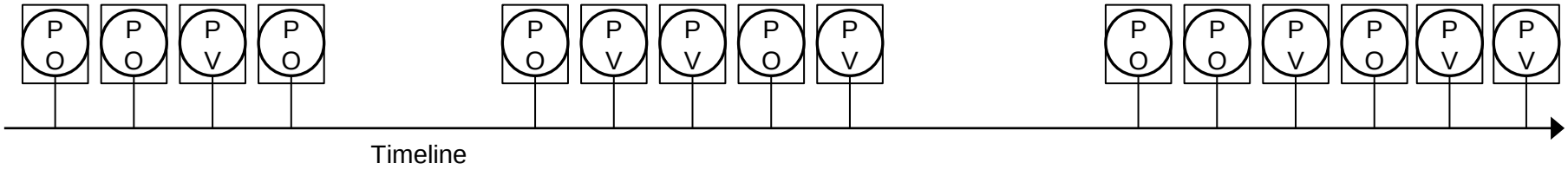
- Deskriptive Statistik der ersten zehn Variablen:

```
> summary(data[1:10])
 session_count    overall_duration_in_days session_duration_in_min mean_session_duration_in_min product_views
 Min.   : 1.000   Min.   :  0.0            Min.   :   0.00         Min.   :  0.000              Min.   :  0.00
 1st Qu.: 2.000   1st Qu.:  0.0            1st Qu.:   6.00         1st Qu.:  3.000              1st Qu.:  4.00
 Median : 2.000   Median : 16.0            Median :  16.00         Median :  6.000              Median :  8.00
 Mean   : 5.188   Mean   : 79.1            Mean   :  43.37         Mean   :  8.586              Mean   : 21.88
 3rd Qu.: 5.000   3rd Qu.:140.0            3rd Qu.:  40.50         3rd Qu.: 11.000              3rd Qu.: 20.00
 Max.   :76.000   Max.   :365.0            Max.   :7262.00         Max.   :330.091              Max.   :460.00
  conversions     mean_conversion_value conversion_value   total_products    product_view_value
 Min.   :0.0000   Min.   :  0.000       Min.   :   0.000   Min.   : 0.0000   Min.   :   0.00
 1st Qu.:0.0000   1st Qu.:  0.000       1st Qu.:   0.000   1st Qu.: 0.0000   1st Qu.:  56.94
 Median :0.0000   Median :  0.000       Median :   0.000   Median : 0.0000   Median :  98.00
 Mean   :0.0803   Mean   :  7.253       Mean   :   9.242   Mean   : 0.1519   Mean   : 125.92
 3rd Qu.:0.0000   3rd Qu.:  0.000       3rd Qu.:   0.000   3rd Qu.: 0.0000   3rd Qu.: 152.33
 Max.   :7.0000   Max.   :526.000       Max.   :1312.000   Max.   :15.0000   Max.   :2679.49
```

Featureentwicklung

- Bei diesem Prozess wird versucht, zusätzliche relevante Features aus den vorhandenen Rohfeatures in den Daten zu erstellen um die Leistung des Algorithmus zu steigern
- Die neuen Features sollen zusätzliche Informationen bereitstellen, die in den ursprünglichen oder vorhandenen Featuregruppen nicht eindeutig erfasst werden können oder nicht leicht ersichtlich sind

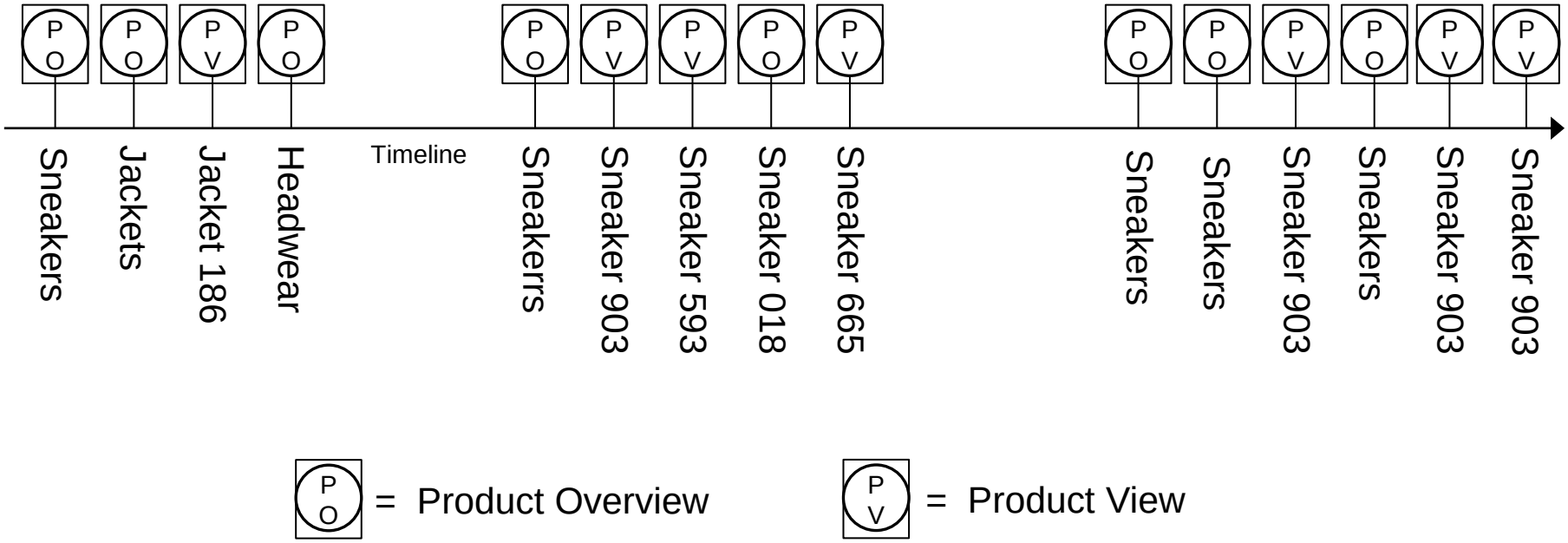
Featureentwicklung: Nutzungskontext



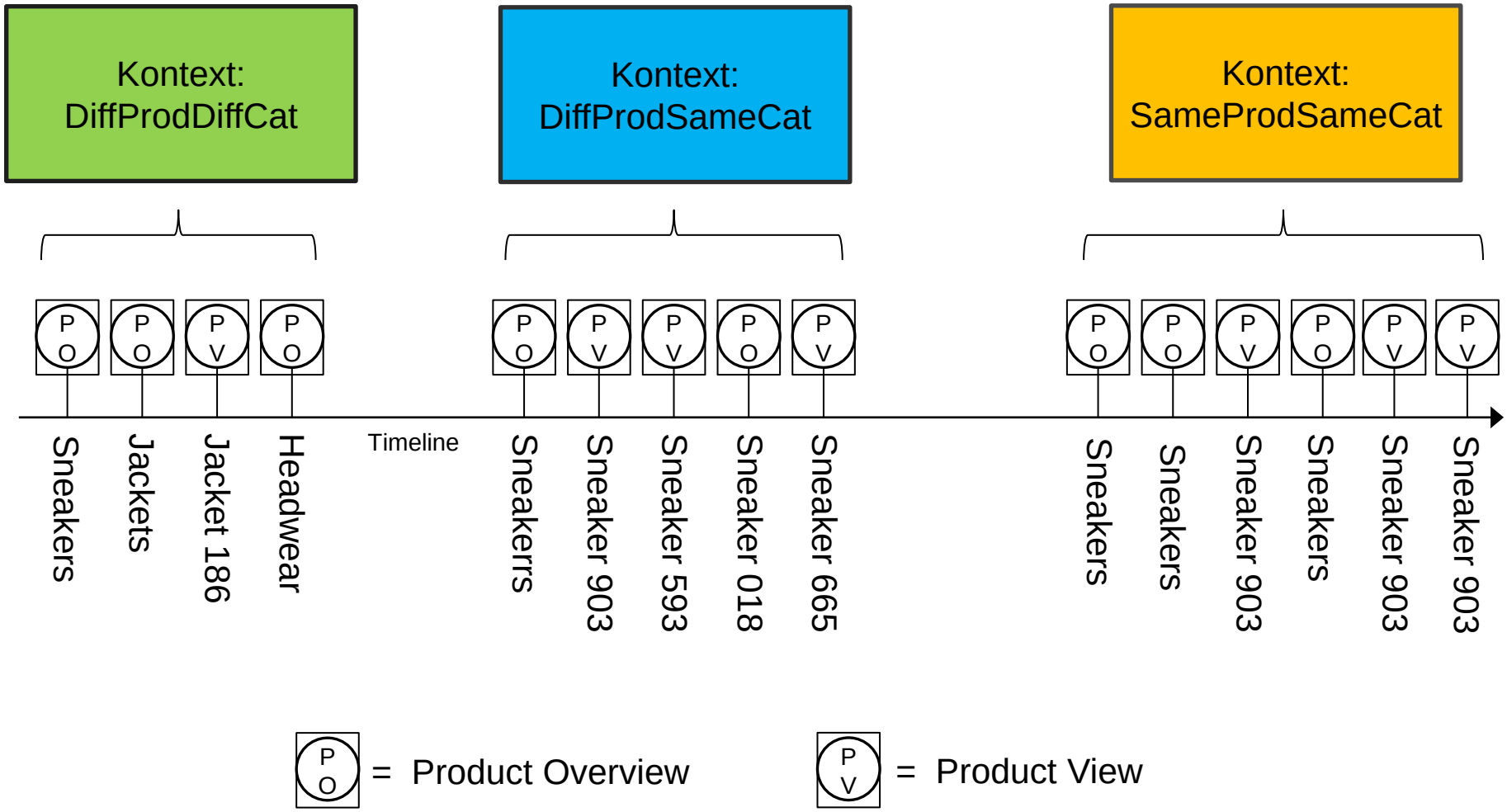
P O = Product Overview

P V = Product View

Featureentwicklung: Nutzungskontext



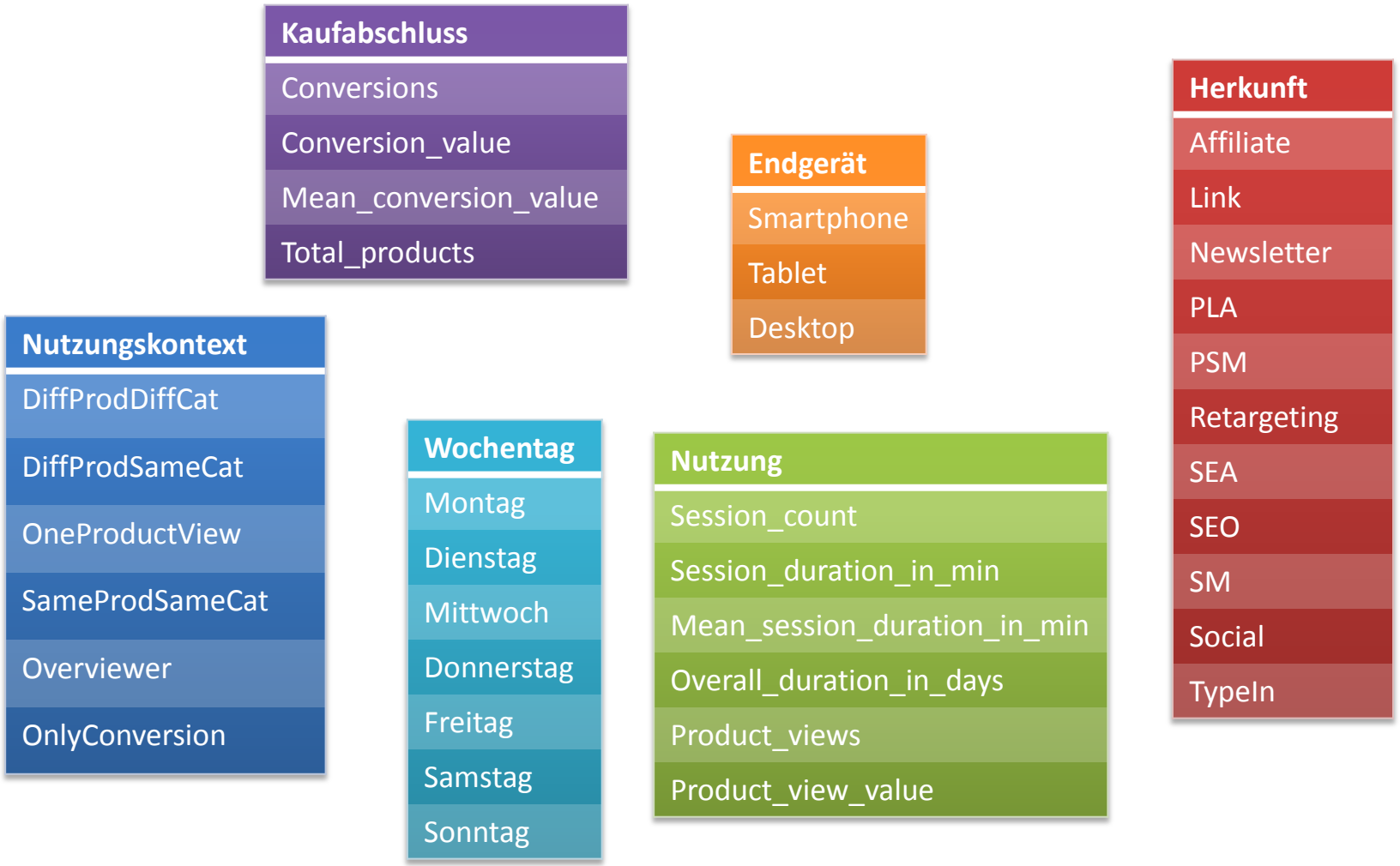
Featureentwicklung: Nutzungskontext



Featureentwicklung: Nutzungskontext

Nutzungskontext	Beschreibung
DiffProdDiffCat	Besucher ruft versch. Produkte in versch. Produktkategorien auf
DiffProdSameCat	Versch. Produkte in gleicher Produktkategorie
OneProductView	Besuch besteht aus einem Produktaufruf (keine weiteren Seitenaufrufe)
SameProdSameCat	Mehrfache Aufrufe des gleichen Produkts
Overviewer	Größtenteils Aufrufe von Produktübersichtsseiten
OnlyConversion	Besuch dient ausschließlich dem Kaufabschluss

Datenfeatures

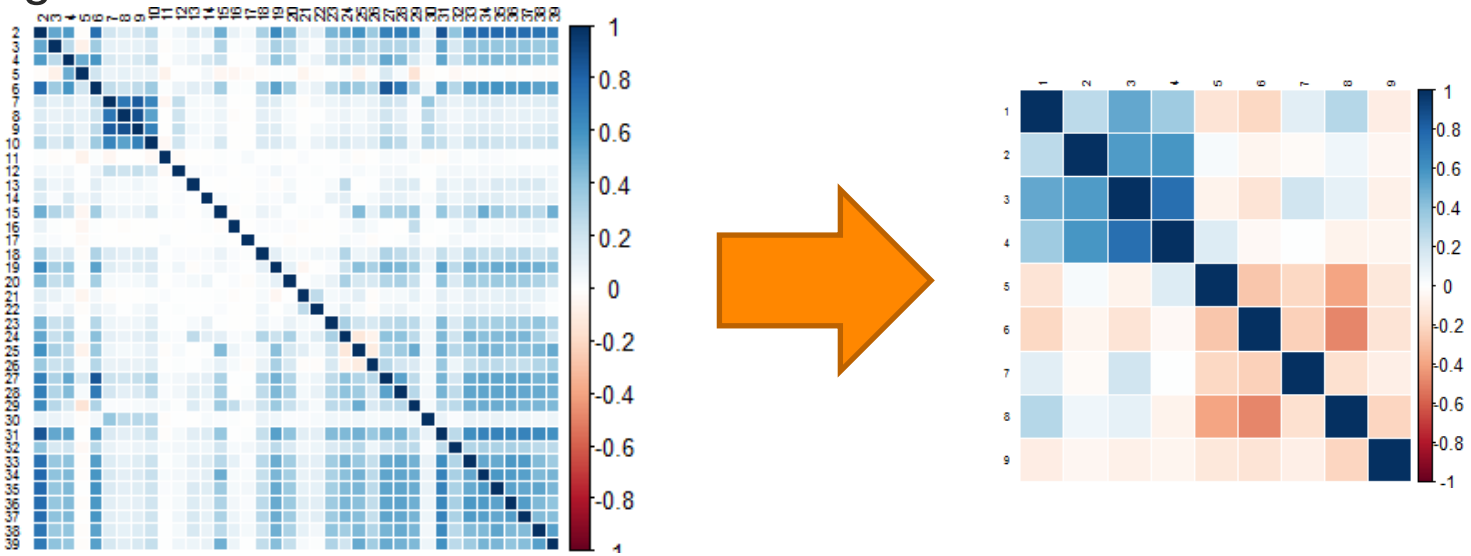


Featureauswahl

- Dieser Prozess wählt bei dem Versuch, die Anzahl von Dimensionen des Datensatzes zu verringern, die wichtigste Teilmenge der ursprünglichen Datenfeatures aus
- Zunächst steigert die Featureauswahl häufig die Ergebnisgenauigkeit durch Eliminieren irrelevanter, redundanter oder hochgradig korrelierter Features. Zweitens sinkt die Anzahl von Features, was den Analysevorgang effizienter gestaltet

Featureauswahl

- Einteilung in Segmentierungsvariablen (werden zum Clustering genutzt) und beschreibende Variablen (dienen ausschließlich zur späteren Beschreibung der Cluster)
- Eine Faktorenanalyse zur Reduktion der Variablen hat keine Vorteile gebracht



```
# correlation matrix
colnames(data) <- 1:length(data)
corrplot(cor(data), method = "color", tl.cex = 0.5, tl.col = 'black')
```

Entfernung von Ausreißern

- Ein Ausreißer ist eine ungewöhnlich große oder kleine Beobachtung. Ausreißer können sich unverhältnismäßig auf statistische Ergebnisse auswirken, z. B. auf den Mittelwert, was zu irreführenden Interpretationen führen kann
- Ein einzelner Wert kann als Ausreißer betrachtet werden, wenn er z.B. außerhalb eines bestimmten Bereichs der Standardabweichung liegt

```
# create function that looks for values > +/- 5 sd from mean
outlier <- function(x) abs(scale(x)) >= 5

# index with the function to remove those values
data <- data[!apply(sapply(as.data.frame(data), outlier), 1, any), ]
```

Wahl des Clusterverfahrens

- Bei der agglomerativen Berechnung einer hierarchischen Clusteranalyse wird zu Beginn zunächst jedes Objekt als ein eigenes Cluster aufgefasst. Nun werden in jedem Schritt die jeweils einander nächsten Cluster zu einem Cluster zusammengefasst. Besteht ein Cluster aus mehreren Objekten, dann muss angegeben werden, wie die Distanz zwischen Clustern berechnet wird und hier unterscheiden sich die einzelnen agglomerativen Verfahren. Das Verfahren kann beendet werden, wenn eine genügend kleine Zahl von Clustern ermittelt worden ist.
- Für die Durchführung einer agglomerativen Clusteranalyse müssen
  - ein Distanz- oder Ähnlichkeitsmaß zur Bestimmung des Abstandes zwischen zwei Objekten und
  - ein Fusionierungsalgorithmus zur Bestimmung des Abstandes zwischen zwei Clustern ausgewählt werden.

Wahl des Proximitätsmaß

- Die Ausgangssituation bei der Konstruktion eines Ähnlichkeits- und Distanzmaßes ist die multivariate Datenmatrix X mit n Objekten und p Merkmalen folgender Form:

$$X = \begin{bmatrix} x_{11} & \cdots & x_{1p} \\ \vdots & \ddots & \vdots \\ x_{n1} & \cdots & x_{np} \end{bmatrix}$$

- Mittels der Ähnlichkeits- und Distanzmaße wird die Datenmatrix in eine Ähnlichkeits- bzw. Distanzmatrix umgewandelt.

$$D = \begin{bmatrix} d_{11} & \cdots & d_{1n} \\ \vdots & \ddots & \vdots \\ d_{n1} & \cdots & d_{nn} \end{bmatrix} \text{ bzw. } S = \begin{bmatrix} s_{11} & \cdots & s_{1n} \\ \vdots & \ddots & \vdots \\ s_{n1} & \cdots & s_{nn} \end{bmatrix}$$

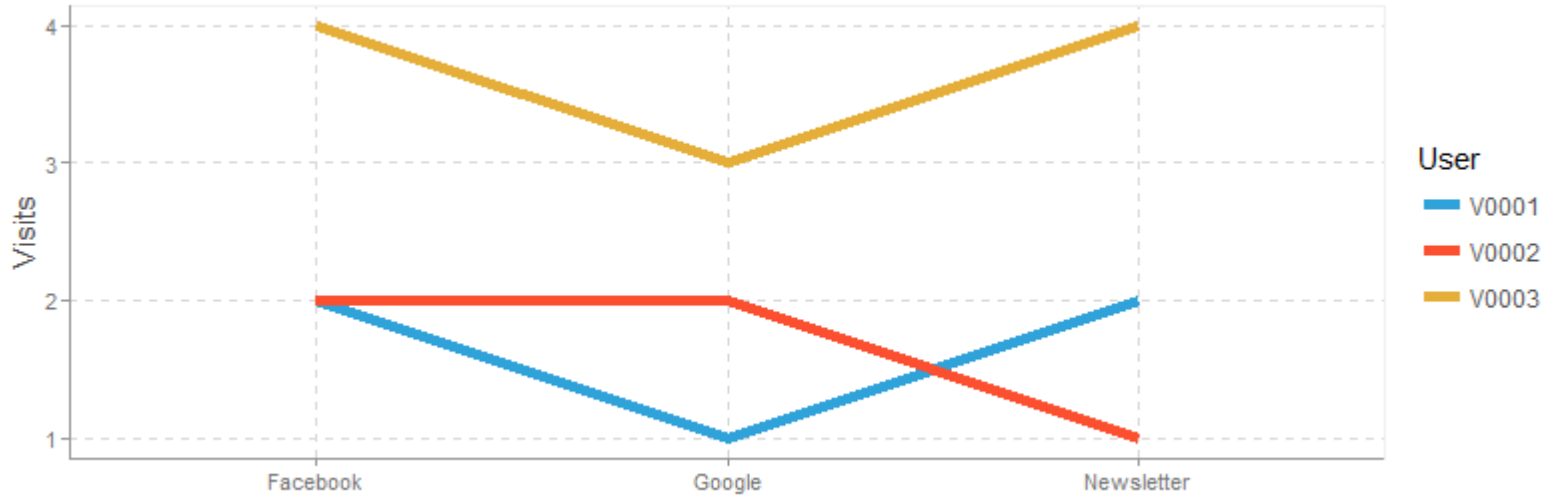
Anforderungen an das Proximitätsmaß

- Ausschließlich metrisch skalierte Variablen
- Korrelative Abhängigkeiten zwischen den Merkmalen
- Absolute Distanz vs. Profil

Distanzmaß	R-Paket
Euklidische Distanz	stats, proxy
Mahalanobis Distanz	stats
Kosinus-Ähnlichkeit (not centered Pearson)	amap

Anforderungen an das Proximitätsmaß

Visitor	Facebook Link	Google Search	Newsletter Link	Σ (Visits)
V0001	2	1	2	**= 5**
V0002	2	2	1	**= 5**
V0003	4	3	4	**= 11**




```
> dist(as.matrix(data), method = "euclidean")
         V0001    V0002
V0002 1.414214
V0003 3.464102 3.741657
```

```
> simil(as.matrix(data), method = "pearson")
          V0001     V0002
V0002 0.4472136
V0003 0.7071068 0.4472136
```

Anforderungen an das Proximitätsmaß

- Ausschließlich metrisch skalierte Variablen
- Korrelative Abhängigkeiten zwischen den Merkmalen
- Absolute Distanz vs. Profil

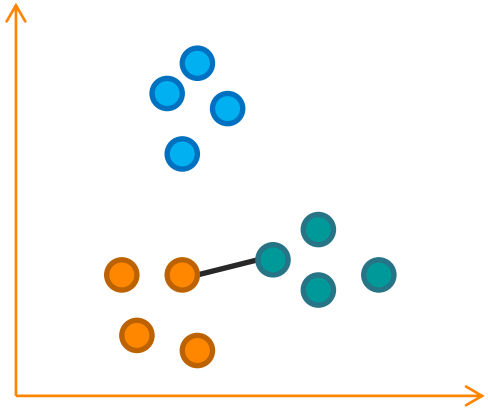
Distanzmaß	R-Paket
Euklidische Distanz	stats, proxy
Mahalanobis Distanz	stats
Kosinus-Ähnlichkeit (not centered Pearson)	amap

Wahl des Fusionierungsalgorithmus: Ward-Verfahren

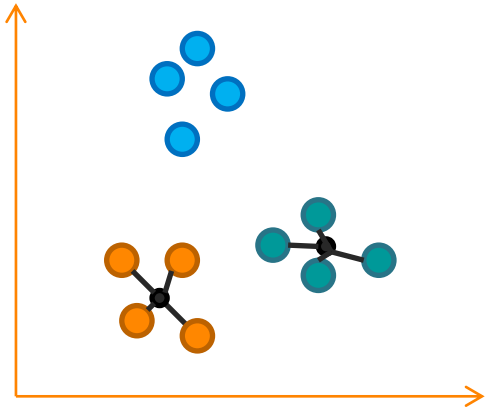
- Das Ward-Verfahren beruht auf folgender Idee: Fusioniere die beiden Cluster, welche die minimalste Erhöhung der Varianz im neuen Cluster (durch das Hinzufügen weiterer Beobachtungen) erzeugen. Dies entspricht einem minimalen Zuwachs der Fehlerquadratsumme durch die Fusion
- Damit lässt sich für das Ward-Verfahren auch sagen: Der auftretende Homogenitätsverlust durch die Fusionierung zweier Cluster soll minimiert werden
- Das Ward-Verfahren neigt zur Bildung von Clustern mit ähnlicher Größe

Wahl des Fusionierungsalgorithmus: Ward-Verfahren

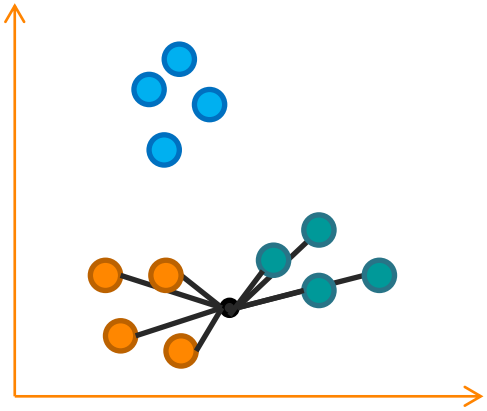
- Single link:



- Ward-Verfahren:



vs.



Clusterbildung

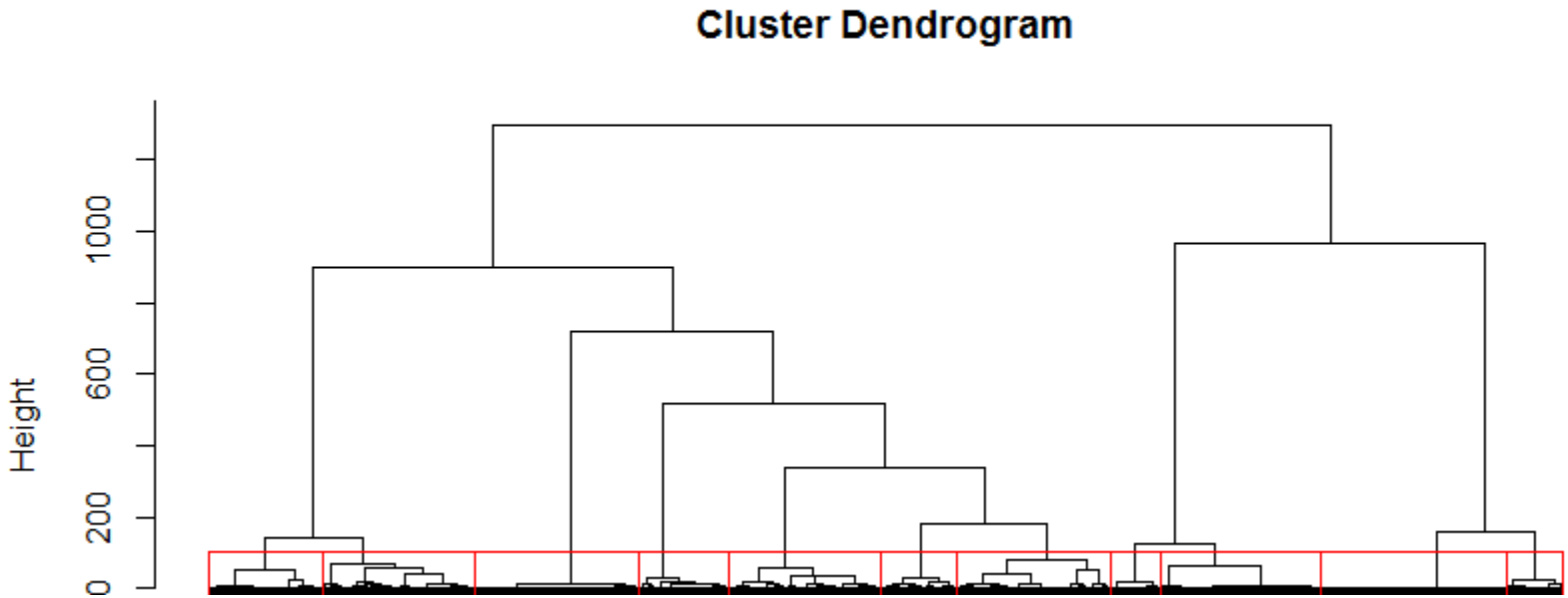
- Unterschiedliche Skalenniveaus zwischen den Merkmalen erfordert die Normalisierung der Daten
- Die Funktion hcluster (amap) ist ein Mix aus hclust und dist (stats) und ist effizienter in der Berechnung als die separate Anwendung der einzelnen Funktionen

```
# scale data
mydata_user <- scale(mydata_user)

# distance matrix
#dm <- Dist(mydata_user, method = "pearson")
#c <- hclust(dm, method = "ward.D2")

# distance matrix and clustring
c <- hcluster(mydata_user, method = "pearson", link = "ward")
```

Wahl der Clusteranzahl

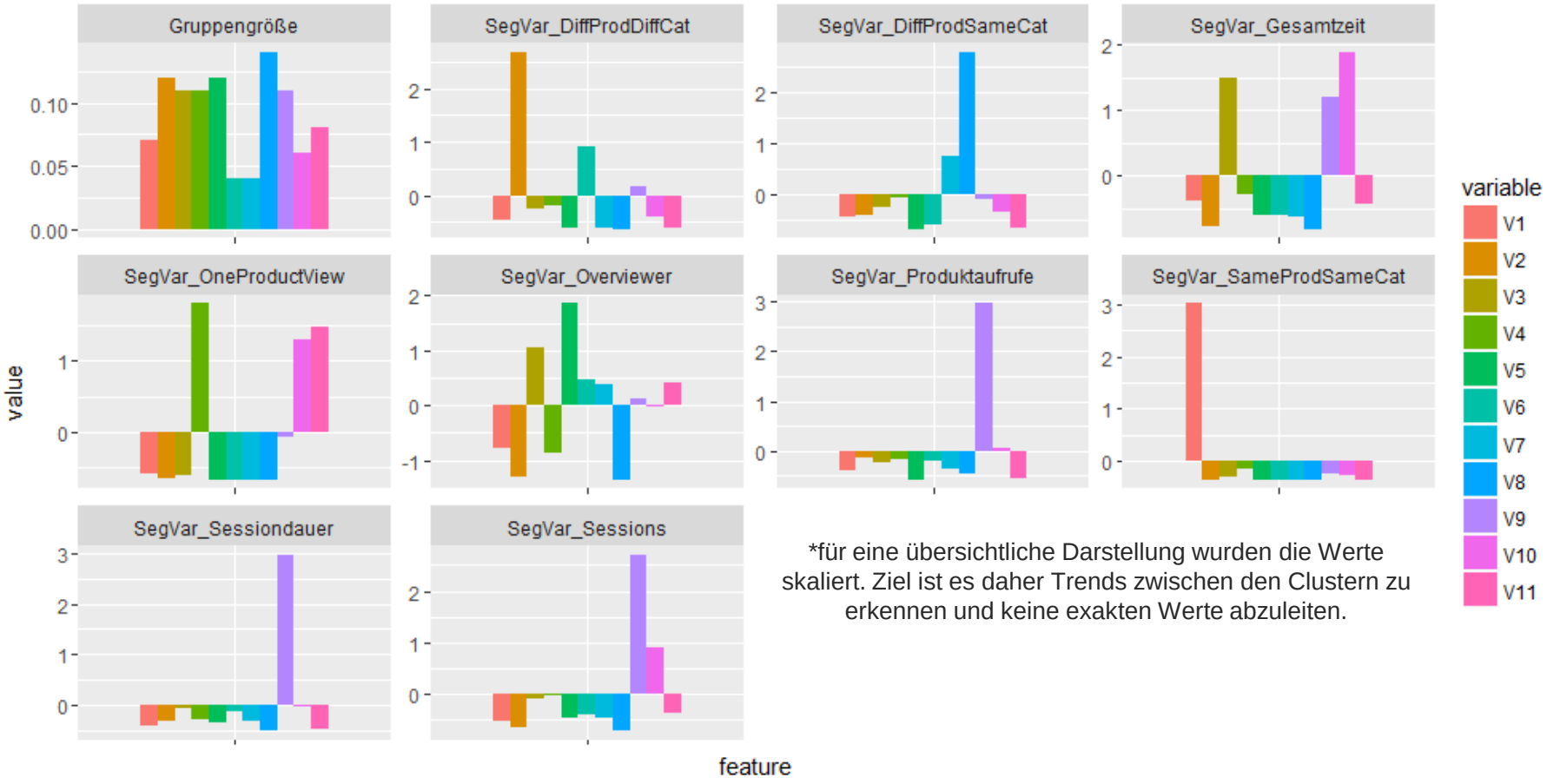



```
# plot dendogram + red border
plot(c)
rect.hclust(c, k=11, border="red")

# write clusters back to data
hcluster <- cutree(c, k=11)
data <- cbind(data, hcluster)
```

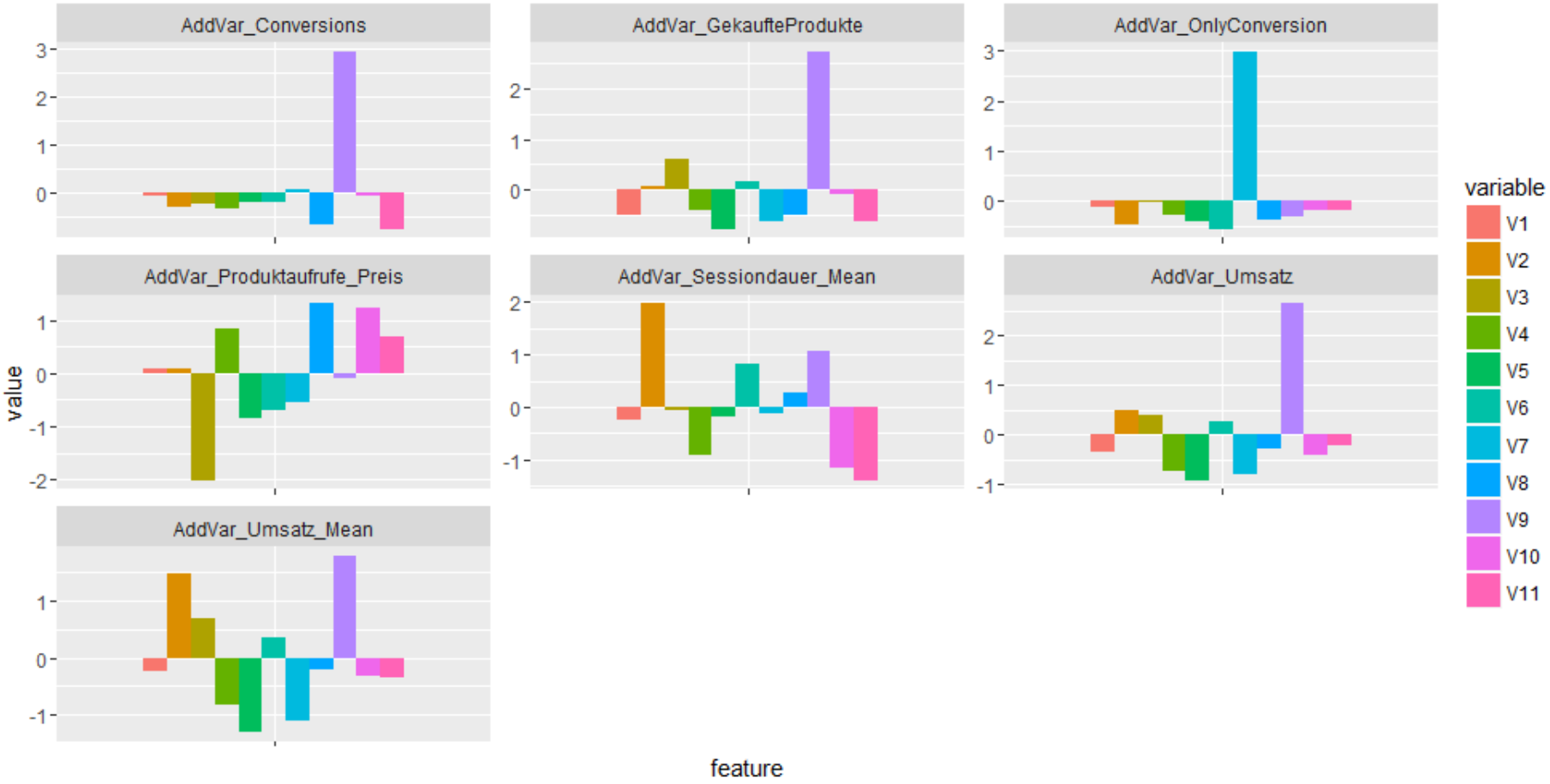
Use-Case: Einblick in die Ergebnisse

Ergebnisse

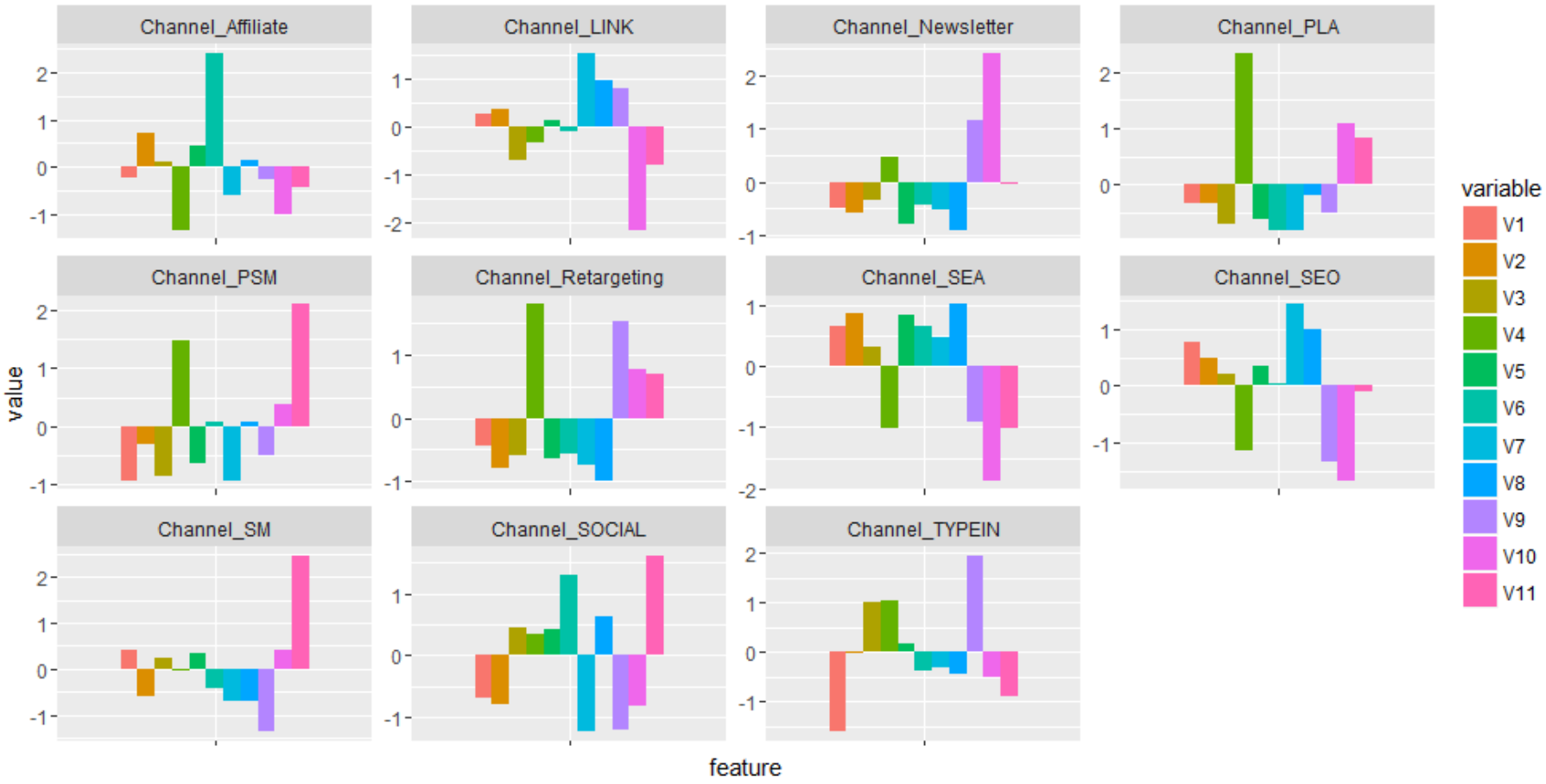


*für eine übersichtliche Darstellung wurden die Werte skaliert. Ziel ist es daher Trends zwischen den Clustern zu erkennen und keine exakten Werte abzuleiten.

Ergebnisse



Ergebnisse



Ableitung von Personas

Cluster 1: Die Schnäppchenjäger

- Bewegen sich vor allem im Sales-Bereich
- Ø 2 Visits im Analysezeitraum
- Ø 1 Artikel pro Bestellung
- Besuchen viele verschiedene Bereiche
- Geringer Warenkorbwert (Ø 25 €)
- Geringe Retourenquote (Ø 16 %)
- Kommen häufig über E-Mails und SEA

Cluster 2: Die Trendsetter

- Bewegen sich vor allem im Trends-Bereich
- Ø 2 Visits im Analysezeitraum
- Ø 1 Artikel pro Bestellung
- Besuchen viele verschiedene Bereiche
- Mittlerer Warenkorbwert (Ø 86 €)
- Nutzen den Kundenservice selten
- Kommen häufig über Retargeting

Cluster 3: Die Überlegten

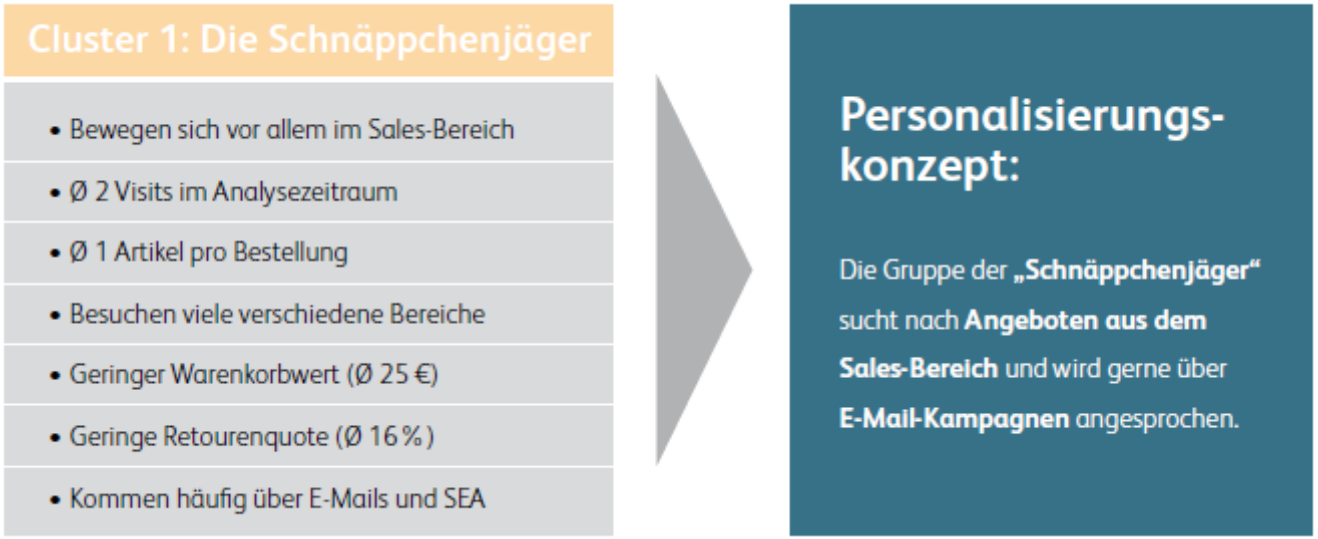
- Viele Visits im Analysezeitraum (Ø 6 Visits)
- Nutzen häufig die Produktsuche
- Viele Produktaufrufe (Ø 9 Produkte)
- Hoher Warenkorbwert (Ø 149 €)
- Nutzen den Kundenservice häufig
- Kommen häufig über SEO und SEA

Cluster 4: Die Zweifler

- Viele Visits im Analysezeitraum (Ø 4 Visits)
- Viele Produktaufrufe (Ø 10 Produkte)
- Viele Artikel pro Bestellung (Ø 4 Artikel)
- Hohe Retourenquote (Ø 31 %)
- Nutzen den Kundenservice häufig
- Mittlerer Warenkorbwert (Ø 84 €)
- Kommen häufig über Retargeting und Social Media

Personalisierung via Clusteranalyse

- Die abgeleiteten Segmente sind nicht nur ein analytisches Konstrukt, um Besucherverhalten besser zu verstehen, sondern können praktisch genutzt werden, um Besucher gezielt anzusprechen und so die Zufriedenheit, Engagement und Umsatz zu steigern.



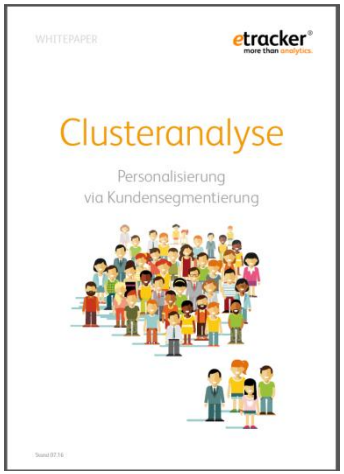
Vorteile von R

Vorteile von R

- Effiziente Datenverarbeitung durch dplyr, data.table, tidyr
- Pearson-Ähnlichkeitsmaß und Ward-Fusionierungsalgorithmus bereits in R implementiert, dokumentiert und getestet
- Aufbereitung der Ergebnisse mit ggplot, shiny

Fragen?

PS: Whitepaper zum Thema Clusteranalyse:



Kontakt

Alexander Kruse
Data Analyst

Tel: +49 40 55 56 59 50
kruse@etracker.com

etracker GmbH
Erste Brunnenstraße 1
20459 Hamburg