

Online-Marketing der Interhyp AG

Analyse von Tracking-Daten

Daniel Fuckner
Markus Vogler
Betreuer: Fabian Scheipl

Statistisches Consulting
Institut für Statistik
Ludwig-Maximilians-Universität München

12.08.2014

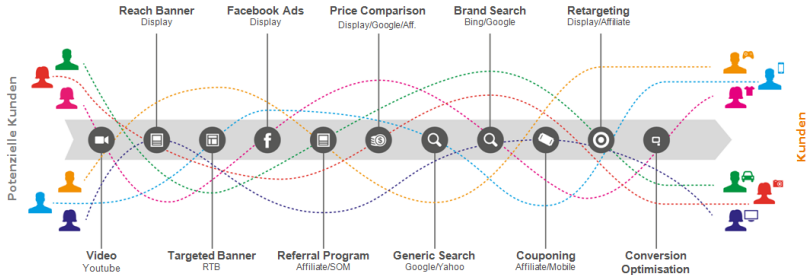
Inhaltsverzeichnis

- 1 Einleitung
- 2 Deskriptive Analyse
- 3 Methoden
- 4 Ergebnisse
- 5 Zusammenfassung

Einleitung

- Interhyp AG ist Vermittler für private Baufinanzierungen
- Primäres Ziel des Marketing ist die Kundenakquise
- Etwa 80% aller Kundenanträge werden online abgeschickt
- Online-Marketing verfügt über verschiedene Kanäle
- Refined Labs GmbH ist verantwortlich für das Online-Tracking der Werbekampagnen der Interhyp AG

Entstehung eines Funnels (Quelle: Interhyp AG)



Unterschiede zwischen konvertierten und nicht-konvertierten Funnels?

Inhalt

- 1 Einleitung
- 2 Deskriptive Analyse
- 3 Methoden
- 4 Ergebnisse
- 5 Zusammenfassung

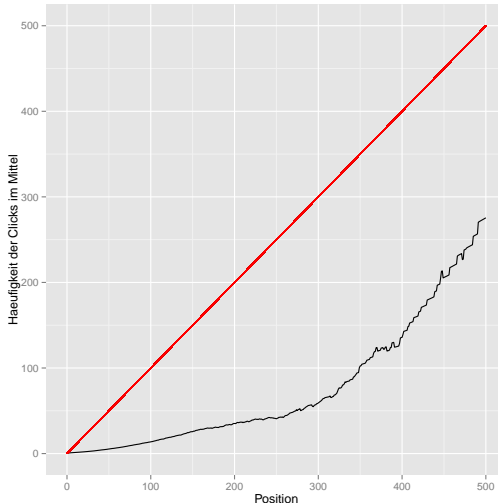
Beispiel für einen Auszug aus der Datenbank

ID	Campaign	Transaction	Position	...
1	Affiliate - Partnerprogramm	0	1	...
1	SEM - Brand	0	2	...
1	Direct	0	3	...
1	Direct	1	4	...
2	Display	0	1	...
2	SEM - Generisch	0	2	...
2	Social Media	0	3	...

Datenlage

- SQL-Dump mit Größe von circa 13 Gigabyte
- Einteilung in konvertierte und nicht-konvertierte Funnels
- Kampagnen in Form einer Baumstruktur organisiert
- Festlegung auf 17 Kategorien
- *Views* liegen in den nicht-konvertierten Funnels nur vor, wenn diese bei einem anderen Kunden der Refined Labs GmbH konvertiert sind
- 297,963 *Clicks* für die konvertierten und 9,550,802 *Clicks* für die nicht-konvertierten Funnels
- Erstellung von Features

clickCount

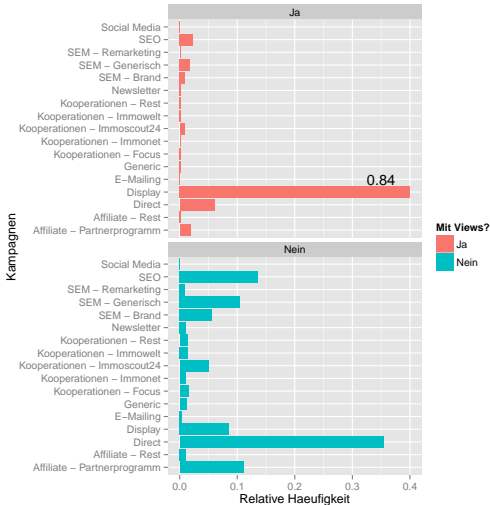


- Anzahl der Clicks bis zur aktuellen Position
- Gemittelt über alle konvertierten Funnels
- Mehr *Views* als *Clicks*

Beschreibung der Kampagnen

Kampagne	Beschreibung
Affiliate - Partnerprogramm	Partner, die Werbemittel einbinden
Affiliate - Rest	Partner, die Zinsvergleich bereitstellen
Direct	Direkte Eingabe von <i>www.interhyp.de</i>
Display	Bannerschaltungen
E-Mailing	Mails an Interessenten, die schon einen Antrag o.ä. gestellt haben
Generic	Unbezahlter Link
Kooperationen - Focus Kooperationen - Immonet Kooperationen - Immoscout24 Kooperationen - Immowelt Kooperationen - Rest	Individuelle Zusammenarbeit mit größeren Partnern
Newsletter	Regelmäßige Rundschreiben
SEM - Brand SEM - Remarketing SEM - Generisch	Bezahlte Suchergebnisse
SEO	Unbezahlte Suchergebnisse
Social Media	<i>facebook</i> und <i>gutefrage.net</i>

campaign



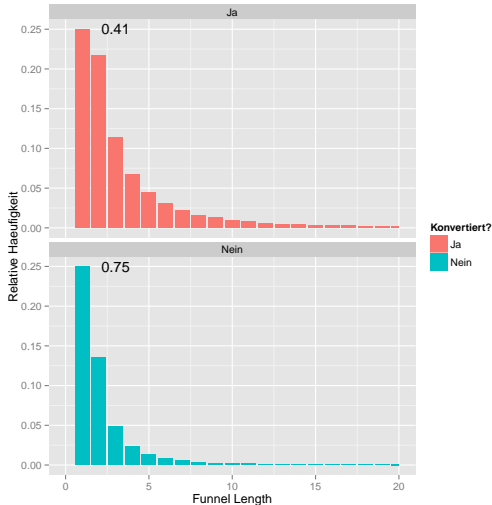
- Hauptsächlich *Display* bei Berücksichtigung der *Views*
- Ausgewogenere Verteilung wenn *Views* gelöscht werden

campaign



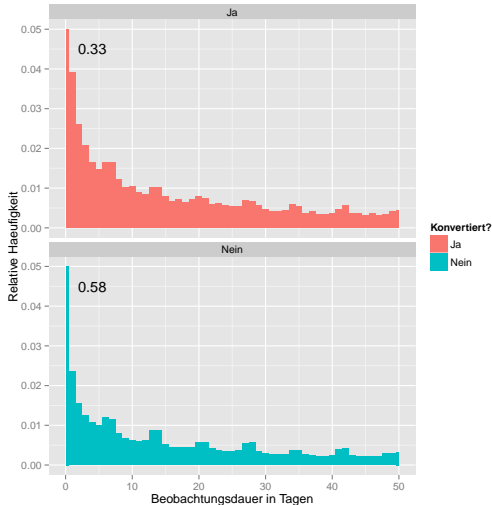
- *Direct* am häufigsten in den konvertierten Funnels
- *Display* und *Affiliate - Partnerprogramm* am häufigsten in den nicht-konvertierten Funnels

funnelLength



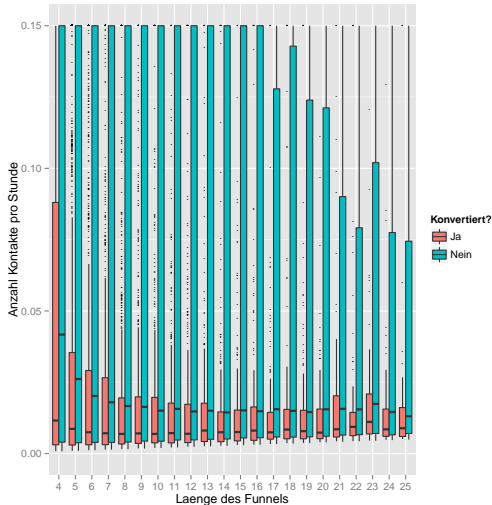
- Anzahl Kontaktpunkte eines Funnels
- Kurze Funnels überwiegen deutlich

timeSinceFirst



- Verstrichene Zeit seit dem ersten Kontaktpunkt
- In der Abbildung wird nur der letzte Kontaktpunkt berücksichtigt
- Funnels mit Länge 1 unberücksichtigt
- *timeSinceLast*: Verstrichene Zeit seit dem vorherigen Kontaktpunkt

freq



- *funnelLength* dividiert durch Gesamt-Beobachtungsdauer in Stunden
- Frequenzen in nicht-konvertierten Funnels höher

Inhalt

- 1 Einleitung
- 2 Deskriptive Analyse
- 3 Methoden**
- 4 Ergebnisse
- 5 Zusammenfassung

- Zeit bis zu einem Ereignis \Rightarrow Konvertierung oder Nicht-Konvertierung bzw. Rechtszensurierung
- Positionen bilden Zeitachse des Modells \Rightarrow Zeitdiskretes Modell
- Stochastic Gradient Boosting mit Stümpfen als Basis-Lerner

- Zielvariable:

$$y_{ip} = \begin{cases} 1 & \text{Beobachtung } i \text{ konvertiert an Position } p \\ 0 & \text{sonst} \end{cases}$$

$$p = 1, \dots, 25, i = 1, \dots, N_p$$

- Hazardrate:

$$\lambda_{ip} = P(y_{ip} = 1 | \text{funnelLength}_i \geq p, x_{ip})$$

- Logit-Modell:

$$y_{ip} | x_{ip} \stackrel{\text{ind}}{\sim} \text{Bin}(1, \lambda_{ip})$$

$$E(y_{ip} | x_{ip}) = P(y_{ip} = 1 | x_{ip}) = \lambda_{ip} = h(f_{ip}) = \frac{\exp(f_{ip})}{1 + \exp(f_{ip})}$$

- Likelihood:

$$L(\lambda_{ip}) = \prod_{i=1}^{N_p} \lambda_{ip}^{y_{ip}} (1 - \lambda_{ip})^{1-y_{ip}}$$

- Log-Likelihood:

$$\begin{aligned} l(\lambda_{ip}) &= \ln(L(\lambda_{ip})) = \sum_{i=1}^{N_p} (y_{ip} \ln(\lambda_{ip}) + (1 - y_{ip}) \ln(1 - \lambda_{ip})) \\ &= \sum_{i=1}^{N_p} (y_{ip} f(x_{ip}) - \ln(1 + \exp(f(x_{ip})))) \end{aligned}$$

- Binomieller Verlust:

$$L(y, f) = -yf + \ln(1 + \exp(f))$$

- Prädiktorfunktion:

$$\begin{aligned} f(x_{ip}) = & f_{weekday,p}(\text{weekday}_{ip}) + \\ & f_{hour,p}(\text{hour}_{ip}) + \\ & f_{campaign,p}(\text{campaign}_{ip}) + \\ & f_{campaignLast,p}(\text{campaign}_{i,p-1}) + \\ & f_{campaignLast2,p}(\text{campaign}_{i,p-2}) + \\ & f_{timeSinceLast,p}(\text{timeSinceLast}_{ip}) + \\ & f_{timeSinceFirst,p}(\text{timeSinceFirst}_{ip}) + \\ & \text{offset}(\hat{\lambda}_{i,p-1}) \end{aligned}$$

Gradient Boosting - Pseudocode

Setze Startwert für $f_{0p}(x_{ip})$

for $m = 1 : n.trees$ **do**

Setze $\lambda_{ip}(x_{ip}) = \frac{\exp(f_{m-1,p}(x_{ip}))}{1 + \exp(f_{m-1,p}(x_{ip}))}$

for $i = 1 : N_p$ **do**

$$r_{imp} = -\frac{\partial L(y_{ip}, f_{m-1,p}(x_{ip}))}{\partial f_{m-1,p}(x_{ip})} = y_{ip} - \lambda_{ip}(x_{ip})$$

end for

$$\theta_{mp} = \arg \min_{\theta} \sum_{i=1}^{N_p} (r_{imp} - h(x_{ip}, \theta))^2$$

$$\beta_{mp} = \arg \min_{\beta} \sum_{i=1}^{N_p} L(y_{ip}, f_{m-1,p}(x_{ip}) + \beta h(x_{ip}, \theta_{mp}))$$

$$f_{mp}(x_{ip}) = f_{m-1,p}(x_{ip}) + \beta_{mp} h(x_{ip}, \theta_{mp})$$

end for

Parameter des Modells

- Trainingsdaten machen Hälfte der gesamten Daten aus - stratifiziert bezüglich Transaction, Campaign, funnelLength
- $n.trees = 3000$
- $cv.folds = 5$
- Shrinkage-Parameter:
$$\mu = 0.01 \Rightarrow f_{mp}(x_{ip}) = f_{m-1,p}(x_{ip}) + \mu\beta_{mp}h(x_{ip}, \theta_{mp})$$
- $interaction.depth = 1$
- $bag.fraction = 0.5 \Rightarrow$ **Stochastic** Gradient Boosting

Output des Modells

- $\hat{f}(x_{ip})$ für jede Beobachtung i und jede Position p

$$\hat{\lambda}_{ip} = \frac{\exp(\hat{f}(x_{ip}))}{1 + \exp(\hat{f}(x_{ip}))}$$

- Relative Wichtigkeit der Features:

$$\hat{l}_{jp} = \sqrt{\frac{1}{M} \sum_{m=1}^{n.trees} \hat{l}_{mp} 1_{jmp}}$$

- Marginale Effekte der Features:

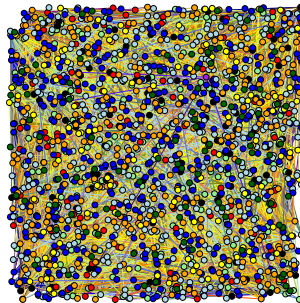
$$\bar{f}_{jp}(x_{jp}) = \frac{1}{N} \sum_{i=1}^{N_p} \hat{f}(x_{jp}, x_{i, \setminus j, p})$$

- ROC-Kurve und AUC

- Menge von Items $I = \{a, b, c, d, e\} \Rightarrow$ Kampagnen
- Datenbank: $[ID1, \langle abcdbaae \rangle]$; $[ID2, \langle edcaa \rangle]$
- 4-Sequenz $s = b \rightarrow b \rightarrow a \rightarrow e$
- Support einer Sequenz: Anteil der IDs, die s unterstützen
- SPADE-Algorithmus findet häufige Sequenzen, deren Support größer als ein festgelegter minimaler Support ist
- Separate Anwendung auf konvertierte und nicht-konvertierte Funnels

- Geordneter Graph $G = (V, E)$ besteht aus Menge V von Knoten und Menge E von Kanten
- Kante $e_i \in E$ besteht aus geordneten Paar von zwei Knoten (v_j, v_k) , wobei $v_j, v_k \in V$
- Startpunkt \rightarrow 17 Kampagnen der ersten Position $\rightarrow Succ_1$, $Fail_1$ und 17 Kampagnen der zweiten Position $\rightarrow Succ_2$, $Fail_2$ und 17 Kampagnen der dritten Position $\rightarrow \dots$
- Kanten sind bezüglich der Anzahl der Nutzer gewichtet
- Relative Ausgänge: relative Häufigkeiten der Kanten, wobei die zugrundeliegende Menge die Summe aller Nutzer ist, die einen Knoten verlassen
- Relative Eingänge: relative Häufigkeiten der Kanten, wobei die zugrundeliegende Menge die Summe aller Nutzer ist, die in einen Knoten gehen

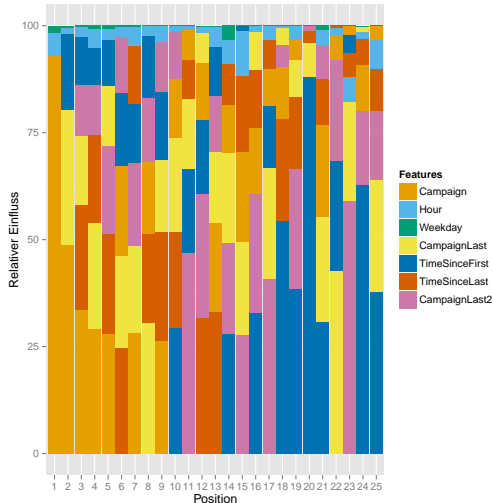
- R-Paket *rgexf* → *gexf*-Datei → *Gephi*
- Berechnung der räumlichen Anordnung der Knoten und Kanten anhand von Algorithmen (z.B. *Force Atlas 2*)
- Manuelle Bearbeitung für die Präsentation von Ergebnissen
- Interaktives Arbeiten mit dem Netzwerk in *Gephi* möglich → Tutorial dazu im Bericht



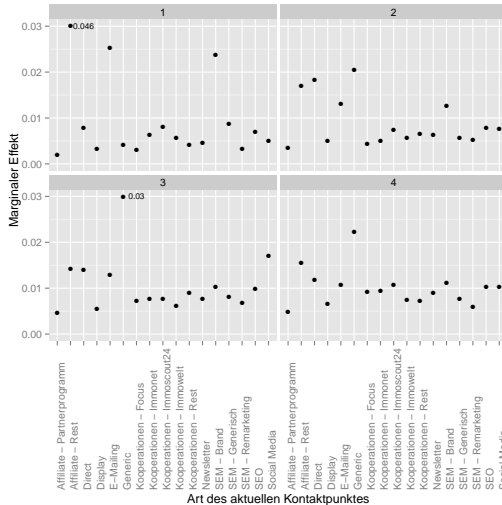
Inhalt

- 1 Einleitung
- 2 Deskriptive Analyse
- 3 Methoden
- 4 Ergebnisse**
- 5 Zusammenfassung

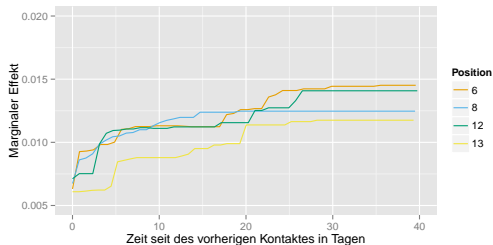
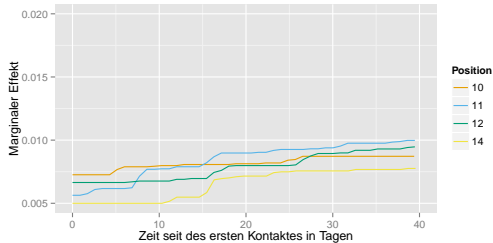
Relative Wichtigkeit der Features



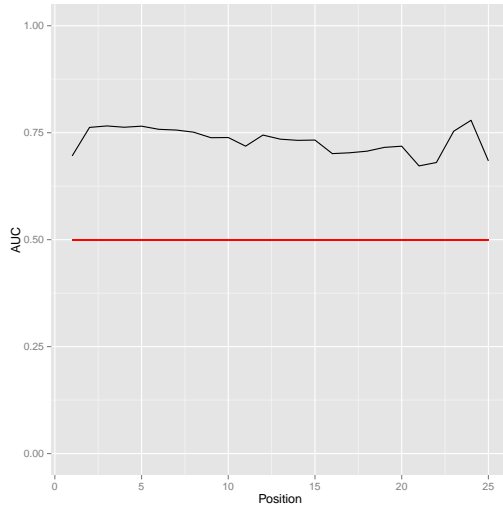
Marginale Effekte - campaign



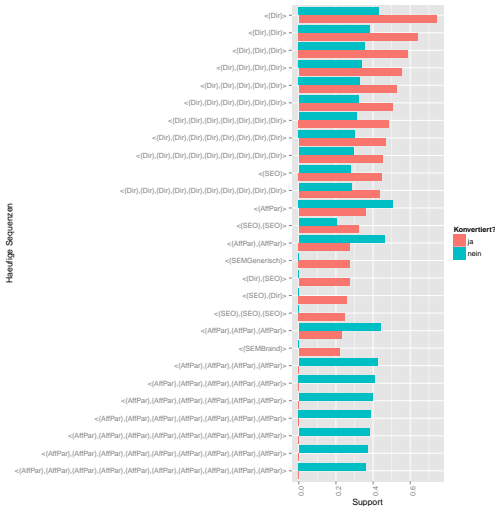
Marginale Effekte - timeSinceFirst & timeSinceLast



AUC

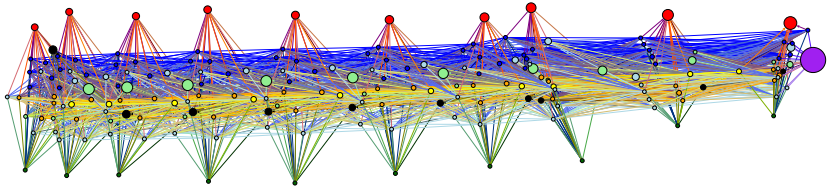


Häufige Sequenzen

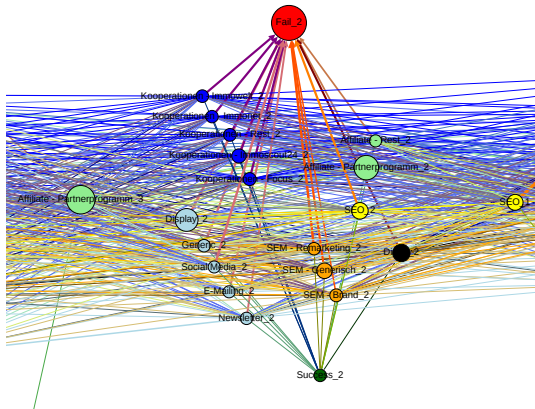


- Kurze Funnels überwiegen
- Nur Funnels mit mindestens 15 Kontaktpunkten berücksichtigt

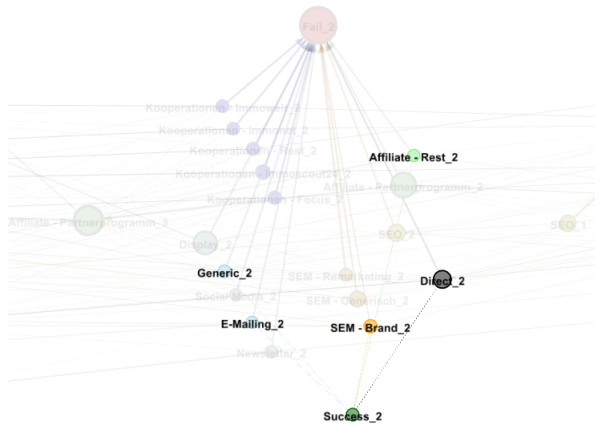
Netzwerk für die ersten 10 Positionen



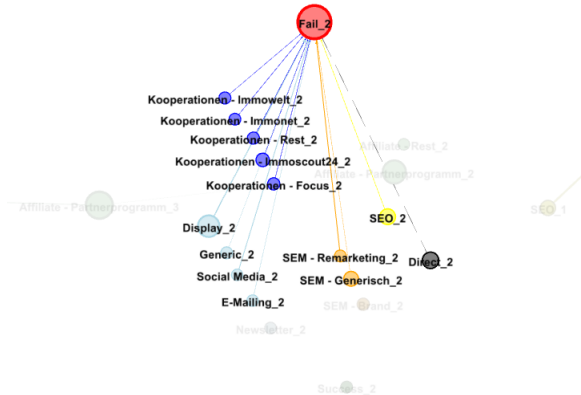
Relative Ausgänge



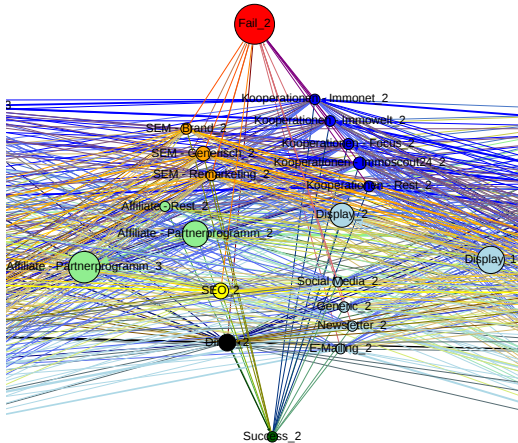
Relative Ausgänge mit Filter 0.02



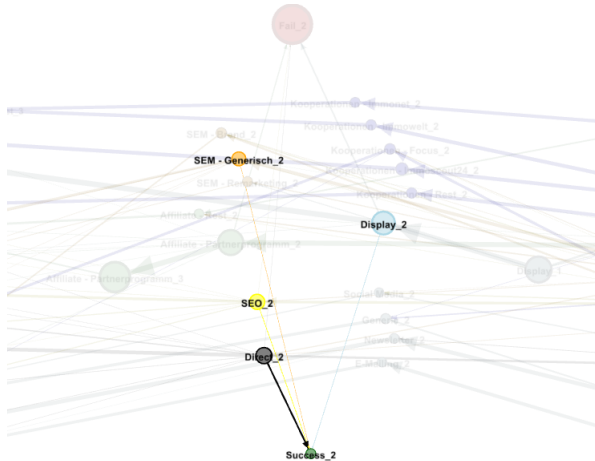
Relative Ausgänge mit Filter 0.5



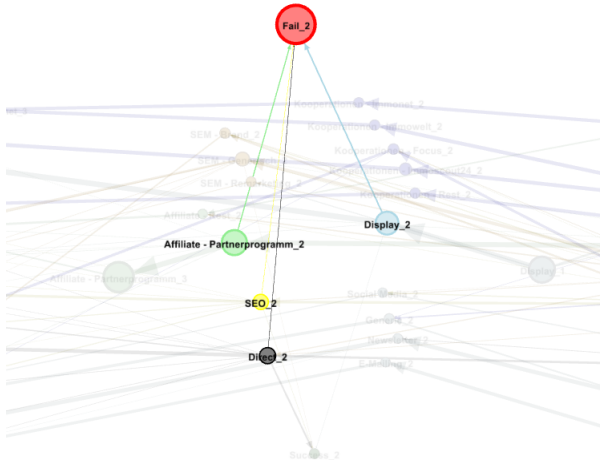
Relative Eingänge



Relative Eingänge mit Filter 0.1



Relative Eingänge mit Filter 0.1



Inhalt

- 1 Einleitung
- 2 Deskriptive Analyse
- 3 Methoden
- 4 Ergebnisse
- 5 Zusammenfassung**

Zusammenfassung der Ergebnisse

- Zeitdiskretes Survival-Modell
 - Klassifikation in konvertierte und nicht-konvertierte Funnels
 - Marginale Effekte der Features
- Sequential Pattern Mining
- Netzwerk
 - Visualisierung der gesamten Daten
 - Bestätigung der Ergebnisse aus Survival-Modell und Sequential Pattern Mining
 - Tutorial zum interaktiven Arbeiten im Bericht

Literatur

-  J. H. Friedman & B. E. Popescu (2005): „Predictive Learning via Rule Ensembles“.
-  R. Agrawal & R. Srikant (1995): „Mining sequential patterns“.
-  M. J. Zaki (2001): „SPADE: An efficient algorithm for mining frequent sequences“.
-  M. Bastian, S. Heymann & M. Jacomy. (2009): „Gephi: An Open Source Software for Exploring and Manipulating Networks“.
-  R-Packages: *gbm*, *rgexf*, *arulesSequences*, *data.table*, *plyr*, *ggplot2*, *doSNOW*, *foreach*.

Vielen Dank für Ihre Aufmerksamkeit!