Markov Chains in R

Divyansh Saxena
Data Scientist @Trainline

“A stochastic model describing a sequence of possible events in which the probability of each event depends only on the state attained in the previous event.”

Deterministic vs Stochastic Model

“A stochastic model describing a sequence of possible events in which the probability of each event depends only on the state attained in the previous event.”

Markov Property

Deterministic Model

In deterministic models, the output of the model is fully determined by the parameter values and the initial conditions.

Eg - Calculation to determine the return on a 5-year investment with an annual interest rate of 7%, compounded monthly.

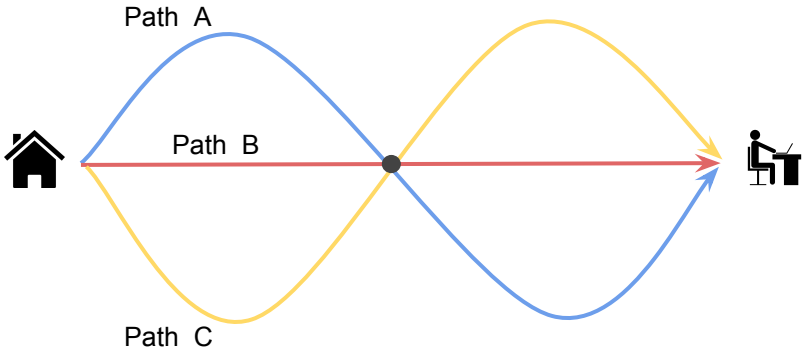
The model is just the equation below

$$F = P(1 + r/m)^{Ym}$$

Stochastic Model

Stochastic models possess some inherent randomness. The same set of parameter values and initial conditions will lead to an ensemble of different outputs.

Eg - A random walk process



Deterministic vs Stochastic Model

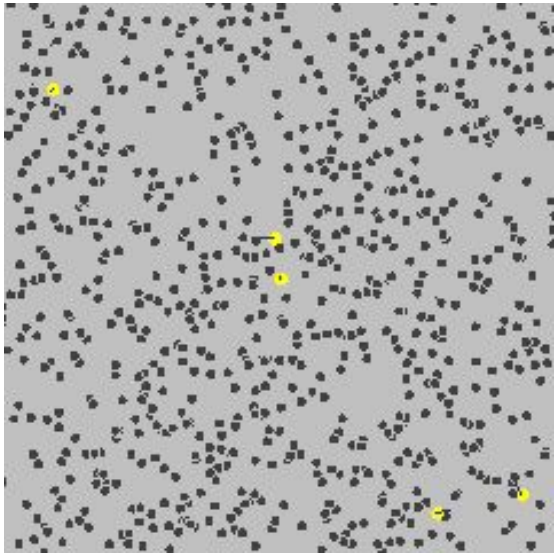
“A stochastic model describing a sequence of possible events in which the probability of each event depends only on the state attained in the previous event.”

Markov Property

Markov Property

In probability theory and statistics, the term Markov property refers to the memoryless property of a stochastic process.

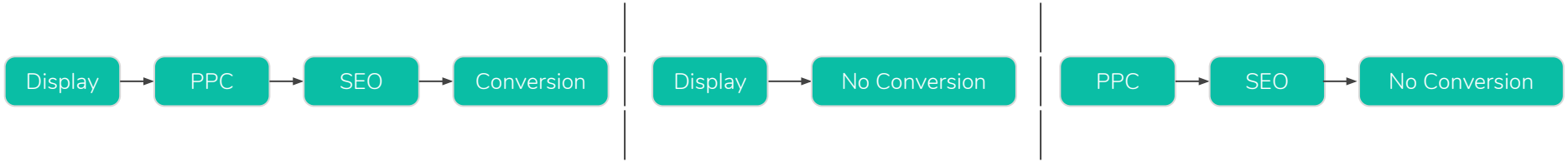
A stochastic process has the Markov Property if the conditional probability distribution of the future states of the process (conditional on both past and present states) depends only upon the present state, not on the sequence of events that preceded it.



$$P(X_n = x_n \mid X_{n-1} = x_{n-1}, \ldots, X_0 = x_0) = P(X_n = x_n \mid X_{n-1} = x_{n-1})$$

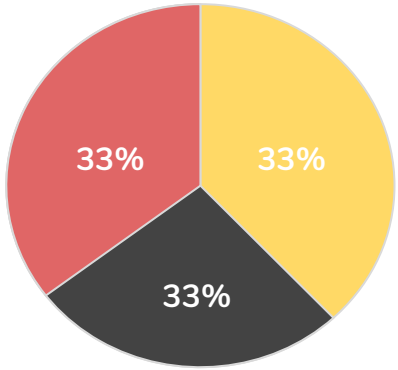
Marketing Attribution

“In marketing, attribution is the identification of a set of user actions ("events" or "touchpoints") that contribute in some manner to a desired outcome, and then the assignment of a value to each of these events.”

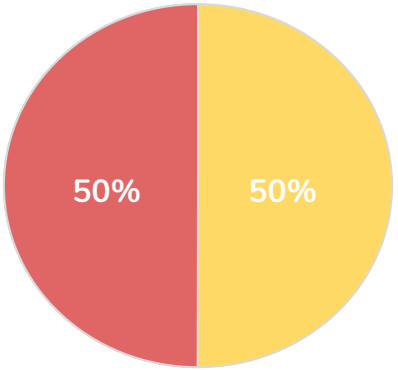


What is the contribution of each unique ad type towards conversion?

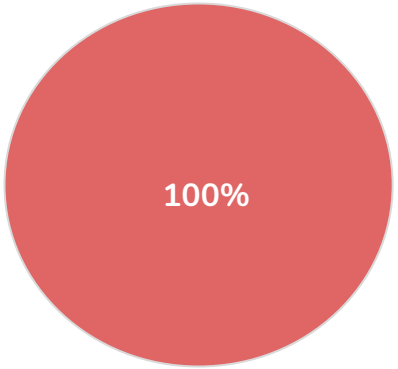
SEO
PPC
Display

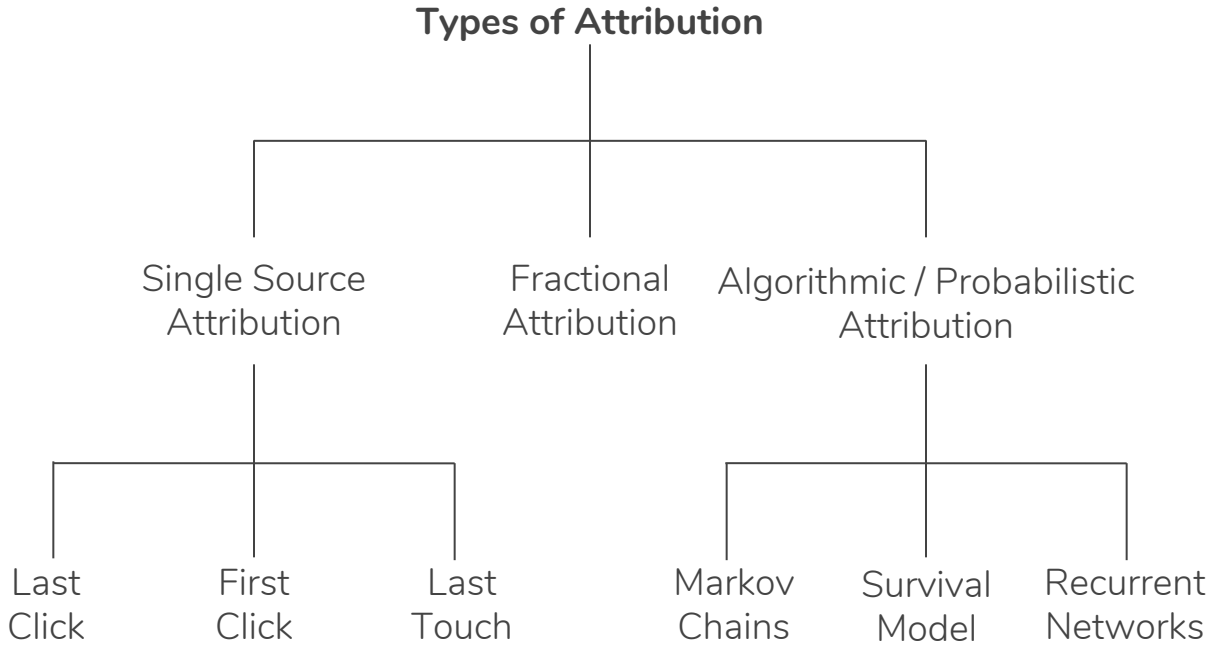


OR

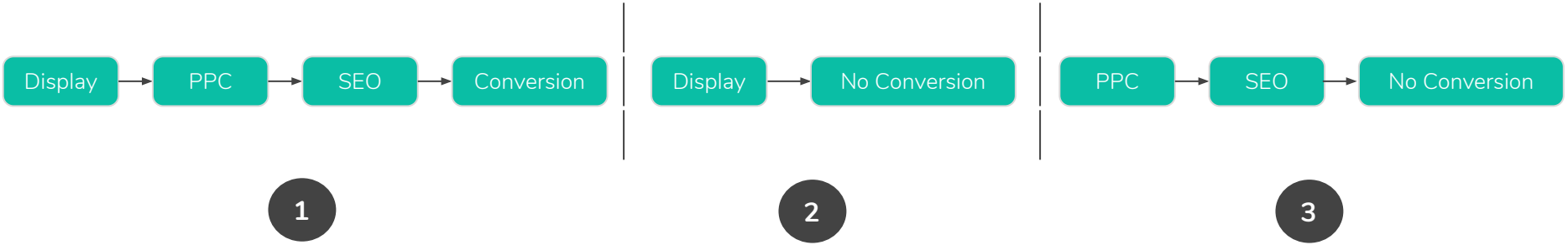


OR

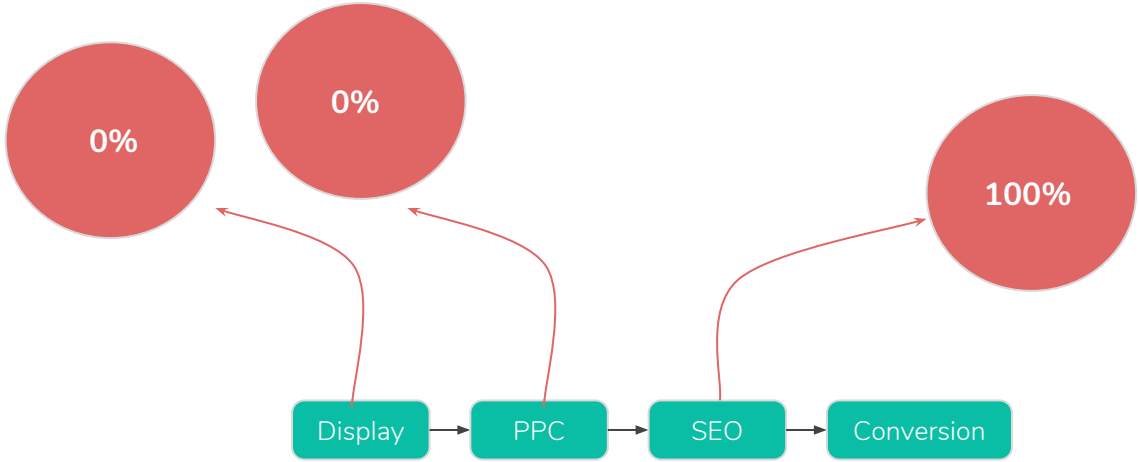



Types of Attribution
Single Source Attribution
Fractional Attribution
Algorithmic / Probabilistic Attribution
Last Click
First Click
Last Touch
Markov Chains
Survival Model
Recurrent Networks

Let’s assume 3 customer journeys



Last Click Method


0%
0%
100%
Display
PPC
SEO
Conversion

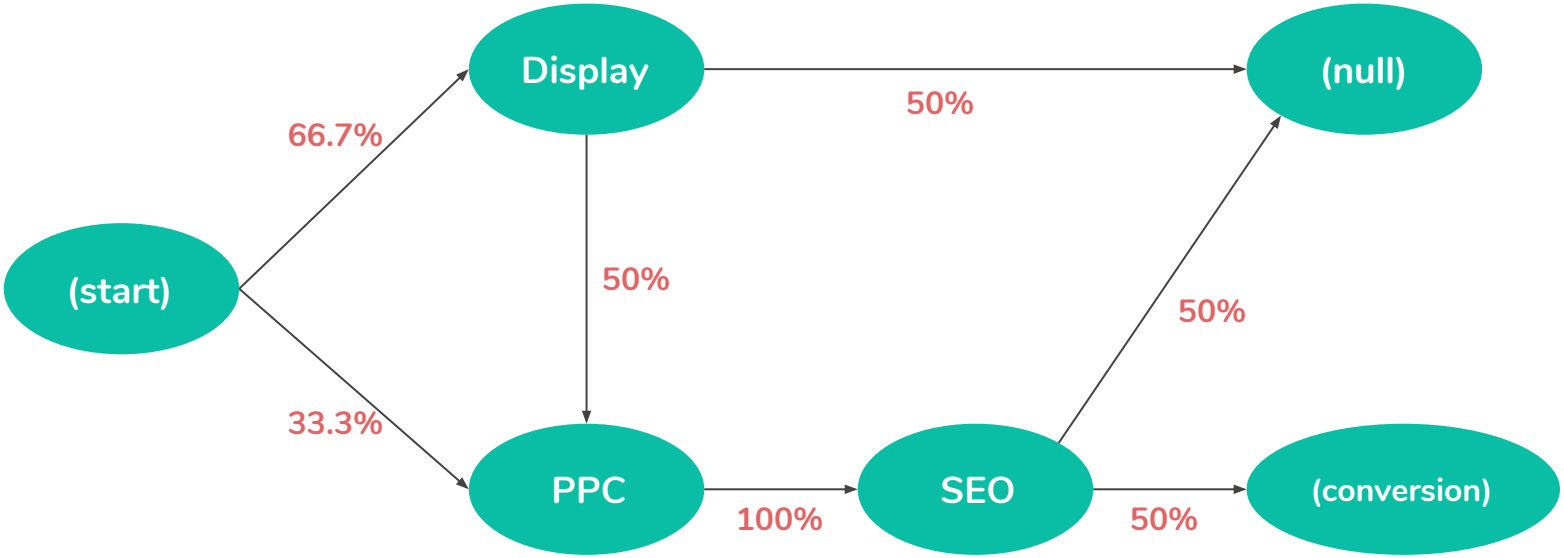
Markov Chain Method

1 - Customer Journey	2 - Transformation	3 - Splitting for pairs
Display > PPC > SEO > Conversion	(start) > Display > PPC > SEO > (conversion)	(start) > Display, Display > PPC, PPC > SEO, SEO > (conversion)
Display	(start) > Display > (null)	(start) > Display, Display > (null)
PPC > SEO	(start) > PPC > SEO > (null)	(start) > PPC, PPC > SEO, SEO > (null)

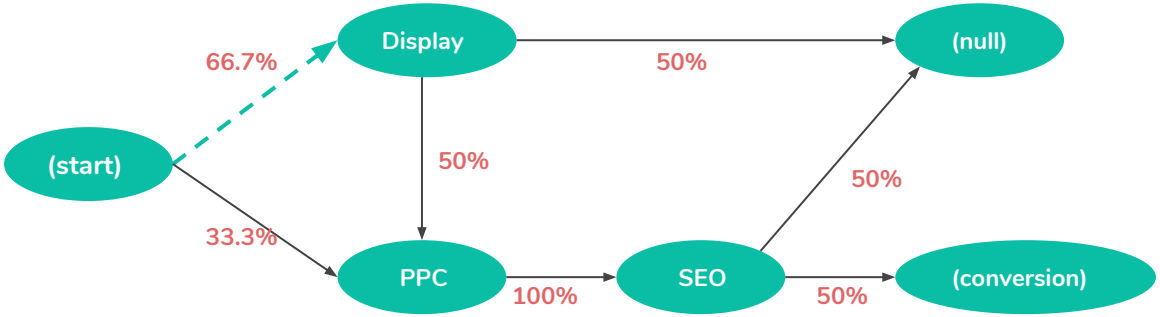
Transition Matrix

from	to	probability	total probability
(start)	Display	⅓	2/3
(start)	Display	⅓	
(start)	PPC	⅓	1/3
Display	PPC	½	1/2
Display	(null)	½	1/2
PPC	SEO	½	1
PPC	SEO	½	
SEO	(conversion)	½	1/2
SEO	(null)	½	1/2

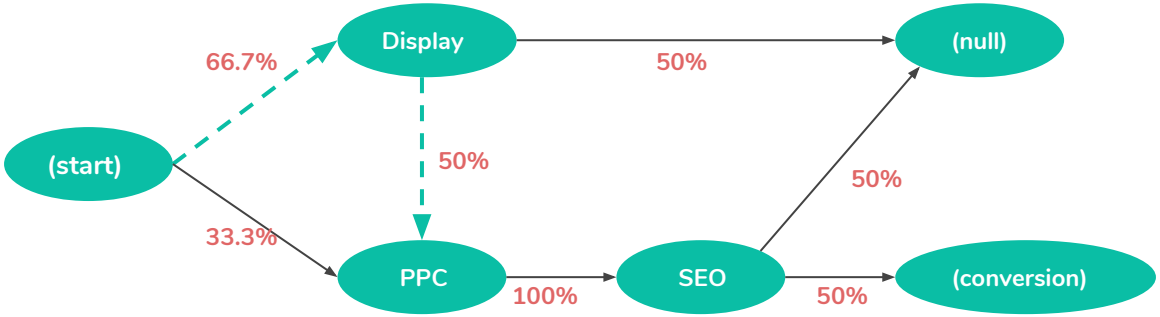
Markov Chain



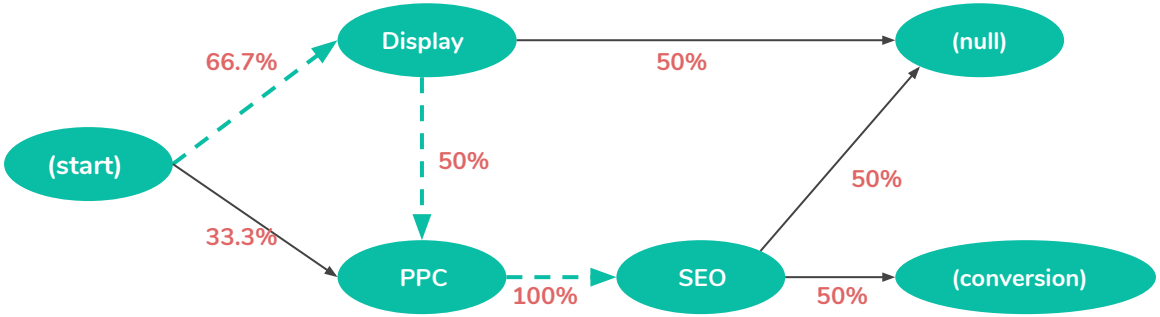
What is the total conversion probability?



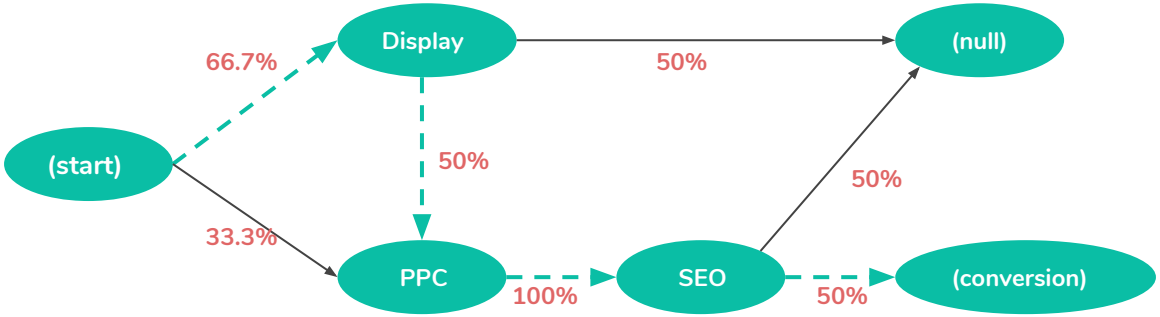
Total Conversion Probability = 66.7%



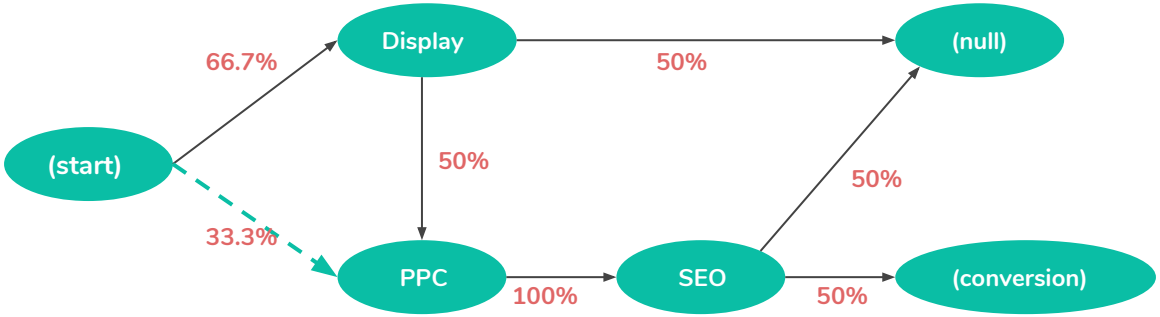
Total Conversion Probability = 66.7% x 50%



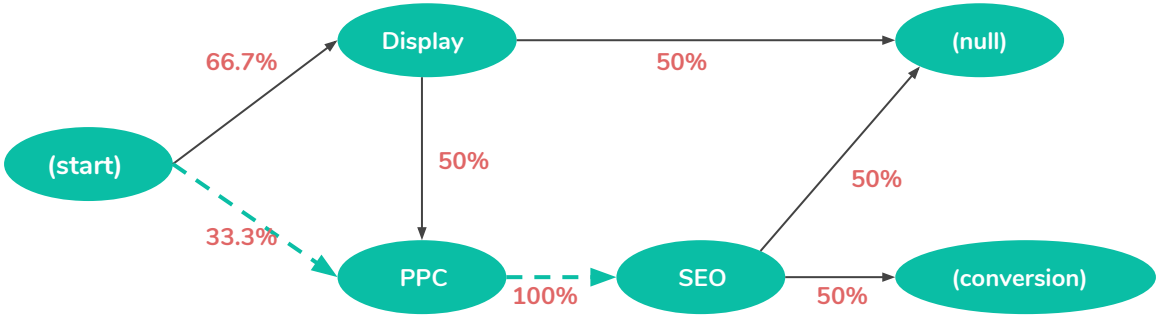
Total Conversion Probability = 66.7% x 50% x 100%



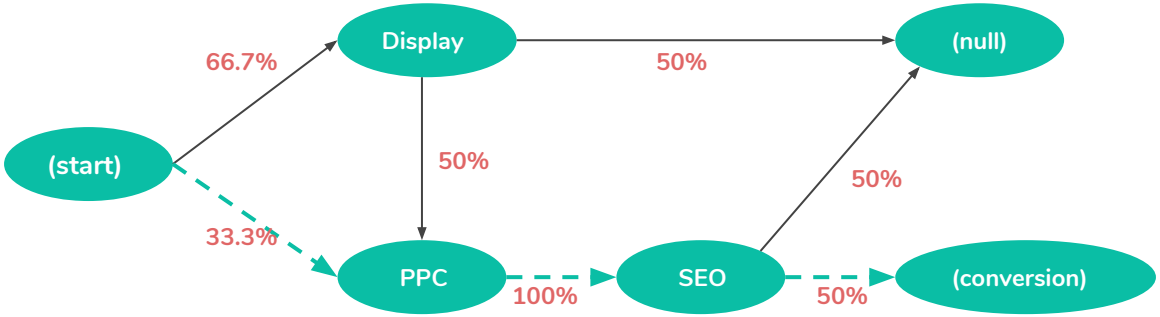
Total Conversion Probability = 66.7% x 50% x 100% x 50%



Total Conversion Probability = (66.7% x 50% x 100% x 50%) + (33.3%)



Total Conversion Probability = (66.7% x 50% x 100% x 50%) + (33.3% x 100%)



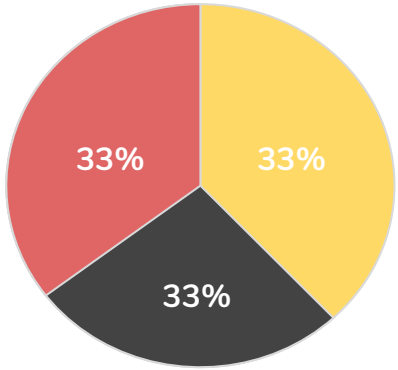
Total Conversion Probability = (66.7% x 50% x 100% x 50%) + (33.3% x 100% x 50%)

= (16.7% + 16.7%)

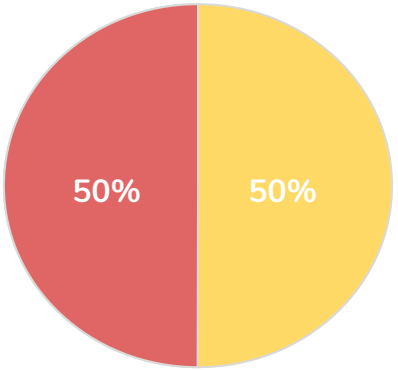
= **33.3%**

What is the contribution of each unique ad type towards conversion?

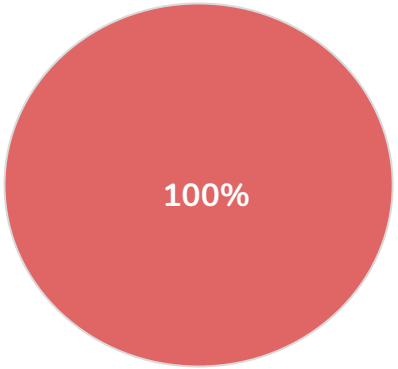
SEO
PPC
Display



OR



OR



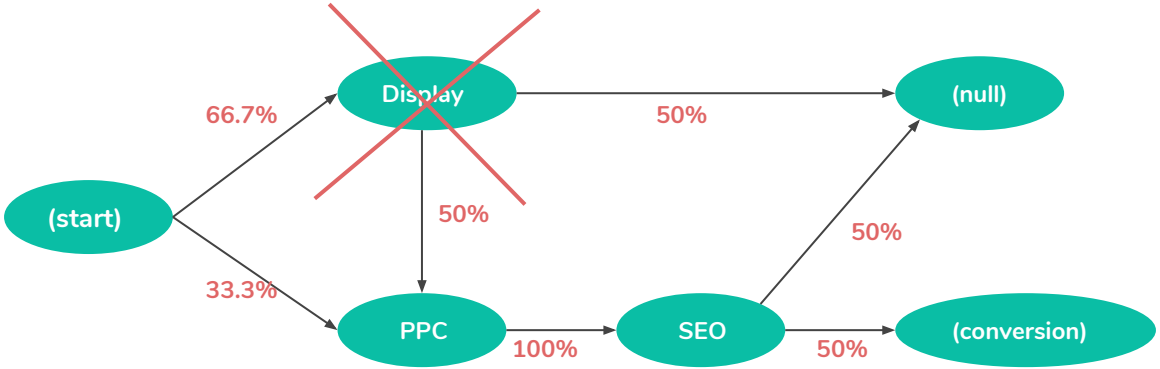
Removal Effects

Removal Effects

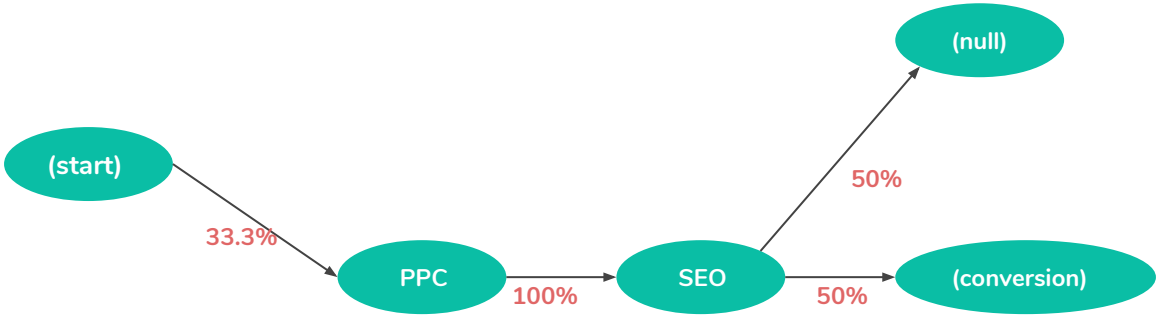
If we were to figure out what is the contribution of Display in our customer's journey from start to end conversion, we will use the principle of removal effect.

Removal effect principle says that if we want to find the contribution of each channel in the customer journey, we can do so by removing each channel and see how many conversions are happening without that channel being in place.

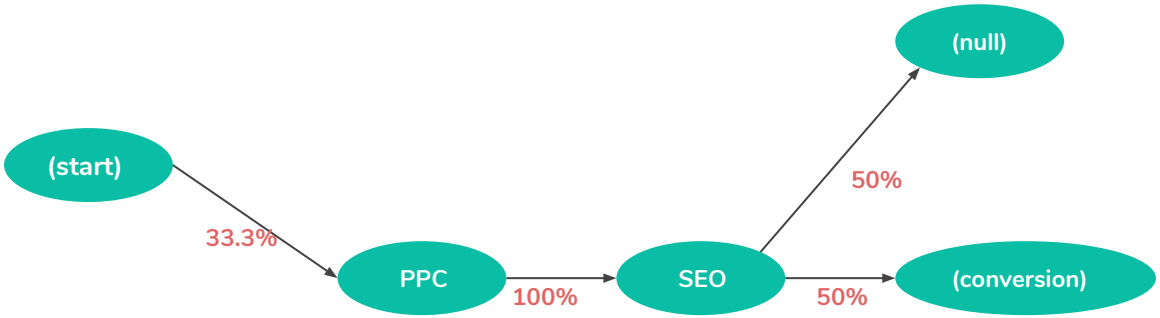
Removal Effect of Display



Removal Effect of Display



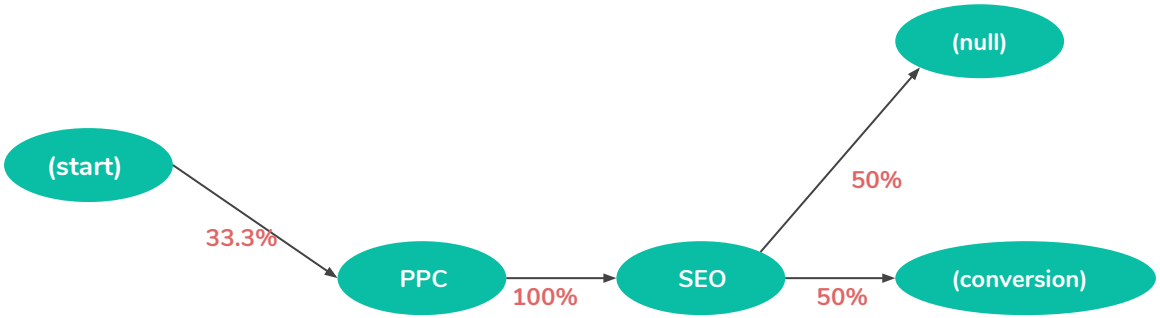
Removal Effect of Display



Conversion probability after removing Display = (33.3% x 100% x 50%)

= **16.7%**

Removal Effect of Display

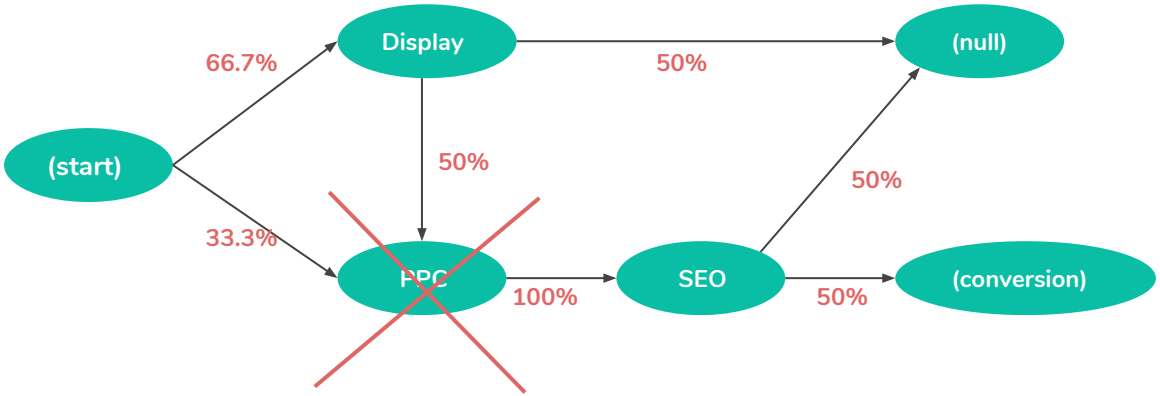


Conversion probability after removing Display = **16.7%**

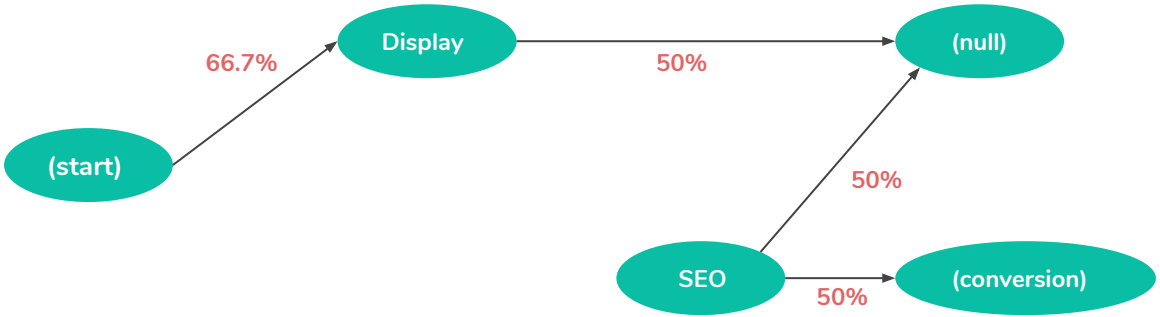
Removal Effect of Display = (33.3% - 16.7%) / 33.3%

= 50%

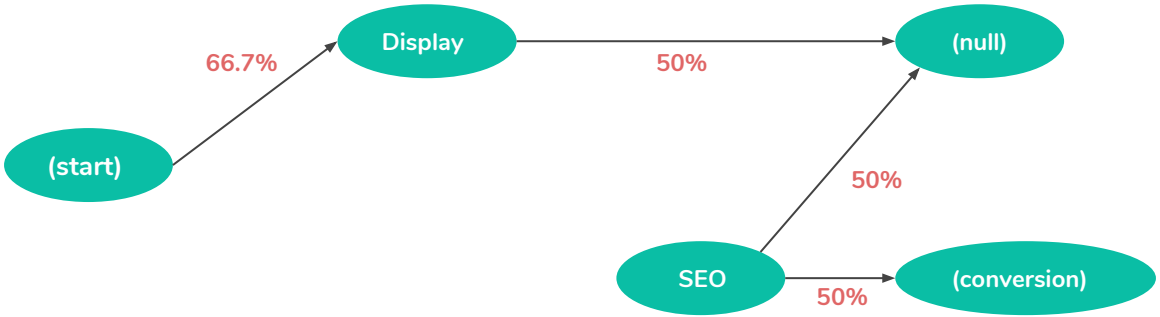
Removal Effect of PPC



Removal Effect of PPC



Removal Effect of PPC

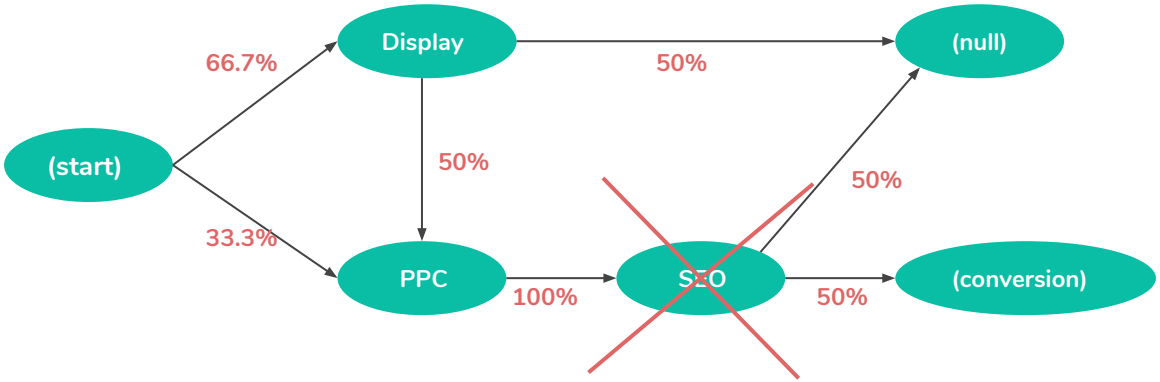


Conversion probability after removing PPC = **0%**

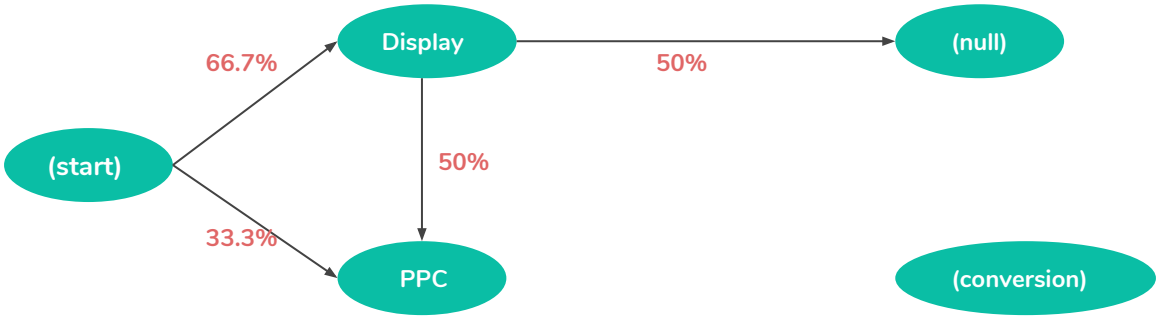
Removal Effect of PPC = (33.3% - 0%) / 33.3%

= 100%

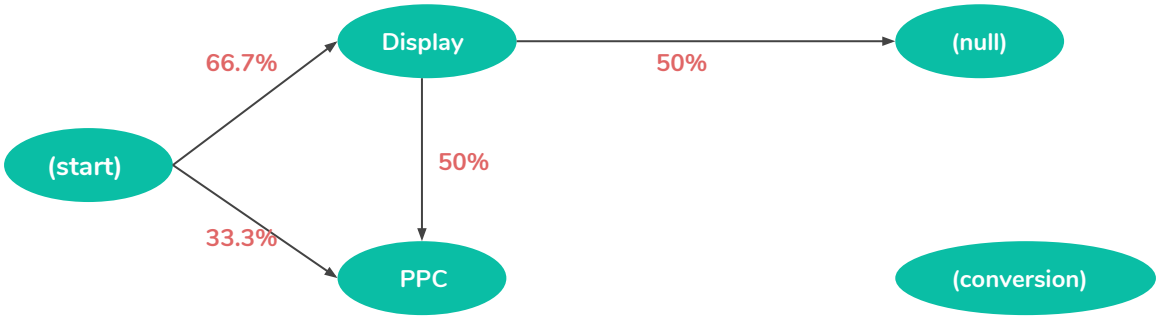
Removal Effect of SEO



Removal Effect of SEO



Removal Effect of SEO

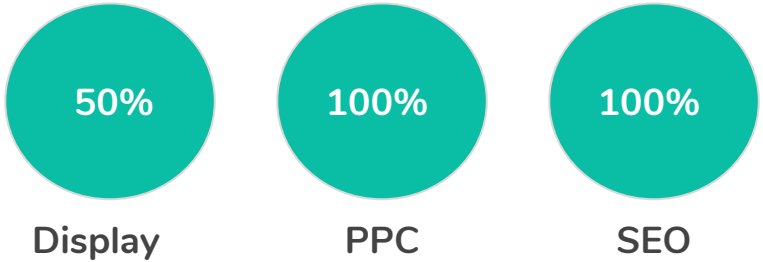


Conversion probability after removing SEO = **0%**

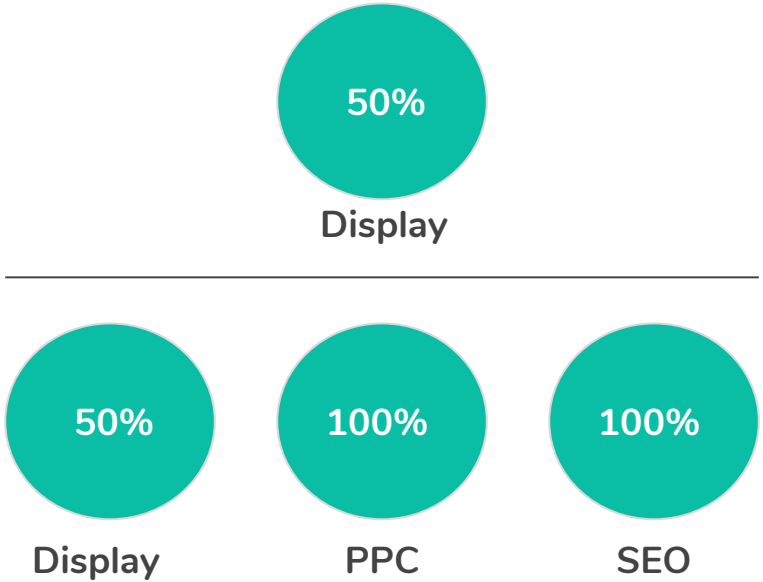
Removal Effect of SEO = (33.3% - 0%) / 33.3%

= 100%

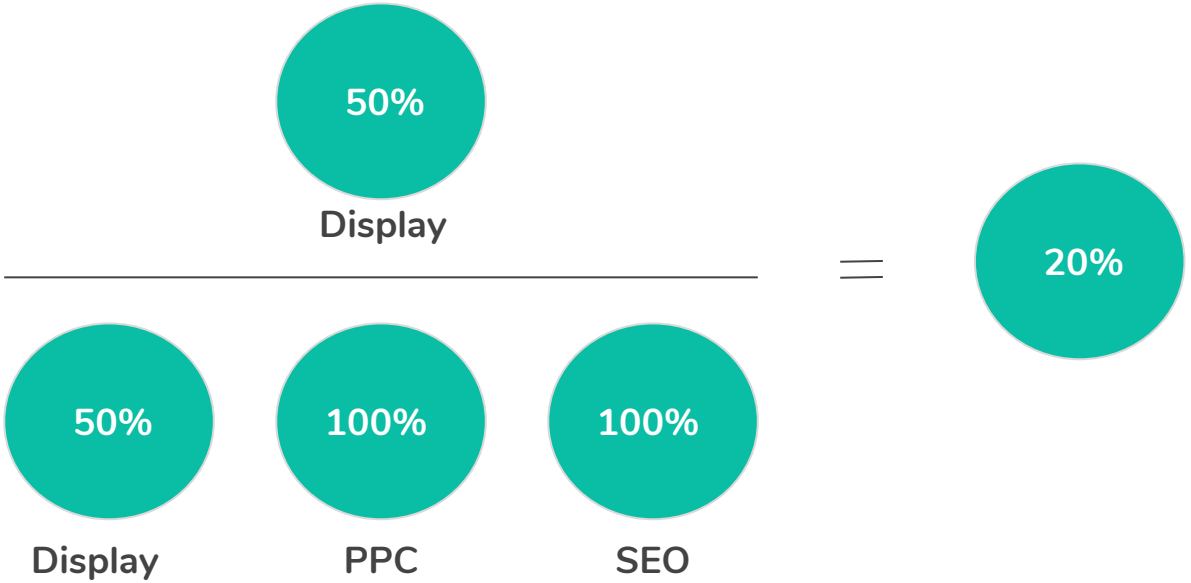
Removal Effect of each ad type / channel



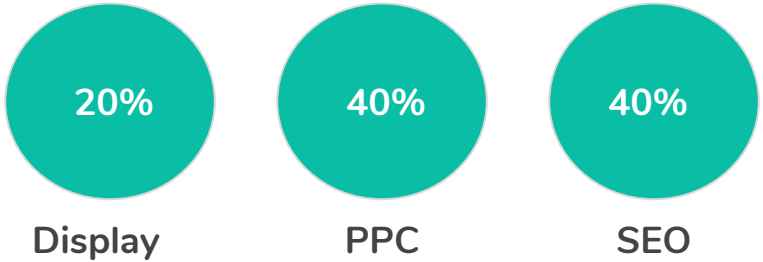
Contribution of Display



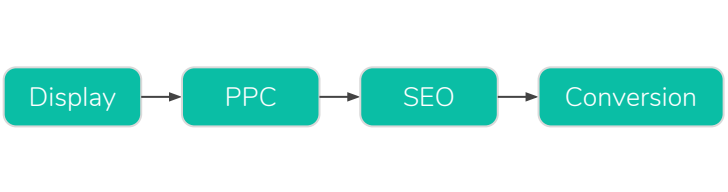
Contribution of Display

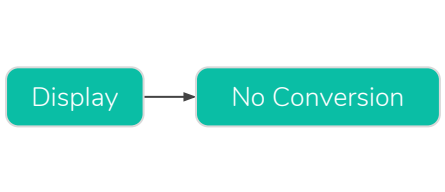


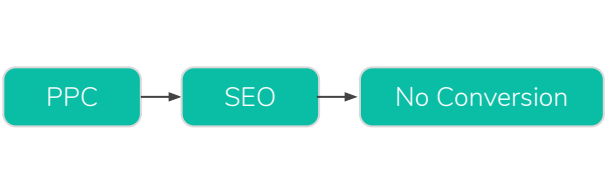
Contribution of each ad type / channel using Markov Chain Method



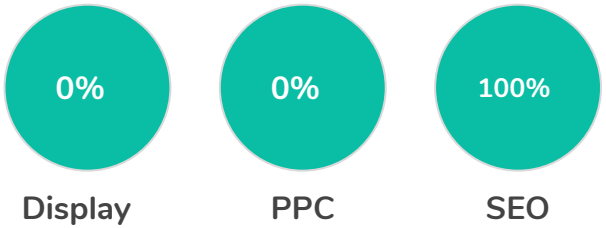
Results



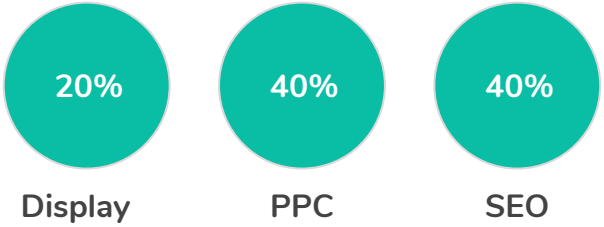




Last Click Method



Markov Chain Method



Markov Chains in R

Package in CRAN - **ChannelAttribution**

Two functions - Markov Model and Heuristic Model

Last Touch
First Touch

```
Console ~/
> library(ChannelAttribution)
> library(dplyr)
> set.seed(354)
> marketing_sample_data <- data.frame(client_id = sample(c(1:1000), 5000, replace = TRUE),
+                                     date = sample(c(1:32), 5000, replace = TRUE),
+                                     channel = sample(c(0:9), 5000, replace = TRUE,
+                                                      prob = c(0.1, 0.15, 0.05, 0.07, 0.11, 0.07, 0.13, 0.1, 0.06, 0.16)))
> marketing_sample_data$date <- as.Date(marketing_sample_data$date, origin = "2015-01-01")
> marketing_sample_data$channel <- paste0('channel_', marketing_sample_data$channel)
```

Filter

	client_id	date	channel
3415	1	2015-01-09	channel_3
349	2	2015-01-09	channel_4
1079	2	2015-02-02	channel_9
2226	2	2015-01-11	channel_9
3120	2	2015-01-14	channel_6
4550	2	2015-01-16	channel_2
4667	2	2015-01-25	channel_1
4964	2	2015-01-30	channel_0
1198	3	2015-01-17	channel_7
1324	3	2015-01-26	channel_9
1991	3	2015-01-22	channel_9
2281	3	2015-01-09	channel_8
2484	3	2015-01-13	channel_8
3639	3	2015-01-26	channel_9
4387	3	2015-01-31	channel_5
4758	3	2015-01-28	channel_0
4938	3	2015-01-23	channel_0
3386	4	2015-01-29	channel_1
4568	4	2015-01-11	channel_9
4831	4	2015-01-23	channel_0
4950	4	2015-01-15	channel_6
359	5	2015-01-14	channel_4
1474	5	2015-01-16	channel_6
2349	5	2015-01-21	channel_4
3641	5	2015-01-09	channel_9

Showing 1 to 25 of 5,000 entries

A customer converted after interacting with 4 channels / ad types.

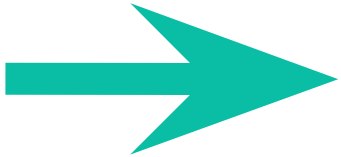
3386	4	2015-01-29	channel_1
4568	4	2015-01-11	channel_9
4831	4	2015-01-23	channel_0
4950	4	2015-01-15	channel_6

channel_9 > channel_6 > channel_0 > channel_1

Console ~/

```
> library(dplyr)
> library(ChannelAttribution)
> set.seed(354)
> marketing_sample_data <- data.frame(client_id = sample(c(1:1000), 5000, replace = TRUE),
+                                     date = sample(c(1:32), 5000, replace = TRUE),
+                                     channel = sample(c(0:9), 5000, replace = TRUE,
+                                                      prob = c(0.1, 0.15, 0.05, 0.07, 0.11, 0.07, 0.13, 0.1, 0.06, 0.16)))
> marketing_sample_data$date <- as.Date(marketing_sample_data$date, origin = "2015-01-01")
> marketing_sample_data$channel <- paste0('channel_', marketing_sample_data$channel)
>
> conversion_path_data <- marketing_sample_data %>%
+     group_by(client_id) %>% arrange(date) %>%
+     summarise(path = paste(channel, collapse = ' > '),
+               # assume that all paths were finished with conversion
+               conv = 1,
+               conv_null = 0) %>%
+     ungroup()
>
```

	client_id	path	conv	conv_null
1	1	channel_3	1	0
2	2	channel_4 > channel_9 > channel_6 > channel_2 > c...	1	0
3	3	channel_8 > channel_8 > channel_7 > channel_9 > c...	1	0
4	4	channel_9 > channel_6 > channel_0 > channel_1	1	0
5	5	channel_7 > channel_9 > channel_4 > channel_6 > c...	1	0
6	6	channel_1 > channel_6 > channel_7 > channel_4 > c...	1	0
7	7	channel_6 > channel_5 > channel_7 > channel_6 > c...	1	0
8	8	channel_8 > channel_5 > channel_6 > channel_5 > c...	1	0
9	9	channel_4 > channel_8	1	0
10	10	channel_1 > channel_1 > channel_0 > channel_4	1	0
11	11	channel_6 > channel_6 > channel_0	1	0
12	12	channel_6 > channel_9 > channel_9 > channel_4 > c...	1	0
13	13	channel_6 > channel_6 > channel_6	1	0
14	14	channel_6 > channel_6 > channel_1 > channel_0	1	0
15	15	channel_7 > channel_2 > channel_4 > channel_7 > c...	1	0
16	16	channel_5 > channel_2 > channel_1 > channel_9 > c...	1	0
17	17	channel_5 > channel_6 > channel_8 > channel_2 > c...	1	0
18	18	channel_6 > channel_8 > channel_6	1	0
19	19	channel_7 > channel_0 > channel_1 > channel_7 > c...	1	0
20	20	channel_6 > channel_1 > channel_7 > channel_5 > c...	1	0
21	21	channel_7 > channel_7 > channel_9 > channel_7 > c...	1	0
22	22	channel_4 > channel_7 > channel_1 > channel_4 > c...	1	0
23	23	channel_1 > channel_8 > channel_3 > channel_9	1	0
24	24	channel_3 > channel_6 > channel_0 > channel_6 > c...	1	0
25	25	channel_0 > channel_5 > channel_1 > channel_6	1	0
26	26	channel_6 > channel_6 > channel_4 > channel_7 > c...	1	0
27	27	channel_8 > channel_4 > channel_7 > channel_4 > c...	1	0



Markov Model in R

```
Console ~/
> # calculating the models (Markov and heuristics)
> model_output <- markov_model(conversion_path_data,
+                              var_path = 'path',
+                              var_conv = 'conv',
+                              var_null = 'conv_null',
+                              out_more = TRUE)
> model_output$
```

```
result
transition_matrix
removal_effects
```

Results

Transition Matrix

Removal Effects

Filter

	channel_name	total_conversions
1	channel_3	71.89903
2	channel_4	111.79065
3	channel_9	135.95758
4	channel_6	125.28111
5	channel_2	57.90731
6	channel_1	137.07329
7	channel_0	98.24545
8	channel_8	66.92221
9	channel_7	100.26660
10	channel_5	84.65677

Results

Transition Matrix

Removal Effects

Filter

	channel_from	channel_to	transition_probability
1	(start)	channel_3	0.06767677
2	(start)	channel_4	0.11414141
3	(start)	channel_8	0.05555556
4	(start)	channel_9	0.14545455
5	(start)	channel_7	0.08989899
6	(start)	channel_1	0.15252525
7	(start)	channel_6	0.14141414
8	(start)	channel_5	0.09090909
9	(start)	channel_0	0.09292929
10	(start)	channel_2	0.04949495
71	channel_0	channel_9	0.11936937
72	channel_0	channel_5	0.08108108
73	channel_0	channel_1	0.14864865
74	channel_0	channel_4	0.09459459
75	channel_0	(conversion)	0.20495495

Results

Transition Matrix

Removal Effects

Filter

	channel_name	removal_effects
1	channel_3	0.254740
2	channel_4	0.396077
3	channel_9	0.481701
4	channel_6	0.443874
5	channel_2	0.205167
6	channel_1	0.485654
7	channel_0	0.348086
8	channel_8	0.237107
9	channel_7	0.355247
10	channel_5	0.299941

Results

Transition Matrix

Removal Effects

Comparison of Markov Model result result with Heuristic Model result

Filter

	channel_name	total_conversions	first_touch	last_touch	linear_touch
1	channel_0	98.29081	92	91	93.95745
2	channel_1	137.25817	151	165	154.29724
3	channel_2	57.99385	49	43	48.67292
4	channel_3	71.73688	67	64	63.09784
5	channel_4	111.73998	113	109	114.06154
6	channel_5	84.52604	90	64	74.63371
7	channel_6	125.08583	140	135	130.97039
8	channel_7	100.39449	89	92	96.29088
9	channel_8	66.97398	55	67	56.31174
10	channel_9	135.99997	144	160	157.70629

Conclusion

Simple
&
Easy to Interpret

Simple
&
Easy to Interpret

Powerfully
Predictive

Simple
&
Easy to Interpret

Powerfully
Predictive

Data - Driven