SAS® Econometrics Econometrics Procedures

MKTATTRIBUTION Procedure

2024.07*

* This document might apply to additional versions of the software. Open this document in SAS Help Center and click on the version in the banner to see all available versions.

SAS® Documentation
July 18, 2024

This document is an individual chapter from *SAS® Econometrics: Econometrics Procedures*.

The correct bibliographic citation for this manual is as follows: SAS Institute Inc. 2024. *SAS® Econometrics: Econometrics Procedures*. Cary, NC: SAS Institute Inc.

SAS® Econometrics: Econometrics Procedures

July 2024

Chapter 21
MKTATTRIBUTION Procedure

Contents

Overview: MKTATTRIBUTION Procedure

The MKTATTRIBUTION procedure supports different types of market attribution models, which have been used to identify which marketing channels drive customer conversions and to help optimize the investment in those channels. Customers are “converted” when they are persuaded by marketing content to take a particular action, such as buying a product, that a business wants them to take. As the array of media channels where businesses can promote their products and services grows, most customers engage with marketing content through multiple channels, such as in television advertising and in ads on social media. Businesses want to know to what degree each channel contributes to their marketing success. This is called the multichannel attribution problem. The Markov attribution model (MAM), one of the market attribution models, approaches the attribution problem in a probabilistic way by using a Markov chain, a particular stochastic process in which the probability distribution of any next state depends only on what the current state is, regardless of any preceding states. The MAM uses a first-order Markov chain to calculate the probability of interaction between pairs of channels in the customer journey and to evaluate the channel’s contribution to customer conversions through the so-called removal effect. The removal effect would enable a business to give a reliable assessment of the marketing contribution of each channel. In addition, the MKTATTRIBUTION procedure provides a number of heuristic attribution models: the first-touch, last-touch, linear, position-based, and time-decay attribution models.

PROC MKTATTRIBUTION requires SAS Cloud Analytic Services (CAS) in order to run, and it does the following: enables you to run on a cluster of machines that distribute the data and the computations, and exploits all the available cores and concurrent threads.

Using CAS Sessions and CAS Engine Librefs

SAS Cloud Analytic Services (CAS) is the analytic server and associated cloud services in SAS Viya. This section describes how to create a CAS session and set up a CAS engine libref that you can use to connect to the CAS session. It assumes that you have a CAS server already available; contact your system administrator if you need help starting and terminating a server. This CAS server is identified by specifying the host on which it runs and the port on which it listens for communications. To simplify your interactions with this CAS server, the host information and port information for the server are stored as SAS option values that are retrieved automatically whenever this CAS server needs to be accessed. You can examine the host and port values for the server at your site by using the following statements:

```
proc options option=(CASHOST CASPORT);
run;
```

In addition to starting a CAS server, your system administrator might also have created a CAS session and a CAS engine libref for your use. You can define your own sessions and CAS engine librefs that connect to the CAS server as shown in the following statements:

```
cas mysess;
libname mylib cas sessref=mysess;
```

The CAS statement creates the CAS session named mysess, and the LIBNAME statement creates the mylib CAS engine libref that you use to connect to this session. It is not necessary to explicitly name the CASHOST and CASPORT of the CAS server in the CAS statement, because these values are retrieved from the corresponding SAS option values.

If you have created the mysess session, you can terminate it by using the TERMINATE option in the CAS statement as follows:

```
cas mysess terminate;
```

For more information about the CAS and LIBNAME statements, see the section “Introduction to Shared Concepts” on page 54 in Chapter 4, “Shared Concepts.”

Getting Started: MKTATTRIBUTION Procedure

This section provides a brief example of using the Markov attribution model that the MKTATTRIBUTION procedure supports.

Consider a market attribution model that has the following transition probability matrix (TPM) for five states $\{channel1, channel2, channel3, conversion, null\}$:

$$\mathbf{A} = \begin{pmatrix} 0.0 & 0.2 & 0.4 & 0.3 & 0.1 \\ 0.1 & 0.0 & 0.4 & 0.5 & 0.0 \\ 0.3 & 0.2 & 0.0 & 0.3 & 0.2 \\ 0.0 & 0.0 & 0.0 & 1.0 & 0.0 \\ 0.0 & 0.0 & 0.0 & 0.0 & 1.0 \end{pmatrix}$$

It also has the following initial state probability vector (ISPV):

$$\pi = \begin{pmatrix} 0.3 \\ 0.3 \\ 0.4 \\ 0.0 \\ 0.0 \end{pmatrix}$$

The following statements simulate the channel visit history, or the customer journey, for 100 customers from the previous model in order to provide test data for PROC MKTATTRIBUTION. The test data table has three columns: customerId is the ID of each customer, t is the time period that the customer was in, and c is the channel that the customer visited at time t. Column c takes value i for channel i, $i = 1, 2,$ and 3, and takes a value of 0 when customer conversion occurs. For a customer, if the last column c value is not 0, this means that the customer is not converted, which also means that the customer is in the null state. Note that, the channel variable takes the natural numbers sequentially from 1 and can't skip any numbers.

```
%let seed=1234;
%let T=10; *the maxsimustep for each customer;
%let nCustomers=100; *the number of customers;

data simu(keep = customerId t c);

  call streaminit(&seed);

  array p0[5] _temporary_ (0.3, 0.3, 0.4, 0, 0.0);
  array p1[5] _temporary_ (0, 0.2, 0.4, 0.3, 0.1);
  array p2[5] _temporary_ (0.1, 0, 0.4, 0.5, 0.0);
  array p3[5] _temporary_ (0.3, 0.2, 0, 0.3, 0.2);
  do customerId = 1 to &nCustomers;
    c_last=0;
    t=0;
    c=1;
    do while(t < &T. and 0<c<=3);
```

```
        c_last=c;
        t+1;
        if t=1 then
          c = rand("Table", of p0[*]);
        else do;
           if c_last=1 then
             c = rand("Table", of p1[*]);
           else if c_last=2 then
             c = rand("Table", of p2[*]);
           else if c_last=3 then
             c = rand("Table", of p3[*]);
        end;
        if c=4 then c=0;
        if c ne 5 then
          output;
      end; *for each customer;
    end; *for all simulations;
run;

proc print data=simu (obs=10) noobs; run;

data mylib.marketData; set simu; run;
```

The first 10 observations of the simulated data are shown in Figure 21.1.

Figure 21.1 Simulated Data

customerId	t	c
1	1	3
1	2	0
2	1	1
2	2	3
3	1	1
3	2	3
3	3	0
4	1	1
4	2	3
4	3	1

The following statements estimate the Markov attribution model that has three channels:

```
proc mktattribution data=mylib.marketData;
   id section=customerId time=t;
   model c / nchannel=3;
run;
```

The model information and number of observations are shown in Figure 21.2.

Figure 21.2 Model Information and Number of Observations

The MKTATTRIBUTION Procedure

Model Information	
Model Type	Markov Attribution
Number of Channels	3

Number of Observations	288
Number of Missing Observations	0

The estimates of the initial state probability vector are shown in Output 21.3.

Figure 21.3 Estimates of Initial State Probability Vector

Initial State Probability Vector

State	Estimate
Channel 1	0.32000
Channel 2	0.30000
Channel 3	0.38000
Conversion	0.00000
Null	0.00000

The estimated transition probability matrix is shown in Output 21.4.

Figure 21.4 Estimated Transition Probability Matrix

Estimated Transition Probability Matrix

State	Channel 1	Channel 2	Channel 3	Conversion	Null
Channel 1	0.00000	0.22388	0.47761	0.22388	0.07463
Channel 2	0.10345	0.00000	0.39655	0.50000	0.00000
Channel 3	0.31183	0.13978	0.00000	0.27957	0.26882
Conversion	0.00000	0.00000	0.00000	1.00000	0.00000
Null	0.00000	0.00000	0.00000	0.00000	1.00000

The conversion rates are shown in Output 21.5.

Figure 21.5 Conversion Rates

Conversion Rates

Method	Rate
Data Based	0.70000
Markov Chain	0.70000

The value in the Data Based row is calculated from the input data. The value in the Markov Chain row is calculated by the estimated TPM. The removal effect of a channel is the decrease in the probability of overall conversion if the channel were removed.

The removal effects are shown in Output 21.6.

Figure 21.6 Removal Effects

Removal Effects	
Channel	Effect
1	0.36540
2	0.40962
3	0.42932

The contributions are the normalized removal effects to indicate each channel's contribution to the conversion. The contributions of each channel through the Markov attribution model and the other three heuristic attribution models are shown in Output 21.7.

Figure 21.7 Channel Contributions

Channel Contributions						
Channel	Markov Chain	First Touch	Last Touch	Linear	Position Based	Time Decay
1	0.30340	0.30000	0.21429	0.27088	0.26352	0.26259
2	0.34012	0.35714	0.41429	0.36942	0.37595	0.37726
3	0.35648	0.34286	0.37143	0.35969	0.36052	0.36015

Syntax: MKTATTRIBUTION Procedure

PROC MKTATTRIBUTION DATA=*libref.data-table* **;**
  ID TIME=*variable* **SECTION=***variable* **;**
  MODEL *channel-variable / * **NCHANNEL=***number* **HALFLIFE=***number* **;**
  OUTPUT < *options* > **;**

Functional Summary

The statements and options available in the MKTATTRIBUTION procedure are summarized in Table 21.1.

Table 21.1 Functional Summary

Description	Statement	Option
Input Data Table Options		
Specifies the input data table	PROC MKTATTRIBUTION	DATA=
Output Data Table Options		
Writes the contribution estimates to the specified output data table	OUTPUT	OUTCONTRIBUTION=
Writes the removal estimates to the specified output data table	OUTPUT	OUTREMOVAL=
Writes the estimated TPM to the specified output data table	OUTPUT	OUTTPM=

Table 21.1 *continued*

Description	Statement	Option
ID Variable		
Specifies the variable that identifies the section	ID	SECTION=
Specifies the variable that identifies the time or sequence	ID	TIME=
Model Options		
Specifies the half-life of the decaying attribution value	MODEL	HALFLIFE=
Specifies the number of channels	MODEL	NCHANNEL=

PROC MKTATTRIBUTION Statement

PROC MKTATTRIBUTION DATA=*libref.data-table* **;**

The PROC MKTATTRIBUTION statement invokes the MKTATTRIBUTION procedure. You must specify the following *option*:

DATA=*libref.data-table*

names the input data table for PROC MKTATTRIBUTION to use. *libref.data-table* is a two-level name, where

libref refers to a collection of information that is defined in the LIBNAME statement and includes the library, which includes a path to the data, and a session identifier, which defaults to the active session but which can be explicitly defined in the LIBNAME statement. For more information about *libref*, see the section “Using CAS Sessions and CAS Engine Librefs” on page 1378.

data-table specifies the name of the input data table.

ID Statement

ID TIME=*variable* **SECTION=***variable* **;**

The ID statement identifies observations in the input data table by specifying two variables for the cross-sectional time series data. That is, for each observation, the combination of the two variables' values must be unique.

You must specify the following options:

TIME=*variable*
: specifies the temporal or sequential order of the observations. The *variable* cannot have missing values.

SECTION=*variable*
: identifies the section or customer ID of each observation. The *variable* cannot have missing values.

MODEL Statement

MODEL *channel-variable* < */ options* > **;**

The MODEL statement specifies the channel variable and the number of channels for the market attribution model. Only one MODEL statement is allowed.

You can specify the following *options* after a forward slash (/):

HALFLIFE=*number*
: specifies the half-life the decaying attribution value for the time-decay attribution model. The *number* must be a positive number. By default, the value of the HALFLIFE= option is 1.

NCHANNEL=*number*
: specifies the number of channels for the market attribution model. The *number* must be a positive integer. The *number* must be greater than the value the channel variable takes. If you specify a *number* greater than the maximum value of the channel variable in the input data set, the procedure produces useless information in the TPM. If you specify a *number* less than the maximum value of the channel variable in the input data set, the procedure stops running. By default, the value of the NCHANNEL= option is the maximum value of the channel variable in the input data set.

OUTPUT Statement

OUTPUT < *options* > **;**

The OUTPUT statement saves the estimation results to the specified output data tables. You can specify the following options:

OUTCONTRIBUTION=*libref.data-table*

writes the contribution results to the specified output data table. *libref.data-table* is a two-level name, where *libref* refers to the library, and *data-table* specifies the name of the output data table. For more information about this two-level name, see the DATA= option and the section "Using CAS Sessions and CAS Engine Librefs" on page 1378.

OUTREMOVAL=*libref.data-table*

writes the removal effects to the specified output data table. *libref.data-table* is a two-level name, where *libref* refers to the library, and *data-table* specifies the name of the output data table. For more information about this two-level name, see the DATA= option and the section "Using CAS Sessions and CAS Engine Librefs" on page 1378.

OUTTPM=*libref.data-table*

writes the estimated transition probability matrix to the specified output data table. *libref.data-table* is a two-level name, where *libref* refers to the library, and *data-table* specifies the name of the output data table. For more information about this two-level name, see the DATA= option and the section "Using CAS Sessions and CAS Engine Librefs" on page 1378.

Details: MKTATTRIBUTION Procedure

Markov Attribution Model

In the Markov attribution model (MAM), customer journeys are represented as Markov chains. A Markov chain is defined by a set of states $\mathbf{S}_t \in \{1, \ldots, K\}$, where t is the time index and K is the number of states; an initial state probability vector (ISPV); and a transition probability matrix (TPM). The ISPV for the MAM can be expressed as a $K \times 1$ vector π, where an element π_i of π denotes the probability of moving from the starting position to the state i; that is,

$$\pi_i = p(\mathbf{S}_1 = i), i = 1, \ldots, K$$

The TPM for the MAM can be expressed as a $K \times K$ matrix $\mathbf{A}$, where an element a_{ij} of $\mathbf{A}$ denotes the probability of moving from past state i to current state j; that is,

$$a_{ij} = p(\mathbf{S}_t = j | \mathbf{S}_{t-1} = i)$$

For example, if a business uses three different channels, C1, C2, and C3, in its online marketing strategy, the model would include three states, C1, C2, and C3. Additionally, all Markov chains contain two special states:

a conversion state that represents a successful conversion, and a null state for customer journeys that have not ended in a conversion during the observation period. The full set of states S in this example would therefore look like this: $\mathbf{S} = \{C1, C2, C3, conversion, null\}$.

The number of states in the Markov chain is the number of marketing channels plus two states: conversion state and null state. If you assume that the number of marketing channels is N, then $K = N + 2$.

If a customer journey ends in a conversion state, the last channel in the journey is connected to the conversion state; otherwise it leads to the null state. For modeling reasons, the conversion state and the null state always lead to themselves and are considered absorbing states.

The conversion rate is defined as the number of conversions divided by the sum of the number of conversions and the number of null results. The removal effect of a channel is defined as the decrease in probability of reaching the conversion state when you remove the channel from the customer's journey. The normalized removal effect is well suited to measuring the contribution of each channel. The contribution per state as a percentage of the sum of all removal effects is reported (excluding the special conversion and null states).

Heuristic Attribution Models

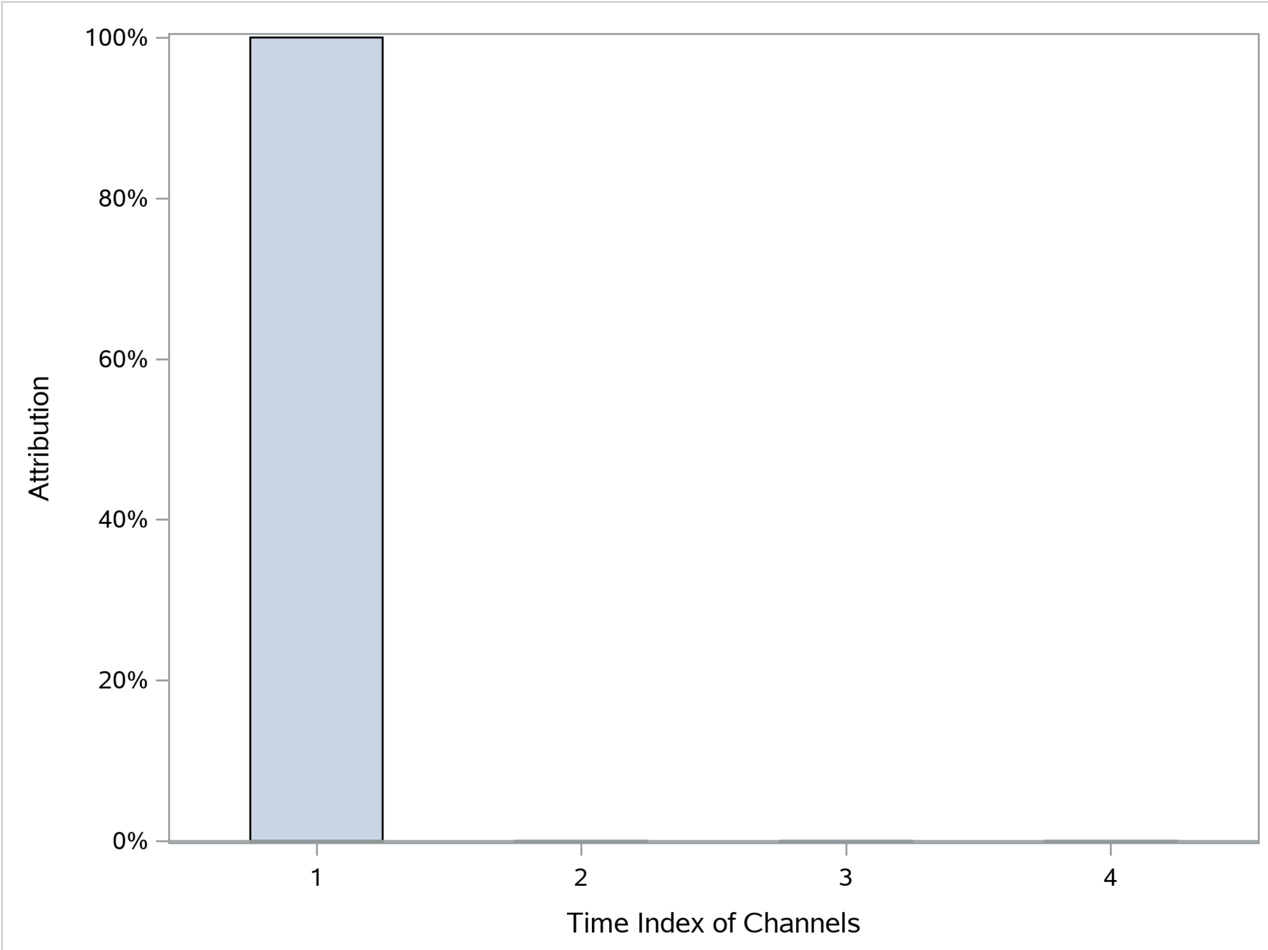
In addition to providing the Markov attribution model, the PROC MKTATTRIBUTION provides several heuristic attribution models: the first-touch, last-touch, linear, position-based, and time-decay attribution models. The first-touch attribution model attributes all credit for the conversion to the first-visited channel in the customer journey. The last-touch attribution model attributes all credit for the conversion to the last-visited channel. The linear attribution model distributes the credit for the conversion equally across all visited channels in the path. The position-based attribution model attributes 40% of credit to the first-visited channel, 40% to the last-visited channel, and the remaining 20% credit equally to the channels in between, if there are more than two visits in the journey. Otherwise, the position-based attribution model is the same as the linear attribution model. The time-decay attribution model attributes the credit increasingly to the channel that is closer to the conversion.

Assuming that there are n customer journeys, the number of conversions is m, where $m \leq n$. $C_j(p)$ is the pth channel in customer j's journey C_j with length l_j, $p = 1, \ldots, l_j$, and $j = 1, \ldots, n$. Let $f(C_j(p) = i|conversion) = 1, i = 1, \ldots, K$, where K is the number of channels, if $C_j(p) = i$ and conversion occurs at the end of this journey; otherwise it equals 0. The contribution to channel i by the first-touch attribution model is calculated by

$$contribution(i) = \frac{\sum_{j=1}^{n} f(C_j(1) = i|conversion)}{m}$$

The bar chart in Output 21.8 shows how the first-touch attribution model distributes the contributions for a journey that visits four channels.

Figure 21.8 Bar Chart for the First-Touch Attribution Model

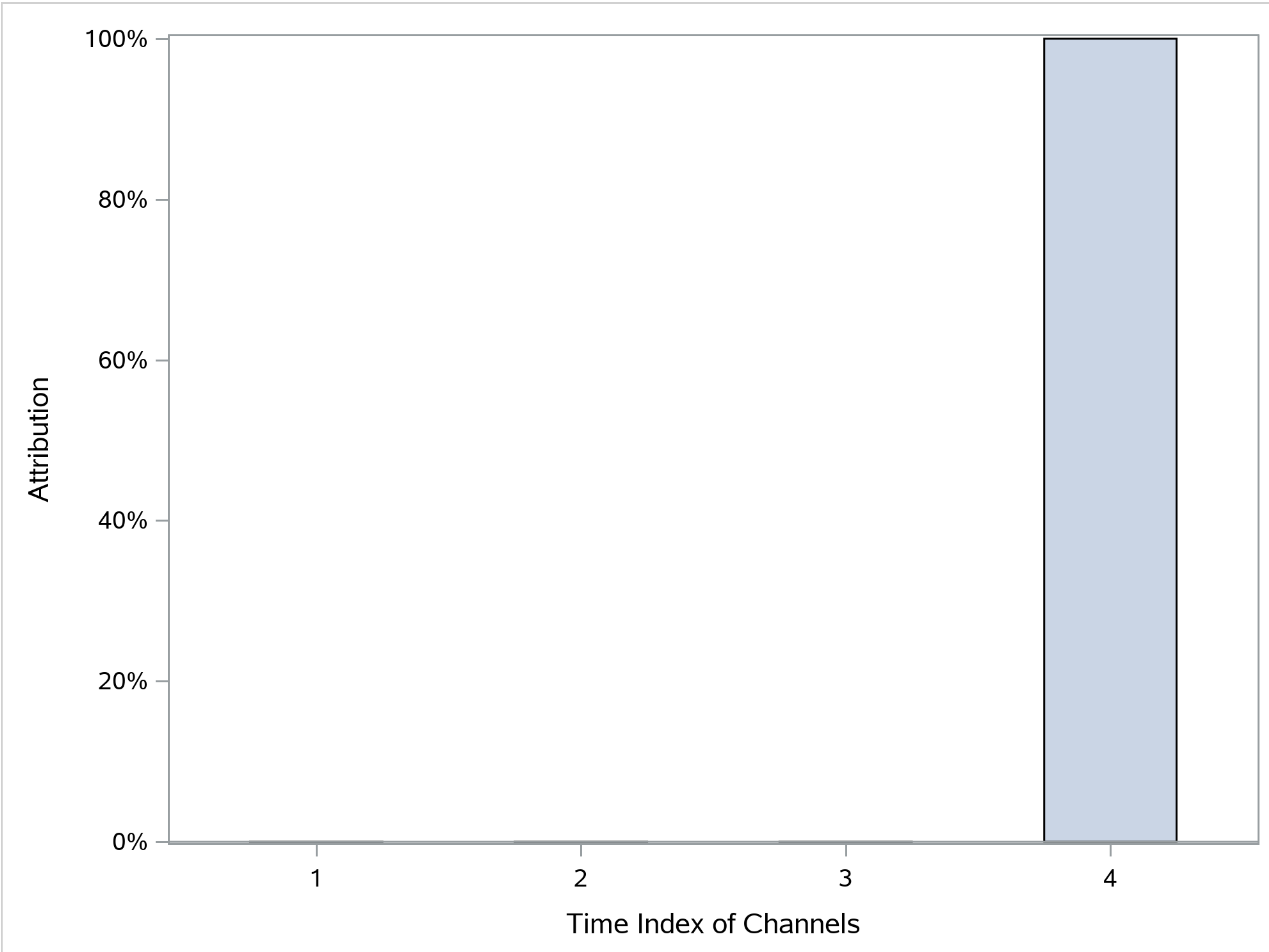


The contribution to channel i by the last-touch attribution model is calculated similarly by

$$contribution(i) = \frac{\sum_{j=1}^{n} f(C_j(l_j) = i|conversion)}{m}$$

The bar chart in Output 21.9 shows how the last-touch attribution model distributes the contributions for a journey that visits four channels.

Figure 21.9 Bar Chart for the Last-Touch Attribution Model

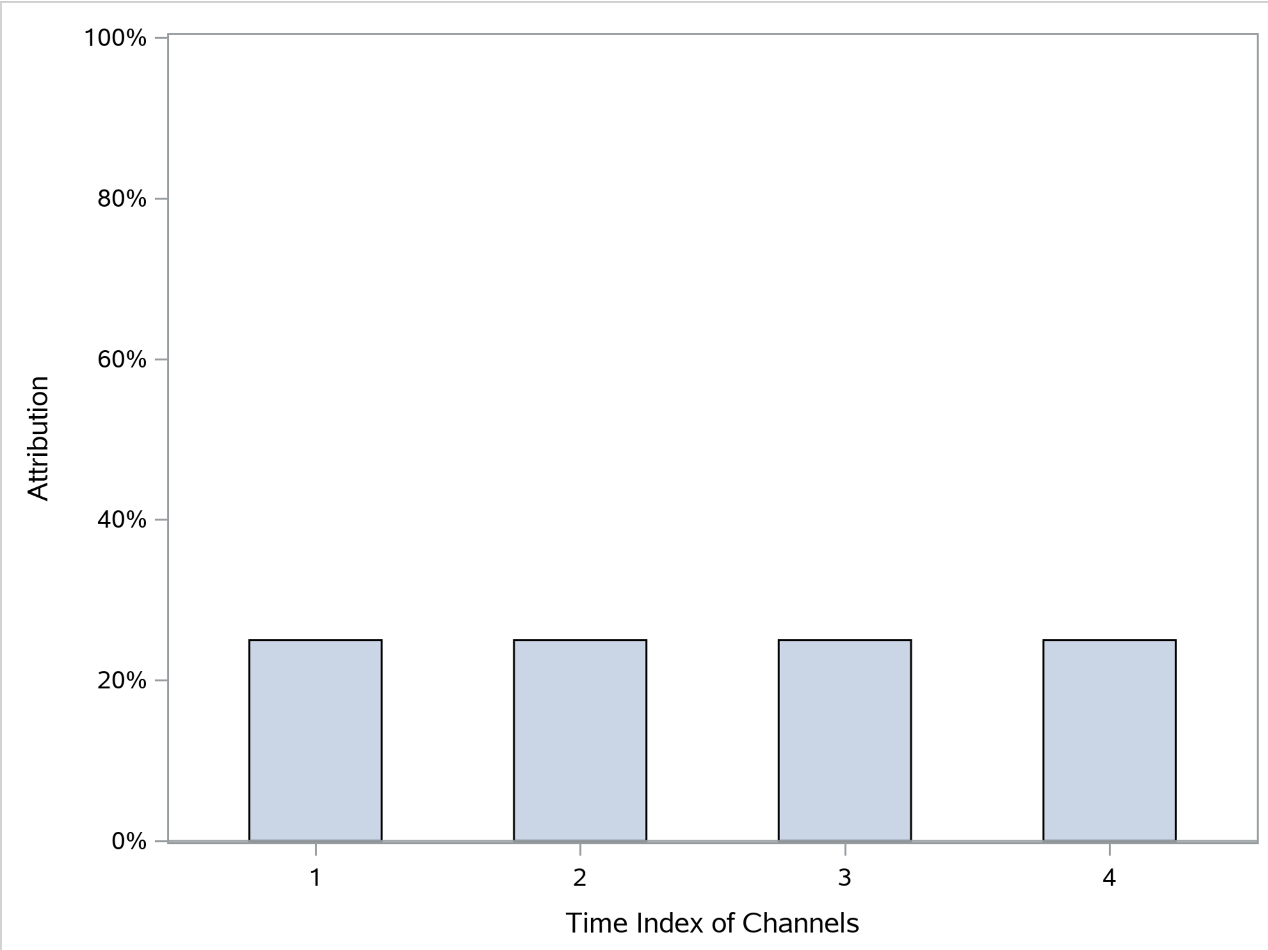


The contribution to channel i by the linear attribution model is calculated by

$$contribution(i) = \frac{\sum_{j=1}^{n} \sum_{p=1}^{l_j} f(C_j(p) = i|conversion)/l_j}{m}$$

The bar chart in Output 21.10 shows how the linear attribution model distributes the contributions for a journey that visits four channels.

Figure 21.10 Bar Chart for the Linear Attribution Model



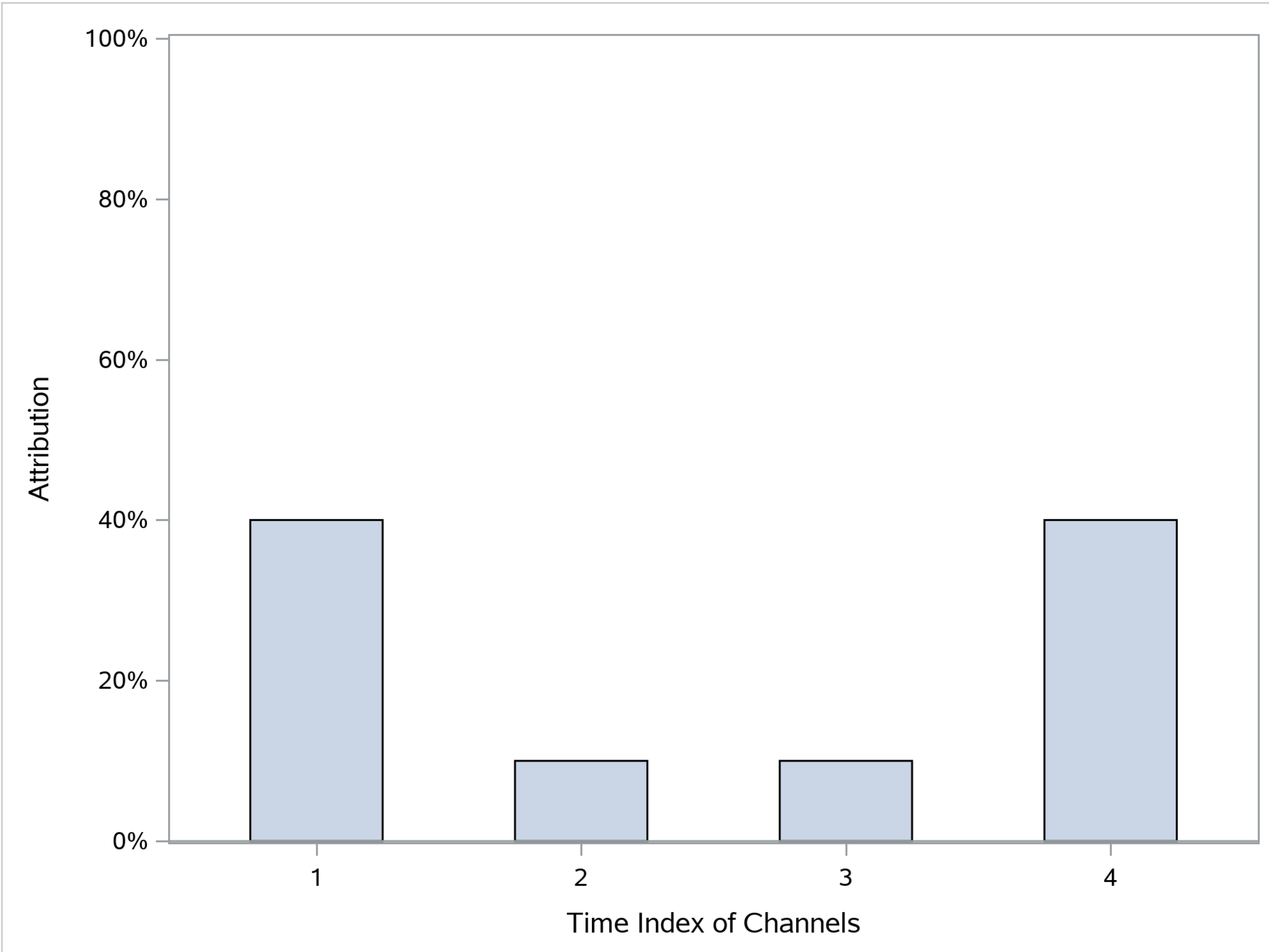
If $l_j <= 2$, the position-based attribution model is the same as the linear attribution model. If $l_j > 2$, then let $h(C_j(p) = i|conversion) = 0.4$ and $e(C_j(p) = i|conversion) = 0.2(l_j - 2)^{-1}$ if $C_j(p) = i$ and if conversion occurs at the end of this journey; otherwise it equals 0. The contribution to channel i by the position-based attribution model is calculated by

$$contribution(i) = \frac{1}{m}\sum_{j=1}^{n}[\mathbf{A}(l_j <= 2) + \mathbf{B}(l_j > 2)]$$

where $\mathbf{A}(l_j <= 2) = \sum_{p=1}^{l_j} f(C_j(p) = i|conversion)/l_j$, if $l_j <= 2$, and otherwise it equals 0; and $\mathbf{B}(l_j > 2) = h(C_j(1) = i|conversion) + h(C_j(l_p) = i|conversion) + \sum_{p=2}^{l_j-1} e(C_j(p) = i|conversion)$, if $l_j > 2$, and otherwise it equals 0.

The bar chart in Output 21.11 shows how the position-based attribution model distributes the contributions for a journey that visits four channels.

Figure 21.11 Bar Chart for the Position-Based Attribution Model



Let f be the half-life parameter, $g(C_j(p) = i|conversion) = (\frac{1}{2})^{(l_j - p)/f}$ if $C_j(p) = i$ and if conversion occurs at the end of this journey; otherwise it equals 0. The contribution to channel i by the time-decay attribution model is calculated by

$$contribution(i) = \frac{1}{m}\sum_{j=1}^{n}[\sum_{p=1}^{l_j} g(C_j(p) = i|conversion)/D]$$

where $\mathbf{D} = \sum_{q=1}^{l_j}(\frac{1}{2})^{(l_j - q)/f}$ is the normalizing coefficient for journey j.

The bar chart in Output 21.12 shows how the time-decay attribution model distributes the contributions for a journey that visits four channels.

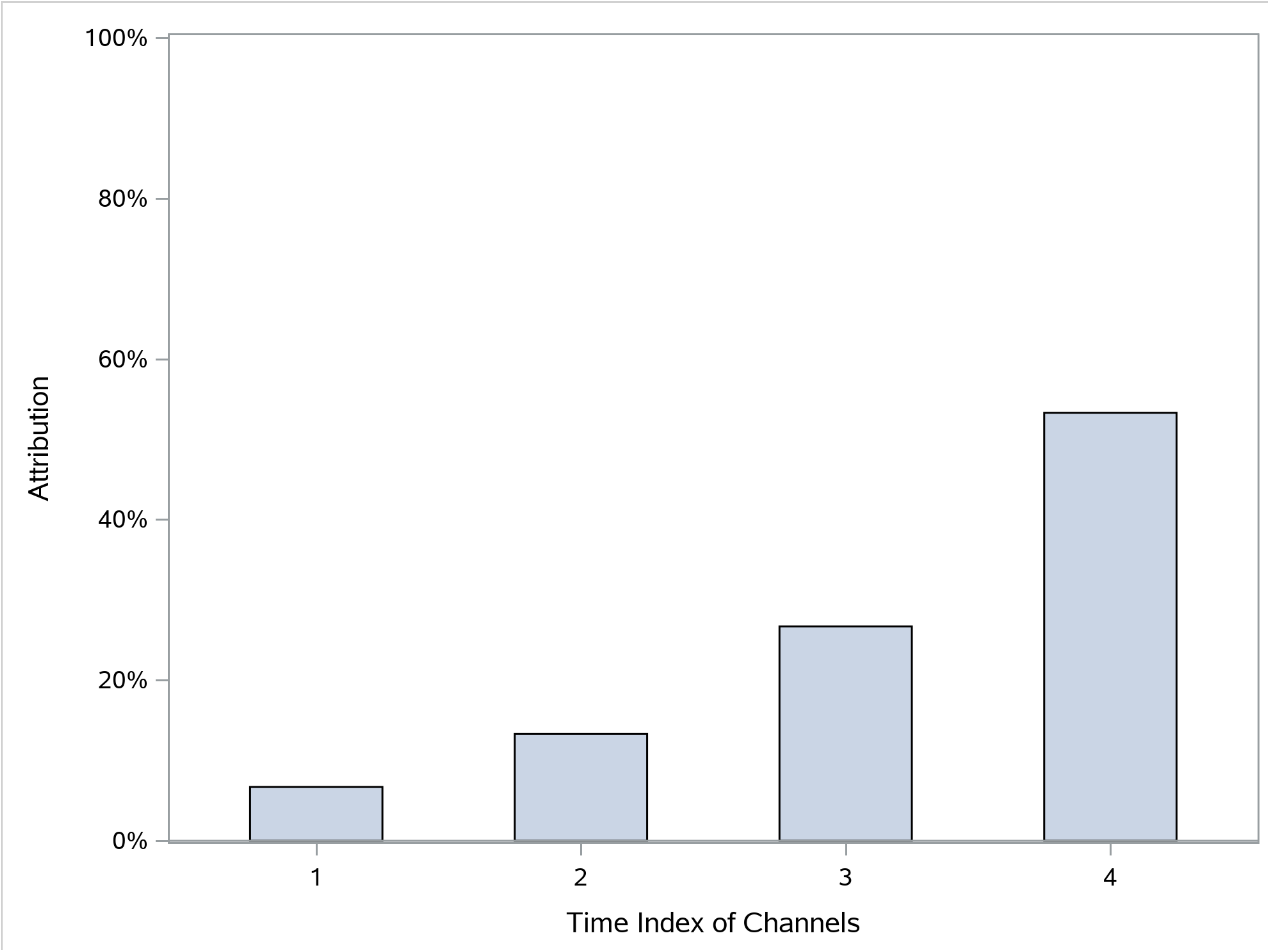


Figure 21.12 Bar Chart for Time-Decay Attribution Model (HALFLIFE=1)

Input Data

The input data set must have three variables: section ID, time ID, and channel. The channel variable must be a nonnegative integer. The section ID variable identifies each customer. The time ID variable sorts the order of visits of each customer. The combination of section ID and time ID must be unique. The channels can appear in raw data as strings, such as “email”, “online search”, and so on. You need to convert them to the natural numbers and zero, where zero represents the conversion, and save them in the channel variable. So the largest value of the channel variable is the number of distinctive channels in the input data. Any customer journey that does not end with zero ends with a null state. Each customer can have at most only one conversion. The procedure skips records that have a missing channel variable. The procedure skips the records in which the conversion occurs at the beginning or middle of the journey. The procedure skips the records in which consecutive channels occur within the same customer journey, so the estimated TPM always has zero values in the diagonal positions.

Data Table Output

The MKTATTRIBUTION procedure can create the data tables that are specified in the following options in the OUTPUT statement: OUTCONTRIBUTION=, OUTREMOVAL=, and OUTTPM=. The column information that each table contains is described in the following sections.

OUTCONTRIBUTION= Data Table Generated from the OUTPUT Statement

The output data table contains the following variables (columns):

Channel	channel for which to calculate the removal effect
Markov Chain	contribution calculated using the Markov attribution model
First-Touch	contribution calculated using the first-touch model
Last-Touch	contribution calculated using the last-touch model
Linear	contribution calculated using the linear attribution model
Position-Based	contribution calculated using the position-based attribution model
Time-Decay	contribution calculated using the time-decay attribution model

OUTREMOVAL= Data Table Generated from the OUTPUT Statement

The output data table contains the following variables (columns):

Channel	channel for which to calculate the removal effect
Value	removal effect for the corresponding channel

OUTTPM= Data Table Generated from the OUTPUT Statement

The output data table contains the following variables (columns):

State	previous state in the Markov chain
Channel*k*	probability of transition to channel k, $k = 1, \ldots, K$, where K is the number of channels
Conversion	probability of transition to the conversion state
Null	probability of transition to the null state

ODS Table Names

The MKTATTRIBUTION procedure assigns a name to each table that it creates. You can use this name to refer to the table when using the Output Delivery System (ODS) to select tables and create output data tables. These names are listed in Table 21.2.

Table 21.2 ODS Tables Produced in the MKTATTRIBUTION Procedure

ODS Table Name	Description	Option
Contributions	Contribution value for each channel calculated for each model	Default
ConversionRates	Conversion rate calculated using input data and from the estimated TPM	Default
ISPV	Initial state probability vector	Default
ModelInfo	Model information	Default
NObs	Observation information	Default
RemovalEffects	Removal effect for each channel	Default
TPM	Transition probability matrix	Default

Examples: MKTATTRIBUTION Procedure

This section provides an example of the model that the MKTATTRIBUTION procedure supports.

The following statements read a data table that contains records of 2 million customer visits. The table has four columns: sec is the number to identify the customers, time is the time when the customer visits, c is a nonnegative integer variable that is converted from the channel variable and is to be used in the model, and channel is the channel that the customer visits at time t.

```
data marketData;
   format time DATETIME.;
   input sec  time:ANYDTDTM40.  c channel & $ 15.  ;
datalines ;
1   24MAY18:00:00:00    1   Direct Mail
1   05JUN18:23:25:55    4   email
1   05JUN18:23:26:36    0   conversion
2   13JUN19:18:05:25    11  social
2   26OCT19:03:44:08    4   email
3   22AUG19:13:25:49    7   paid social
4   28JUL19:00:52:21    7   paid social

   ... more lines ...

1169566 15JUN18:11:18:33    5   organic
1169567 05MAY18:00:00:00    1   Direct Mail
1169567 03APR20:05:40:37    5   organic
```

```
1169567 03APR20:12:02:20    0   conversion
1169568 07SEP19:21:50:07    7   paid social
1169569 04MAY19:15:44:22    5   organic
1169570 14AUG18:15:14:56    11  social
1169570 21OCT19:00:00:00    1   Direct Mail
1169571 15MAY19:19:32:36    7   paid social
;
```

The following statements create the table (Output 21.13) to show the one-to-one relationship of channel to c:

```
data channel;
  set marketData(keep=c channel);
run;
proc sort data=channel
  dupout=channel_NoDupkey nodupkey;
  by c;
run;
proc print data=channel noobs; var c channel; run;
```

Figure 21.13 Market Channels

c	channel
0	conversion
1	Direct Mail
2	app
3	display
4	email
5	organic
6	paid search
7	paid social
8	phone
9	print
10	sms
11	social
12	video
13	web

In addition to the conversion, which equals 0, in the c variable, there are 13 distinctive channels shown in Output 21.13.

The following statements estimate the Markov attribution model that has 13 channels:

```
proc sort data=marketData out=marketData nodupkey;
   by sec time;
run;
data mylib.marketData2;
   set marketData;
run;
proc mktattribution data=mylib.marketData2;
   id section=sec time=time;
   model c / nchannel=13;
   output outremoval=mylib.rm outcontribution=mylib.contribution outtpm=mylib.tpm ;
run;
```

The model information and number of observations are shown in Output 21.14.

Figure 21.14 Model Information and Number of Observations

The MKTATTRIBUTION Procedure

Model Information	
Model Type	Markov Attribution
Number of Channels	13

Number of Observations	2045016
Number of Missing Observations	0

The estimates of the initial state probability vector are shown in Output 21.15.

Figure 21.15 Estimates of Initial State Probability Vector

Initial State Probability Vector	
State	**Estimate**
Channel 1	0.20354
Channel 2	0.00003
Channel 3	0.00706
Channel 4	0.10067
Channel 5	0.20121
Channel 6	0.04194
Channel 7	0.34759
Channel 8	0.00027
Channel 9	0.00159
Channel 10	0.00141
Channel 11	0.07281
Channel 12	0.00330
Channel 13	0.01859
Conversion	0.00000
Null	0.00000

The estimated transition probability matrix are shown in Output 21.16.

Figure 21.16 Estimated Transition Probability Matrix

Estimated Transition Probability Matrix												
State	Channel 1	Channel 2	Channel 3	Channel 4	Channel 5	Channel 6	Channel 7	Channel 8	Channel 9	Channel 10	Channel 11	Channel 12
Channel 1	0.00000	0.00002	0.00896	0.34996	0.23078	0.07311	0.02876	0.00083	0.01499	0.00441	0.00760	0.00112
Channel 2	0.08046	0.00000	0.00000	0.12644	0.45977	0.01149	0.00000	0.00000	0.00000	0.02299	0.00000	0.00000
Channel 3	0.22110	0.00005	0.00000	0.15465	0.12813	0.02383	0.01592	0.00000	0.00070	0.00154	0.00423	0.00035
Channel 4	0.37618	0.00004	0.00918	0.00000	0.20614	0.01010	0.00176	0.00001	0.00038	0.00321	0.00217	0.00016
Channel 5	0.20906	0.00006	0.00776	0.19783	0.00000	0.02327	0.00432	0.00079	0.00284	0.00252	0.00988	0.00053
Channel 6	0.06761	0.00004	0.00680	0.03876	0.20502	0.00000	0.00380	0.00014	0.00236	0.00043	0.00192	0.00068
Channel 7	0.04391	0.00000	0.00372	0.01489	0.02980	0.00188	0.00000	0.00000	0.00002	0.00041	0.01513	0.00004
Channel 8	0.03718	0.00000	0.00000	0.00489	0.29746	0.00489	0.00000	0.00000	0.00000	0.00000	0.00000	0.01272
Channel 9	0.03907	0.00000	0.00108	0.01286	0.17187	0.02150	0.00049	0.00000	0.00000	0.00020	0.00118	0.00010
Channel 10	0.27363	0.00017	0.00670	0.15231	0.23827	0.01039	0.00318	0.00000	0.00034	0.00000	0.00134	0.00017
Channel 11	0.08723	0.00000	0.00374	0.02587	0.06770	0.00275	0.02786	0.00001	0.00015	0.00037	0.00000	0.00002
Channel 12	0.03582	0.00000	0.00361	0.01947	0.12188	0.01803	0.00264	0.00505	0.00096	0.00048	0.00048	0.00000
Channel 13	0.05314	0.00012	0.00258	0.07075	0.26652	0.01966	0.00255	0.00006	0.00317	0.00177	0.00494	0.00112
Conversion	0.00000	0.00000	0.00000	0.00000	0.00000	0.00000	0.00000	0.00000	0.00000	0.00000	0.00000	0.00000
Null	0.00000	0.00000	0.00000	0.00000	0.00000	0.00000	0.00000	0.00000	0.00000	0.00000	0.00000	0.00000

Estimated Transition Probability Matrix			
State	Channel 13	Conversion	Null
Channel 1	0.01082	0.00920	0.25946
Channel 2	0.01149	0.12644	0.16092
Channel 3	0.00473	0.04432	0.40047
Channel 4	0.00488	0.06617	0.31964
Channel 5	0.01339	0.18544	0.34231
Channel 6	0.01079	0.53286	0.12880
Channel 7	0.00043	0.01731	0.87246
Channel 8	0.00391	0.63796	0.00098
Channel 9	0.00982	0.72006	0.02179
Channel 10	0.00419	0.08043	0.22889
Channel 11	0.00180	0.02622	0.75627
Channel 12	0.00505	0.71875	0.06779
Channel 13	0.00000	0.20251	0.37110
Conversion	0.00000	1.00000	0.00000
Null	0.00000	0.00000	1.00000

The conversion rates are shown in Output 21.17.

Figure 21.17 Conversion Rates

Conversion Rates	
Method	Rate
Data Based	0.18771
Markov Chain	0.18771

The removal effects are shown in Output 21.18.

Figure 21.18 Removal Effects

Removal Effects	
Channel	**Effect**
1	0.07980
2	0.00003
3	0.00379
4	0.06379
5	0.11014
6	0.05682
7	0.01567
8	0.00079
9	0.00833
10	0.00157
11	0.00668
12	0.00335
13	0.01071

The removal effect of a channel is the decrease in the probability of overall conversion if the channel were removed. In Output 21.18, you see that channel 5 (organic) leads to the greatest decrease in the probability of conversion if it is missing.

The contributions are just normalized removal effects to indicate each channel's contribution to the conversion. The contributions of the Markov attribution model and the other three heuristic attribution models are shown side by side in Output 21.19. The last-touch model does not place as much value on channel 1 (direct mail) as other methods do, but it attributes the most value to channel 5 (organic). The first-touch model does not value channel 4 (email) as much as other methods do, but it attributes the most value to channel 1 (direct mail). The Markov and linear attribution models have similar attribution values for most of the channels, except that there is a 9% difference between the two models on the contribution of channel 1 (direct mail). The position-based model and time-decay model both attribute less credit to channel 4 (email), which is the third most important channel according to the Markov model. The reason might be that the email channel always appears in the middle of the customer journey.

Figure 21.19 Channel Contributions

Channel Contributions						
Channel	**Markov Chain**	**First Touch**	**Last Touch**	**Linear**	**Position Based**	**Time Decay**
1	0.22078	0.47248	0.02320	0.23493	0.24179	0.23236
2	0.00008	0.00003	0.00006	0.00005	0.00005	0.00005
3	0.01049	0.00238	0.00486	0.00442	0.00393	0.00408
4	0.17649	0.03489	0.13081	0.08445	0.08369	0.08707
5	0.30470	0.24707	0.43312	0.34435	0.34127	0.34541
6	0.15719	0.15831	0.26382	0.21558	0.21376	0.21484
7	0.04336	0.02774	0.03385	0.03106	0.03094	0.03090
8	0.00217	0.00136	0.00356	0.00255	0.00251	0.00258
9	0.02304	0.00714	0.04001	0.02437	0.02405	0.02434
10	0.00434	0.00092	0.00262	0.00176	0.00177	0.00188
11	0.01847	0.01194	0.01223	0.01237	0.01222	0.01219
12	0.00926	0.01408	0.01631	0.01515	0.01517	0.01521
13	0.02964	0.02166	0.03556	0.02897	0.02884	0.02908