UNIVERSIDAD COMPLUTENSE DE MADRID
FACULTAD DE CIENCIAS MATEMÁTICAS

UNIVERSIDAD POLITÉCNICA DE MADRID
ESCUELA TÉCNICA SUPERIOR DE INGENIEROS DE TELECOMUNICACIÓN

MÁSTER EN TRATAMIENTO ESTADÍSTICO COMPUTACIONAL DE LA INFORMACIÓN

TRABAJO DE FIN DE MÁSTER

Attribution Models in Real-Time bidding

Alejandro Morales Fernández

Directores

Adrián Matías Diez (Digilant, S.L.)

Juan Antonio Tejada Cazorla (UCM)

Madrid, 2016

Resumen

Uno de los grandes problemas que plantea la publicidad en tiempo real es proporcionar una atribución justa a cada canal, cuando impactan conjuntamente. Desde la aparición de la publicidad digital, el modelo hegemónico en este ámbito ha sido el modelo de última interacción. Sin embargo, en los últimos años ha crecido el interés en las empresas por explorar otros modelos de atribución que utilicen más información.

Aunque es difícil resumir de forma concisa la labor realizada en el presente trabajo, este engloba inicialmente dos objetivos principales: Uno de ellos es hacer una introducción a los modelos de atribución en la compra programática en marketing digital y obtener modelos en dicho mercado más allá del modelo de última interacción. Por otro lado, los modelos de atribución han sido desarrollados mediante la teoría de juegos. El otro objetivo es mostrar y desarrollar cómo se ha utilizado la teoría de juegos en este entorno.

La tesis está dividida en dos partes: la primera parte hace una puesta al día de dos conceptos: en el primer capítulo a la compra programática y en el segundo capítulo se hace una introducción a los juegos cooperativos. En la segunda parte se explica cómo aplicar la teoría de juegos al problema concreto y su implementación en la práctica. Se muestran diversos resultados de salida e información relevante. Por último, se incluye un capítulo de conclusiones.

Abstract

Among the most important problems that online publicity poses is providing a fair attribution to each impact channel. Since the appearance of digital publicity, the last interaction model has been the hegemonic model in this environment. However, in the last few years the interest of companies in exploring other attribution models which use more information has risen.

Despite the fact it is difficult to summarize this thesis, it includes initially two main objectives: one of them is performing an introduction to attribution models within programmatic buying in digital marketing and obtaining models beyond the last interaction one. Secondly, attribution models have been developed by means of game theory. The other goal is therefore showing and developing how game theory has been used inside this environment.

The dissertation is divided into two parts: the first refreshes two concepts: programmatic buying in the first chapter and cooperative games in the second chapter. In the second part application of game theory and implementation in practice is covered. Some output results and further relevant information is shown. Finally, the last chapter contains some achieved conclusions.

Contents

Part I

Introduction to Attribution Problems in Real-time Bidding

Chapter 1

Introduction

Online marketing was born in 1993 and has grown at the same time as the evolution of new technologies. Nowadays, technology allows us to organize this market fulfilling the goals negotiated by both buyers and sellers, in a response time measured in milliseconds. The explosion of online users caused the increase in the number of websites. Figure (1.1) shows the how the amount of internet users raised between 1990 and 2014. That is the reason why the ecosystem needed an easier mediation between demand and supply.

The appearance in 2008 of programmatic trading, the technology behind the RTB, which allows us to buy and sell digital publicity in real time, implies the beginning of a new generation of players in this ecosystem. It is the moment when trade was carried out mechanically according to different criteria defined by buyers and sellers, giving rise to Real-Time Bidding or programmatic buying.

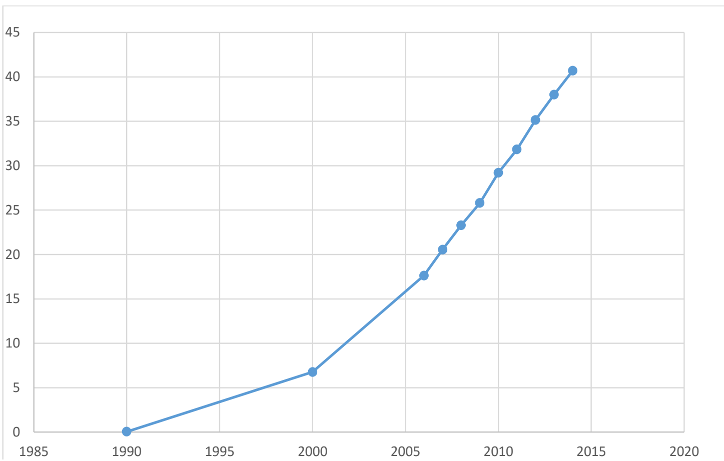

Figure 1.1: Internet users rise. *Retrieved from* `http://data.worldbank.org/`

Real-Time Bidding (RTB) ([1]) is a method for trading online display advertising placements through real-time auctions that occur in the time it takes a website to download. The auction process takes place across media exchanges that connect sellers (publishers) and buyers (advertisers).

The trade does not differ from any other market. From the demand point of view, the advertiser is the entity charged with executing an advertising campaign. This includes finding the right audience and showing the suitable ad. On the other hand, the publisher or inventory is the place where ads will be shown and will be the supplier in this market. These publishers offer appropriate spaces with the objective of obtaining the biggest earnings.

1.1 RTB Marketplace

Before we get into some of the inner essentials of this ecosystem, let us have a short and intuitive clarification of the terminology of the players in the display advertising ecosystem. We will divide the classification in two groups according to supply and demand.

- **Supply**

  We can find one important player when it comes to supply. The *inventory* or publisher, which is the place where ads will be shown. In particular, it could be any website which has room available for ads. To allow advertising networks to optimize the inventory performance automatically, there are advertising technology platforms called *Supply Side Platform (SSP).*

- **Demand**

  On the Demand side, we can find two important players, *marketers* and *advertisers*. The marketer or brand is the one who is interested in advertising. They determine the total available budget for the campaign. It could be for instance *Coca Cola* that wants to promote a new product. The advertiser is the entity charged with executing the campaign. This includes finding the right audience and showing them the ad. The advertiser uses a technology called *Demand Site Platform (DSP)*. Sometimes the advertiser itself is called informally DSP. These, also know as bidders, “bid” for the inventory using data from the audience, so that the purchase is done impression by impression. An impression is an ad that is reproduced in the inventory. DSPs provide technology and knowledge for:

  - Valuing every available impression individually and buying the optimal ones for the campaign. In this process diverse methods and algorithms for optimization are used.
  - Integrating data, adding the entire information available (own, from data providers or even from the marketer) to improve the decision criteria.

  An example of DSP company is Digilant S.L., where this dissertation has been done.

- **Other important players**

  There are other players or elements who participate in this market, but they are not labeled neither in demand or supply. The most important one is the *Ad-Exchange*, an online platform that facilitates automation of real-time bidding. Therefore, it is the place where demand and supply are reunited.

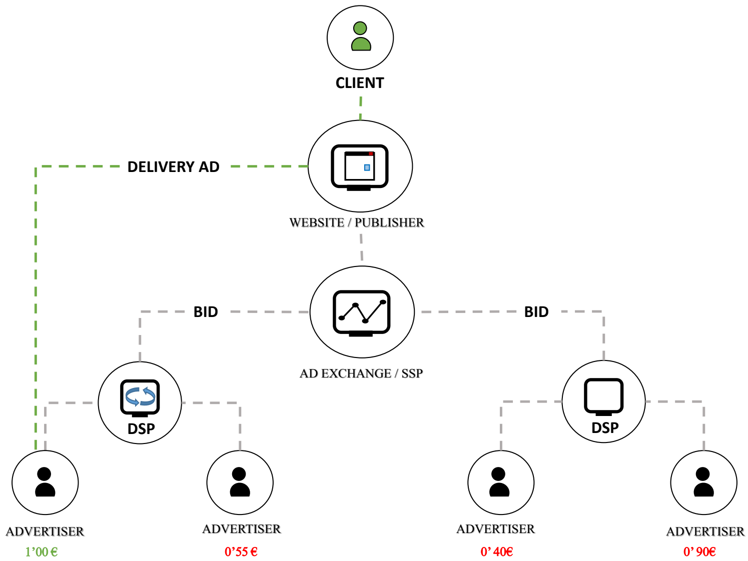

Figure 1.2: Simplified diagram of the real time bidding ecosystem

1.2 What is a cookie and its life cycle

A *cookie* ([2, 10]) is conceptually a person who is going to receive the ad. From the advertiser's perspective:

1. A user visits a website, where the DSP has a pixel, and a cookie is written in his web browser to recognize him again.
2. DSP anonymously track the browser's Internet behavior.
3. The SSP computes a floor price for this inventory.
4. A bid request is made and sent to Ad exchange.
5. Every particular DSP receives the bid request and computes a bid:
   (a) Previously, the browsing history has been used to score the user. If he has a score high enough, he is placed into a discrete *segment*. It is useful to discretely limit the set of eligible cookies, for which the RTB forwards big requests. Each campaign

has multiple segments for different thresholds and different models. Typically only about 1% of the browsers qualify for any segment of a given campaign.

(b) At the time of the request, DSPs already know the user's cookie, the current inventory, as well as additional generic information about the user agent, etc. Thus, they know which segment the user is in, and decide the optimal bid according to their information.

6. The DSP with the highest bid wins the auction and gets to show the creative for the campaign and pay the price for the second highest bid. The ad is displayed and this event is called **impression**.

7. The cookie is observed for some campaign-specific conversion period, recording if the user converts.

8. If the user does not convert, the entire process is repeated, and is re-scored the more data is collected.

9. At some point, the user may delete its cookie and is no longer known. If it is observed again, he will receive a new cookie.

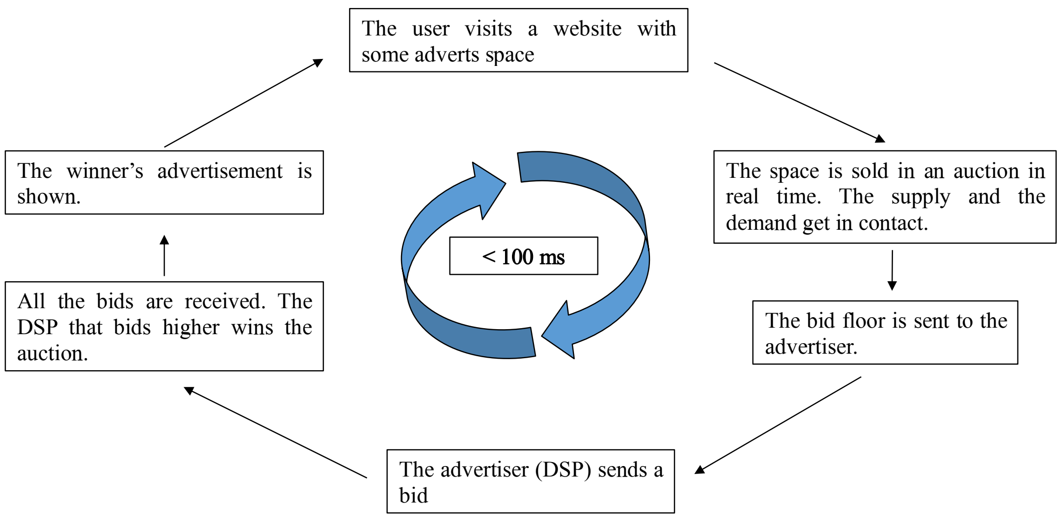

Figure 1.3: Cookie Life Cycle.

1.3 The problem of evaluating different channels

In the real-time bidding environment, a user or cookie is impacted through diverse channels. These channels may be physical channels as computers, tablets, smartphones, etc. However, the channels will be defined as something more abstract. Previously, let us define the conversion concept. A **conversion** is an event that indicates that the selected user has bought the desired product or simply a user that has done a beneficial purchase. Knowing formally what a conversion is, a **channel** is defined as any feature from the data available that has influence on the conversions.

Example 1.3.1. Suppose we have the following 6 rows extracted from a database sorted by time. Each row contains information of impressions.

User ID	Device	Day Time	Ad's size
User1	PC	Morning	Small
User2	PC	Morning	Large
User1	Tablet	Afternoon	Medium
User3	Smartphone	Afternoon	Small
User3	PC	Evening	Small
User2	Smartphone	Evening	Small

It means that at some point in the morning, user number 1 has been impacted with a small add in his pc, later has been impressed in the afternoon with a medium size add in his tablet and so forth.

Typically, the previous information will be together with the information of which users have been converted. For instance, it can be stored in the following table:

User ID	Conversion
User1	0
User2	1
User3	1

The previous table means that users 2 and 3 have converted, whereas user 1 has not. As first consequence we deduce that channel "smartphone" could be a very good channel because every user who has been impacted through it has converted. In this manner, channel "Small" should be also good, because all the smartphones impacts are done through small ads.

Sometimes the conversion information is attached in the data table and "conversion" it is modelled as a type of event. There can be also clicks. A click occurs if a user literally clicks on the displayed ad. The data usually mixes different types of events as shows the following table:

User ID	Device	Day Time	Ad's size	Event
User1	PC	Morning	Small	Impression
User2	PC	Morning	Large	Impression
User1	Tablet	Afternoon	Medium	Click
User3	Smartphone	Afternoon	Small	Impression
User3	PC	Evening	Small	Conversion
User2	Smartphone	Evening	Small	Conversion

From the previous example, we could think of a channel just as an attribute from a categorical variable. Moreover, it will be also useful to understand it as a player in a cooperative game as will be seen later.

The problem of rating channels according to their performance of conversions is a difficult problem and in principle there could be more than one approach to solve this problem. For instance, classification problem techniques and other mathematical tools can be utilized for this purpose. In this work, we will use cooperative game theory and determine a fair payment to each channel.

1.4 Attribution theory

In the core of the problem of evaluating different channels based on conversions is the *attribution theory*. This theory has become a major research paradigm on social psychology and is concerned with how individuals interpret events and how this relates to their thinking and behaviour. Heider (1958) was the first to propose a psychological theory of attribution, but Weiner (1974) and other scientists developed a deeper theoretical framework.

A few of the possible things we can study by means of the attribution theory are:

- How do we attach meaning to other's behaviour, or our own?
- Is someone angry because is bad-tempered or because something bad happened?
- In the real-time bidding environment, has a conversion occurred because of the last impression we served or due to a long path of impressions which has influenced the user's behaviour?

The models which are built in this monograph are developed using mathematical tools. However, it is important to take into consideration that many of studies of behaviour attribution have been done from a psychological point of view.

1.5 Attribution Models in real-time bidding

An attribution model is a rule or a set of rules, which determine how conversions are assigned to some impressions or ads. As has been said above, there are several ways to approach the problem. In this section several important concepts are defined that will be used throughout the thesis.

In order to have better knowledge about the users that are being impacted in any publicity campaign, we can store the information about all the impressions that the user has received. This leads to the following definition of path:

Definition 1.5.1. A path of a given cookie is a vector containing all the events in the history of a given campaign for that cookie.

In example 1.3.1, the device type smartphone has been given an important attribution because the last device through an impression has been posted is smartphone. Thus, only a piece of information has been used to assign attribution. The aim is to use more information of data to make our conclusions safer.

1.5.1 Attribution models classified by time

There are various types of models according on how the issue of time is addressed. The time is relative to the conversion event. It can be depicted ordering the events towards conversion or from conversion. In figure 1.4 an example can be seen with events from conversion. However, we will represent it the other way for the rest of this work.

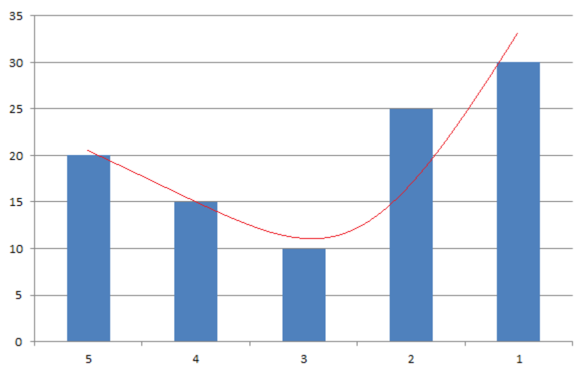

Figure 1.4: Attribution model that gives more attribution to the first events.

Last interaction attribution model

If we think about the way of reasoning in example 1.3.1, we assigned more attribution to the events that impacted right before the conversions than the other events. This is done because intuitively we think the last ads have had more influence on the conversions.

The model that includes this idea is the *last interaction attribution model*. Assigns 100% of the conversion value to the last channel with which the user interacted before converting. In figure 1.5 a last interaction model can be seen. The x-axes represents 5 different touchpoints in this case, ordered by time, being the fifth one the closest to a conversion event. The y-axes represents the amount of attribution given to every touchpoint, in percentage.

This model is the most extended one and the easiest to implement compared to the others. However, it could ignore an important amount of information because sometimes paths exceed a length of 1000 impressions.

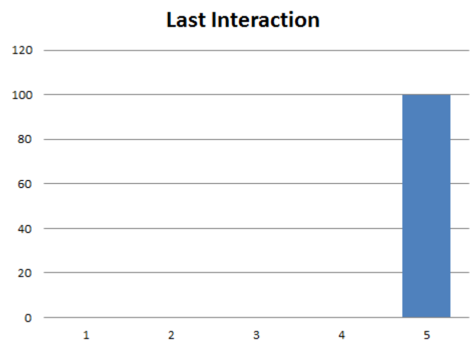

Figure 1.5: Last Interaction Model. All the attribution goes to the last touchpoint.

First interaction attribution model

Assigns 100% of the conversion value to the first channel. It is useful in channels where the awareness effect takes the best importance. However, it is not widely seen.

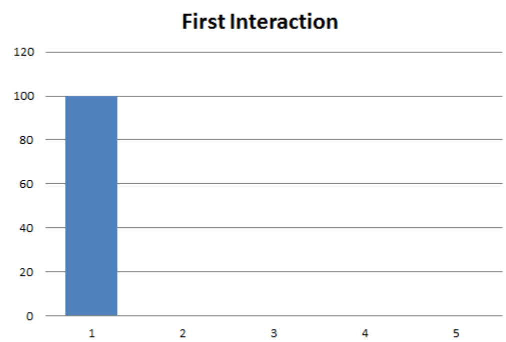

Figure 1.6: Last Interaction Model. All the attribution goes to the first touchpoint.

Presence attribution model

Also called flat attribution model or “One Each” Each touchpoint is equally important during the attribution process.

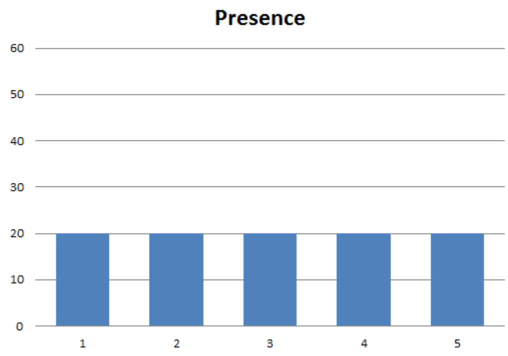

Figure 1.7: Presence Interaction Model. The attribution is shared equally by all the touchpoints.

Time decay attribution model

This model is based on the concept of exponential decay and most heavily credits the touchpoints that occurred nearest to the time of conversion. It has the shape of an increasing function. At first glance, the name may be misleading, because the function is increasing, not decreasing as the name suggests. The reason is the following: if we represent the histogram according to the distance to conversion as in 1.4, it has the shape of a decreasing function. That is the reason of its name.

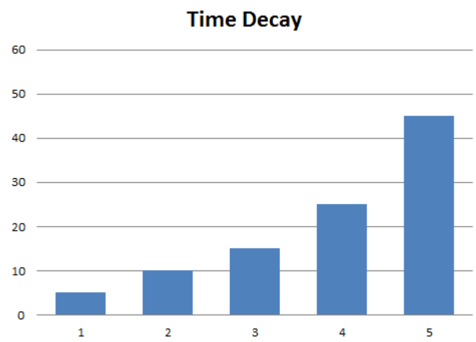

Figure 1.8: Time decay Model.

Customized attribution model

Finally, it is common to design customized models according to results or experience.

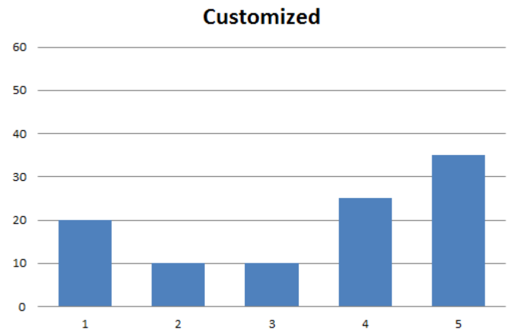

Figure 1.9: Customized model. The analyst chooses the attribution of each touchpoint.

1.6 Types of Strategies in Real-time bidding

Another important issue to face is choosing what strategies to take in the RTB environment. Note that the concept of whether a strategy is better than another can be subjective and change depending on what the goals of the campaign are. If it is preferable to achieve as many conversions as possible, then we will opt for an aggressive strategy, bidding at a high price in the auctions. On the contrary, if we have a small budget, we will opt for cheaper offers and will bid more intelligently.

To summarize, one way of possible classification of tactics or strategies in real-time bidding is *prospecting* and *retargeting*. The first one seeks to create awareness of a single product, whereas the aim of retargeting is to impact a limited number of users who have previous engagement and therefore a high probability of conversion. Note that they are exclusive, this means we have to use both strategies, first to create product awareness and second to make conversions.

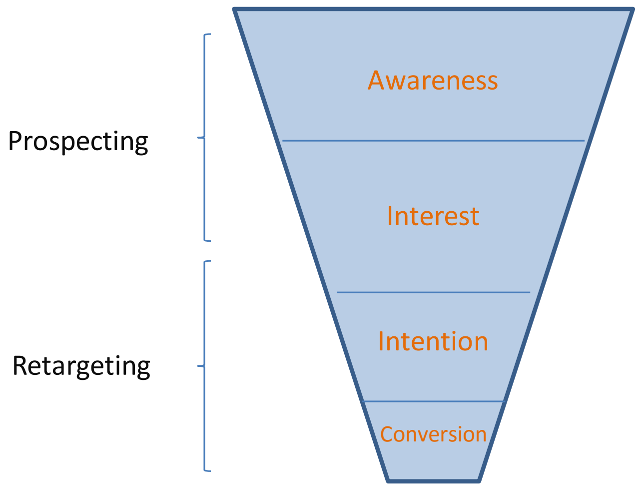

Figure 1.10: Main phases in a cookie life towards conversion.

It is important to highlight that most of the cookies will remain in the beginning of the funnel, not being impacted through any retargeting strategy.

In the cookie available data, a feature called strategy will exists in most of the cases. And possibly another feature called *Line Item* (substrategy) may exists. The purpose of this last one is to set numerical specifications to the strategies, or focus them to a reduced segment of cookies. Finally, we can rate all these strategies or line Items with the attribution model, and check their performance.

Chapter 2

Review of cooperative games. The Shapley Value

2.1 Cooperative Games

One of the objectives of this work is to give a fair attribution to each channel in a way which will be seen later. One of the possible approaches to do so is by means of cooperative game theory, and the method for doing this will be seen in chapter 3. For now we will focus the attention on recalling basic theory of cooperative games.

Definition 2.1.1. A cooperative or coalitional game with transferable utilities consists of:

- A finite set of players $N = \{1, \ldots, n\}$
- A characteristic function $v : 2^N \to \mathbb{R}$ that assigns to every subset S of N, a real number $v(S)$ (worth of the coalition), being $v(\emptyset) = 0$.

Therefore, a game G is determined if a pair (N, v) is given. Once a game is given, the main aim should be giving a fair payoff to each player. Formally is called a *solution concept* and is a vector $x \in \mathbb{R}^n$.

Example 2.1.1. Let (N, v) be a game of $N = \{1, 2, 3\}$ players and a total prize of 1 million \$. If at least two players form a coalition, they get the total prize, but any player separately cannot attain any prize. Thus,

$$v(1) = v(2) = v(3) = 0, \quad v(1,2) = v(1,3) = v(2,3) = v(1,2,3) = 1$$

The game is normalized, so that the total prize is set to 1. If a coalition is formed, let us say, $\{1, 2\}$, with equal payoffs for each player in that coalition, $x(1) = x(2) = \frac{1}{2}$, one can argue that every offer from the third player to for example the second one will be accepted if that offer is greater than $\frac{1}{2}$. Thus, every coalition can be answered with another offer and it is not possible to find a stable solution as in non-coalitional games.

That is why a different approach is followed in this type of game and it consists in finding a fair payoff for each player x_i with the condition that the sum over all the payoffs is the total prize: $\sum_{i=1}^{n} x_i = v(N)$. In game theory, the previous condition is called *efficiency*.

Definition 2.1.2. A game $G = (N, v)$ is monotone if $\forall S \subset T \subset N$ with $S \cup T = \emptyset$,

$$v(S) \leq v(T)$$

Definition 2.1.3. A game $G = (N, v)$ is said to be simple if $v(S) \in \{0, 1\}$ $\forall S \subset N$ and is monotone.

Thus, the previous game is simple.

Definition 2.1.4. A game $G = (N, v)$ is superadditive if $\forall S, T \subset N$ such that $S \cap T = \emptyset$,

$$v(S) + v(T) \leq v(S \cup T)$$

It is immediate to prove that every superaditive game which takes positive values is monotone. The most important and most intuitive games are the superadditive ones. The latter follow the idea of "The union makes the force". Non-superadditive games are uncommon.

Definition 2.1.5. A game $G = (N, v)$ is $(0, 1)-$normalized or just normalized if

$$v(\emptyset) = 0, \;\; v(N) = 1,$$

$$0 \leq v(C) \leq 1, \; \forall C \subset N$$

Note that the last condition is dispensable if the game G is superadditive.

2.2 Set solutions for coalitional games

To delimit the set of solutions of the game one can add suitable restrictions according to find a suitable payoff for each player. The most basic one is to impose efficiency, and this set is called the set of preimputations:

$$PI(v) = \{x \in \mathbb{R} / \sum_{i=1}^{n} x_i = v(N)\}$$

However, the previous set is very large and barely delimits the possible set of payoffs. Another plausible restriction is to impose $x_i \geq v(i)$ $\forall i \in N$. The meaning of this is that every possible payoff is at least as large as the quantity the player would get forming a single coalition. Formally, this set is called the set of imputations:

$$I(v) = \{x \in PI(v) / x_i \geq v(i) \; \forall i \in N\}$$

According to the previous idea, one could impose not only $x_i \geq v(i)$ in single coalitions, but $\sum_{i \in S} x_i \geq v(S)\ \forall S \subset N$, i.e., also in coalitions with more than one player. This set is called the core of the game:

$$C(v) = \{x \in PI(v) / \sum_{i \in S} x_i \geq v(S)\ \forall S \subset N\ \}$$

Note that those can be many restrictions and sometimes the core could be empty. For instance, in example 2.1.1, the restrictions are:

$$x_i \geq 0\ \ \forall i \in \{1,2,3\} \quad ,\ x_1 + x_2 + x_3 = 1$$

$$x_1 + x_2 \geq 1,\ \ x_1 + x_3 \geq 1,\ \ x_2 + x_3 \geq 1$$

If we sum the three last restrictions we reach:

$$2x_1 + 2x_2 + 2x_3 \geq 3\ \ \Rightarrow x_1 + x_2 + x_3 \geq 1.5$$

In contradiction to $x_1 + x_2 + x_3 = 1$, therefore the core is empty.

2.3 Punctual solutions. The Shapley value

Now instead of finding set of payoffs, the aim is to find punctual solutions called *values*. Formally, if G^N is the set of games with N players, a value φ is a function such that

$$\varphi : G^N \rightarrow \mathbb{R}^n$$

$$(N, v) \rightarrow \varphi(v)$$

There are different values such as the nucleolus, the $\tau-$value and the Shapley Value, which have been studied by numerous researchers. We are going to focus on the Shapley Value, which is the most famous one and also very intuitive.

The Shapley value is based on the concept of *marginal contribution*. Given a set $S \subset N$ and a player $i \notin S$, the marginal contribution of i in S is given by

$$v(S \cup \{i\}) - v(S) \tag{2.1}$$

The Shapley value is defined in every player $i \in N$ by the following formula

$$\phi_i(v) = \sum_{\substack{S \subset N \\ i \notin S}} \frac{s!(n-s-1)!}{n!}[v(S \cup \{i\}) - v(S)] \tag{2.2}$$

and can be seen as a weighted average over all possible subsets of N. The meaning of the weights $\frac{s!(n-s-1)!}{n!}$ follows next: Imagine that the players meet in a meeting point. Every order of arrivals must have the same probability. Given $i \notin S \subset N$, the probability that the whole coalition S have arrived before i is exactly $\frac{s!(n-s-1)!}{n!}$.

This reasoning leads to another way of calculating the value. First of all, let us recall the definition of a permutation: a permutation is a biyection of any single set S (generally $S = \{1, 2, \ldots, n\}$) with itself. Normally the permutations are given in the two line notation, for example one permutation of 4 elements may be:

$$\pi = \begin{pmatrix} 1 & 2 & 3 & 4 \\ 2 & 4 & 1 & 3 \end{pmatrix}$$

The set of all possible permutations of a given set S is noted by $\pi(S)$. So the idea is not grouping the coalitions as in formula 2.2, but sum over all possible permutations of N directly, thus

$$\phi_i(v) = \frac{1}{N!} \sum_{\pi \in \pi(N)} [v(P_i^\pi \cup \{i\}) - v(P_i^\pi)] \tag{2.3}$$

where $P_i^\pi = \{j : \pi(j) < \pi(i)\}$. The disadvantage of this formula against 2.2 is that there are more addends to sum, but it can be useful in some situations.

Once reached this point, we wonder of what interesting properties the Shapley value satisfies and if it can characterized by some of them.

The answer is that it is the unique value that satisfy the following four properties:

1. *Efficiency*: The total payment of the game must be divided, i.e.
$$\sum_{i=1}^{n} \phi_i = v(N)$$

2. *Symmetry*: If i and j are two players who are equivalent in the sense that
$$v(S \cup \{i\}) = v(S \cup \{j\}) \qquad \forall S \subset N : i, j \notin S$$
Then $\phi_i(v) = \phi_j(v)$.

3. *Zero player behaviour*: The Shapley value of a null player is zero, $\phi_i(v) = 0$. A player i is null for v if $v(S \cup \{i\}) = v(S)$ for all coalitions in N.

4. *Linearity*: If two cooperative games described are by the gain functions v and w, the sum game $(v + w)(S) = v(S) + w(S)$ should correspond to the gains derived from v and w separately:
$$\phi_i(v + w) = \phi_i(v) + \phi_i(w) \qquad \forall i \in N$$
Furthermore, for every $a \in \mathbb{R}$,
$$\phi_i(av) = a\phi_i(v)$$

The three last properties are trivial to prove from the definition. Thus, let us prove efficiency is satisfied. Using the second definition to calculate the Shapley value,

$$\sum_{i=1}^{n} \phi_i(v) = \sum_{i=1}^{n} \frac{1}{N!} \sum_{\pi \in \pi(N)} [v(P_i^\pi \cup \{i\}) - v(P_i^\pi)] = \frac{1}{N!} \sum_{\pi \in \pi(N)} \sum_{i=1}^{n} [v(P_i^\pi \cup \{i\}) - v(P_i^\pi)]$$

$$= \frac{1}{N!} \sum_{\pi \in \pi(N)} [v(N) - v(\emptyset)] = v(N)$$

In example 2.1.1, the Shapley value results $\phi = (1/3, 1/3, 1/3)$

Part II

Addressing attribution problems by means of the Shapley Value

Chapter 3

Using cooperative games for attribution problems

The main goal of this monograph is to use game theory to build attribution models. The first task to solve is to choose a suitable game $G = (N, v)$. In this game, the set of players N would be the different values that can take a single feature of the data [1].

Thus, in example 1.3.1, if we choose the feature device, the set of players is:

$$N = \{PC, Tablet, Smartphone\}$$

The next problem that arises is to choose the worth of the game v for each coalition. It has not been immediate. The first idea that occurs immediately is to set the worth of every coalition as the amount of conversions accomplished by the coalition. If we note by C_T (equivalently $C_{j_1,\ldots,j_l}$, $T = \{j_1, \ldots, j_l\}$) the number of conversions achieved by $T \subset N$, the worth of every coalition is simple:

$$v(T) = C_T \quad \text{(equivalently } v(j_1, \ldots, j_l) = C_{(j_1,\cdots,j_l)}\text{)}$$

The problem with this definition is that the game is not superaditive. In particular, it is not monotone. In this manner, the single coalition $\{PC\}$ could achieve more conversions than the coalition $\{PC, Tablet\}$. In the example 1.3.1, the first coalition achieves 1 conversion, but the second does not achieve any conversion.

As a consequence, the incremental contributions 2.1 are negative. This is not intuitive, we follow the simple idea that says the more players in a coalition are, the better the coalition is.

Thus, we can think of adding all the conversions achieved by each element to the worth coalition. Then, the conversion associated game is defined in every subset of players $S \subset N$ as

$$v(S) = \sum_{T \subset S} C_T \tag{3.1}$$

[1]The algorithm is also designed to combine more than one feature and build a game with the cartesian product of the features.

where each C_T is the number of conversions achieved through the set of channels T simultaneously. If the previous formula is developed, it leads to:

$$v(S) = \sum_{j \in S} C_j + \sum_{\substack{j_1 > j_2 \\ j_1, j_2 \in S}} C_{j_1, j_2} + \cdots + \sum_{\substack{j_1 > \cdots > j_{k-1} \\ j_1, \ldots, j_{k-1} \in S}} C_{j_1, \ldots, j_{k-1}} + C_S$$

Result: The superaditive property is guaranteed in any game defined as above.

Proof. Let be $S, T \subset N$. Let us prove that $v(S \cup T) \geq v(S) + v(T)$:

$$v(S \cup T) = \sum_{U \subset S} C_U + \sum_{V \subset T} C_V + \sum_{\substack{W \subset S \cup T \\ W \not\subset S, T}} C_W \leq \sum_{U \subset S} C_U + \sum_{V \subset T} C_V = v(S) + v(T)$$

□

The goal is to calculate the Shapley Value on the previous game to give a fair attribution to each channel.

The Shapley value has many advantages as it allows us to calculate diverse attributions and it is very intuitive as it weighs the marginal contributions. However, its main drawback is the fact that its computation has an exponential complexity as n increases. In some cases that I have found throughout this dissertation, the number of different channels can exceed 40, so computationally it is extremely difficult to calculate the Shapley value. This is the reason why another approach needs to be chosen to calculate the attributions.

In [9] a sample method is used to determine approximately the Shapley Value in polynomial time. However, analyzing the behaviour of the different campaings for the clients, we observed the interaction among the channels does not usually exceed the number of 3 channels as it can be seen in figure 3.1.

Knowing this useful information, the following approach follows the formula of the Shapley value, but instead of calculating every element in the sum.

$$\phi_i(v) = \sum_{\substack{S \subset N \\ i \notin S}} \frac{s!(n-s-1)!}{n!} [v(S \cup \{i\}) - v(S)]$$

only the combinations

$$\phi_i(v) = \sum_{\substack{S \subset N \\ s \leq k \\ i \notin S}} \frac{s!(n-s-1)!}{n!} [v(S \cup \{i\}) - v(S)] \tag{3.2}$$

A similar idea can be used as well in the second formula of the Shapley Value:

$$\phi_i(v) = \frac{1}{N!} \sum_{P} \left[v(S_P^i \cup \{i\}) - v(S_P^i) \right]$$

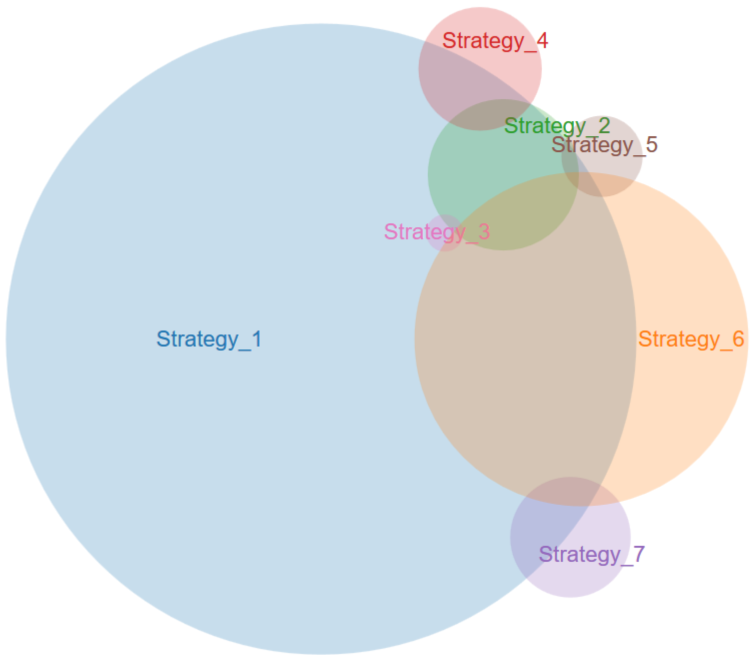

Figure 3.1: Venn Diagram for the interaction of users split by different strategies for an automobile company campaign.

But summing only among the permutations

$$\phi_i(v) = \frac{1}{N!} \sum_{\substack{\pi_k \in \pi_k(N) \\ i \leq k}} [v(P_i^{\pi_k} \cup \{i\}) - v(P_i^{\pi_k})] \tag{3.3}$$

where π_k are all the possible k-permutations of N. The k-permutations of a set N of $n > k$ elements are all the different subsets of N with cardinal k. For instance, all the possible 3-permutations of $N = \{1, 2, 3, 4\}$ are

$$\pi_k = \{1, 2, 3\}, \pi_k = \{1, 2, 4\}, \pi_k = \{1, 3, 4\}, \pi_k = \{2, 3, 4\}$$

P_i^{π} is intuitively defined after the definition of P_i: $P_i^{\pi} = \{j : \pi_k(j) < \pi_k(i)\}$.

As said, the interactions among channels are not often large in number, so a small k will suit.

3.1 Potential Information to extract using an attribution model

With the powerful tool of game theory we can evaluate the different sources that influence on conversions, (also on clicks and visits).

Among the things that our model can evaluate are: it can split the time of the day and evaluate what time is better. Also, it is available extract the location information of the data and can build an attribution model on the cities. The program can evaluate different strategies and it can rate the types of formats for our ads.

Example 3.1.1. If we have a campaign with the following data:

ID	Time	Event	Device	Format
User 1	2016-03-01 09:37:23	Imp	Smartphone	60x100
User 2	2016-03-01 09:37:43	Imp	PC	160x200
User 2	2016-03-01 09:37:58	Imp	PC	300x250
User 2	2016-03-01 09:38:03	Imp	PC	300x250
User 3	2016-03-01 09:38:20	Imp	Tablet	160x200
User 2	2016-03-01 09:38:23	Conversion	PC	300x250
User 3	2016-03-01 09:38:49	Imp	Tablet	160x200
User 4	2016-03-01 09:39:04	Imp	Smartphone	60x100
User 3	2016-03-01 09:40:28	Conversion	PC	300x250
User 4	2016-03-01 09:42:07	Imp	Smartphone	60x100
User 4	2016-03-01 09:44:31	Conversion	Tablet	160x200

Remark that the information a conversion row has involves the information of the last impression the cookie has. Thus, a conversion row brings implicitly 2 types of information: a conversion itself and the information of the last impression the cookie was impacted.

The conversion associated game as defined in 3.1, for feature device, with a last interaction attribution model would be:

$$v(PC) = 2,\ v(Tablet) = 1,\ v(Smartphone) = 0,\ v(Tablet, Smartphone) = 1$$

$$v(PC, Tablet) = 3,\ v(PC, Smartphone) = 2,\ v(PC, Tablet, Smartphone) = 3$$

So if we calculate the Shapley Value the result is very intuitive:

$$\phi(PC) = 2,\ \phi(Tablet) = 1,\ \phi(Smartphone) = 0$$

because the values obtained are the number of conversions lastly touched for each channel. For a standarized game,

$$\phi(PC) = \frac{2}{3},\ \phi(Tablet) = \frac{1}{3},\ \phi(Smartphone) = 0$$

Obviously, this is a small example and if we were using real data we would have to use formulas 3.2 or 3.3. If we make a presence attribution model for the same feature, the worth of each channel is:

$$v(PC) = 1,\ v(Tablet) = 0,\ v(Smartphone) = 0,\ v(Tablet, Smartphone) = 1$$

$$v(PC, Tablet) = 2,\ v(PC, Smartphone) = 2,\ v(PC, Tablet, Smartphone) = 3$$

The Shapley value is:

$$\phi(PC) = \frac{3}{2},\ \phi(Tablet) = 1,\ \phi(Smartphone) = \frac{1}{2}$$

And if the game is standarized,

$$\phi(PC) = \frac{1}{2},\ \phi(Tablet) = \frac{1}{3},\ \phi(Smartphone) = \frac{1}{6}$$

So the presence Shapley makes $Smartphone$ participate in the last conversion, and that is the reason why it shares some of that conversion attribution with $Tablet$.

3.2 Key Performance Indicators

Sometimes the number of conversions does not provide the required or desired information. Often it is desirable to have the information of how efficient the channels are and other useful data. This information is known in the business environment as KPI's (Key performance indicators). The most used ones in real-time bidding are the Conversion Rate, CPA, CPC and CPM.

Definition 3.2.1. The Conversion Rate for a given feature is the proportion of impressions served for each conversion, i.e.

$$CR = \frac{Number\ of\ Conversions}{Number\ of\ Impressions}$$

In example 3.1.1, the number of impressions for PC, Tablet and Smartphone are respectively 4,4 and 3. So in the last interaction case,

$$CR_{PC} = 0.5\ \ CR_{Tablet} = 0.25\ \ CR_{Smartphone} = 0$$

The Shapley presence leads to:

$$CR_{PC} = 0.5\ \ CR_{Tablet} = 0.125\ \ CR_{Smartphone} = 0.167$$

This can be better visualized in figure 3.2

Definition 3.2.2. The CPA (cost per acquisition) for a given feature is the amount of money that costs in mean every conversion. i.e.,

$$CPA = \frac{Total\ Cost}{Number\ of\ Conversions}$$

Definition 3.2.3. The CPC (cost per click) for a given feature is the amount of money that costs in mean every click. i.e.,

$$CPC = \frac{Total\ Cost}{Number\ of\ Clicks}$$

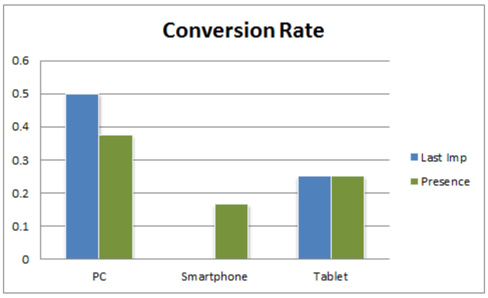

Figure 3.2: Conversion Rate for the different devices.

Definition 3.2.4. a CPM (cost per mille) is the amount of money that costs in mean every thousand of impressions, i.e.,

$$CPA = \frac{Total\ Cost}{Number\ of\ Impressions} \cdot 1000$$

3.3 Time Decay Modelling

As has been said previously, new channels can be created according to features of data or relevant information modifying some fields.

We have created a new feature which arises from splitting the time into intervals. Each interval will be now a channel to rate, and those channels will be split according to time before conversion.

So time has been split into minutes before conversions, creating 5 intervals. Another additional interval has been added to include the rest of the time as it can be seen in figure 3.3. Figure 3.4 shows similar results for hours.

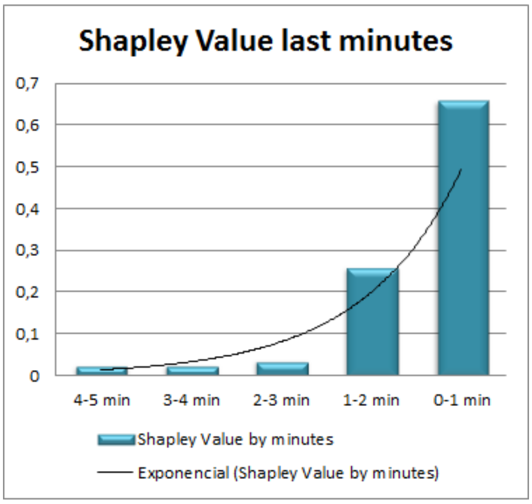

Figure 3.3: Time by minutes. The chart is a zoom of the last 5 minutes.

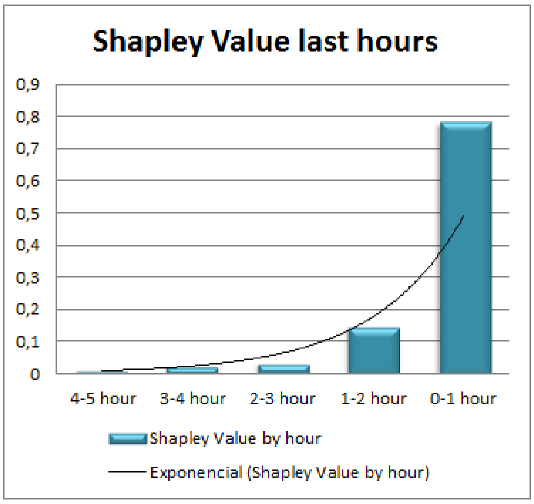

Figure 3.4: Time by hours. The chart is a zoom of the last 5 hours.

3.3.1 Creation of a Time Decay Model

According to figures 3.3 and 3.4 the behavior of conversions follows an exponential decay shape. In this manner, the attribution function does not get concentrated in the last impression as a last interaction model, but on the other side it is not constant as the presence model. So it is clear that the function has another shape, so the aim of this section is to roughly find it. Recency (time to conversion) distribution of impressions has been studied for different campaigns as figure 3.5 shows.

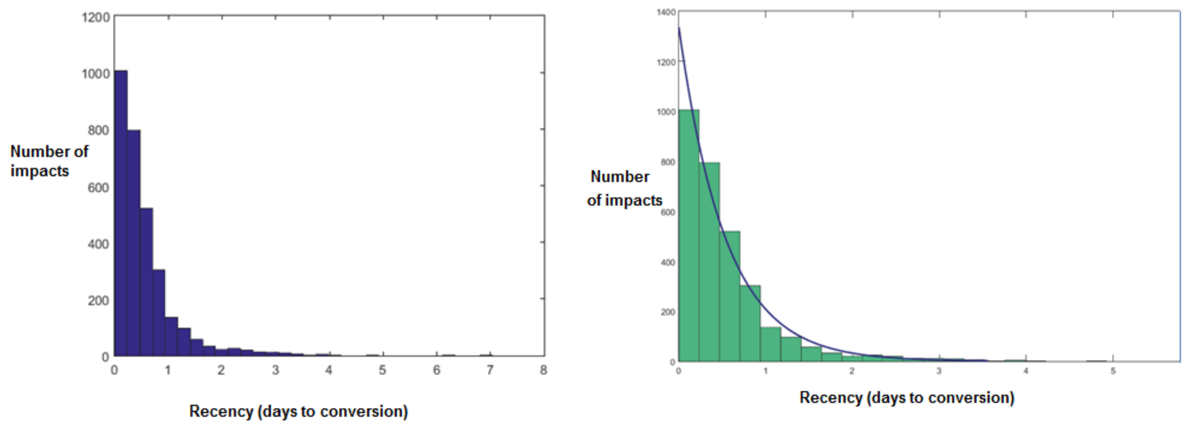

Figure 3.5: Time Decay.

There are some ways we could think to build an attribution model. The method chosen takes intervals such as the ones in figures 3.3 and 3.4 and combine those channels with the set of channels N we want to rate. Formally, a new extended game $G = (N^*, v^*)$ is defined, and the players of this game are

$$N^* = N \times \{Receny_0, Recency_1, \ldots, Recency_{n_i}\}$$

where $\{Receny_0, Recency_1, \ldots, Recency_{n_i}\}$ are set of intervals and $\times$ represents the cartesian product. Note that the computation of the Shapley Value for this game will take much longer than the original one because there are more channels now. Even if we use the formulas 3.2 or 3.3, we have to use a greater k parameter, because the interaction will increase due to the increased number of channels.

The worth of each coalition is simply defined by the number of conversions each coalition $S \subset N$ achieves in each interval:

$$v^*(S, Recency_j) = \sum_{T \subset S} C_T^j$$

Where

$$C_T^j = \text{Number of conversions achieved by } i \text{ in } Recency_j \text{ interval}$$

Also, as there is no interaction among the intervals,

$$v^*(S, R) = \sum_{j \in R} v^*(S, Recency_j)$$

where R is any set of intervals.

For example 3.1.1, if we have 3 intervals $\{Receny_0, Recency_1, Recency_2\}$, the players of the game would be

$$(PC, Recency_0), (PC, Recency_1), (PC, Recency_2)$$

$$(Tablet, Recency_0), (Tablet, Recency_1), (Tablet, Recency_2)$$

$$(Smartphone, Recency_0), (Smartphone, Recency_1), (Smartphone, Recency_2)$$

and we could set worth to each coalition using the method above defined.

The method to set a value to each channel is to solve the extended game $G = (N^*, v^*)$, so that each "player" $(i, Recency_j)$ has a value

$$x^*(i, Recency_j) = \text{Number of conversions attributed to } i \text{ in } Recency_j \text{ interval}$$

And finally the value of each channel is defined as

$$x_i = \sum_{j=1}^{n_i} x^*(i, Recency_j)$$

The last question to answer is what intervals we should choose. The cuts have been chosen in such a way it makes the histograms of impressions have the same area as in figure 3.6.

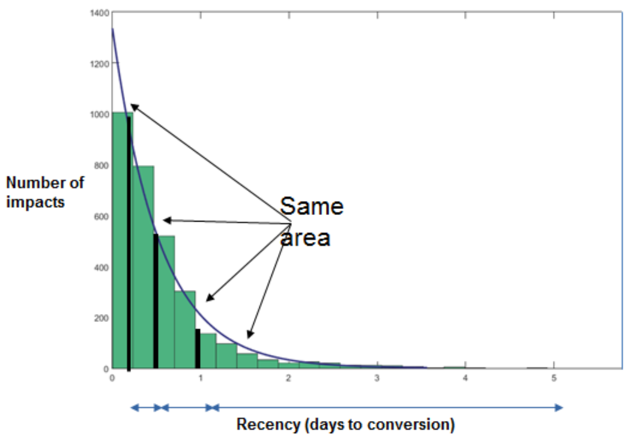

Figure 3.6: Where to cut recency?

Chapter 4

Results of Applying to Real Data

The data that is going to be used in this work comes from 2 campaigns for 2 different companies:

- A campaign in the USA for an automobile company from the 1^{st} March of 2016 to the 31^{st} of the same month. It consists of 7,621,852 rows or instances of 9 fields each row. The fields follow next: cookieID, town, device, timestamp, event, format, domain, website, lineitem and strategy.
- A famous Spanish insurance company between the 1^{st} of November of 2015 and the 1^{st} of January of 2016 (2 months). It has 7,044,755 rows of 9 fields each row. The fields are exactly the same as in the automobile campaign.

4.1 Comparison of Strategies

First of all, it is important to understand what the differences are between the two different ways to impact users: prospecting and retargeting. The first one aims to create awareness of the company, whereas the second one aims to accomplish as many conversions as possible. Figure 4.1 shows the proportion of conversions achieved by prospecting and retargeting in the automobile campaign.

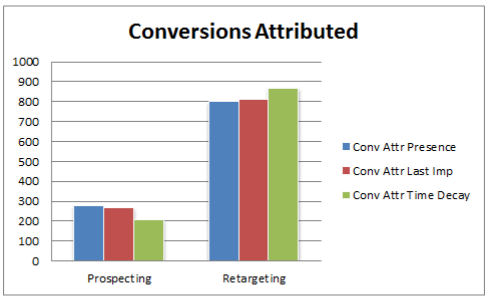

Figure 4.1: Conversions achieved for every strategy.

Obviously, the more impressions a strategy consumes, the more conversions it achieves. That is why the appears into scene to solve that problem. The proportion of impressions used by every strategy can be seen in figure 4.2.

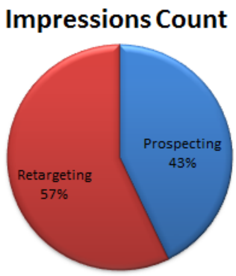

Figure 4.2: Proportion of impressions required for each strategy.

Retargeting has always better conversion rates as it shows figure 4.3. This occurs because the cookies targeted by this strategy have high probability of conversion, and therefore more conversions are achieved with the same number of impressions.

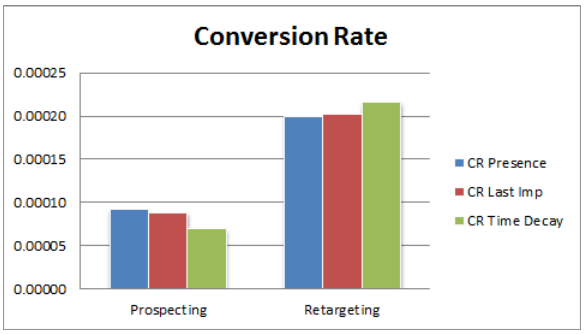

Figure 4.3: Conversion Rate for every strategy.

In figure 4.4 a scatter plot of cookies according to the impressions they received can be observed. The x axes shows the number of prospecting impressions (in a logarithmic scale) the cookie has received and the y axes represents the number of retargeting impressions received (in a logarithmic scale).

In order to see better the location of conversions, the ratio between converters and non converters has been increased as in figure 4.5. Note that the majority of converters have been impacted by a large number of retargeting impressions.

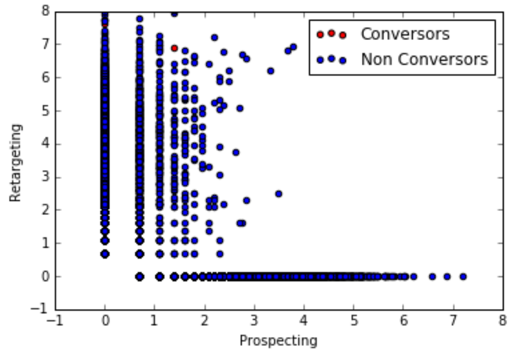

Figure 4.4: Ratio of converters 1:30.

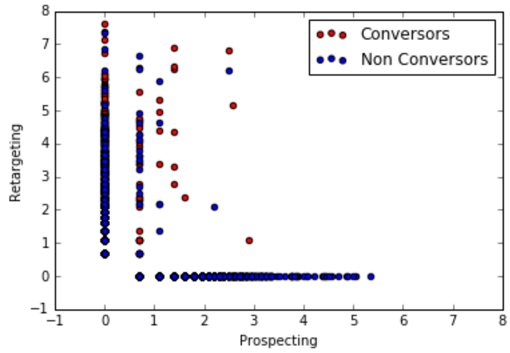

Figure 4.5: Ratio of converters 1:3.

4.2 Comparison of Substrategies

Different ways of approaching the goal of getting a good number of conversions spending a small amount of money are used in companies within the programmatic buying ecosystem.

For example, in some circumstances it could be useful to show a large amount of prospecting-type adds to create awareness. On the other hand, sometimes it could become more convenient to show more personalized adds. There are many options and parameters to choose for a given campaign.

For example, one retargeting-type line item could cover the goals of:

- Serving 3×10^5 impressions
- Achieving 1000 clicks
- Achieving 300 conversions

For example, for the automobile company the names of the strategies have been made anonymous, and the conversions accomplished by every strategy can be seen in figure 4.6. The number of impressions and the total cost in shown in figure 4.7. Strategy 1 is the one that gets the majority of conversions. However, it is the most expensive one, which makes think it will not have a good CPA. Indeed, as shows figure 4.8, Strategy 1 has a low CPA. This strategy is a retargeting-type one, because it consumes many impressions, but it is expensive, and that usually happens to retargeting.

On the top of CPA with have Strategy 2, which does not achieve many conversions, but in mean they are very cheap.

In figure 4.9 we can see the Conversion Rate and see that Strategy 6 has the best performance, because of the low number of conversions required to get 5000 conversions.

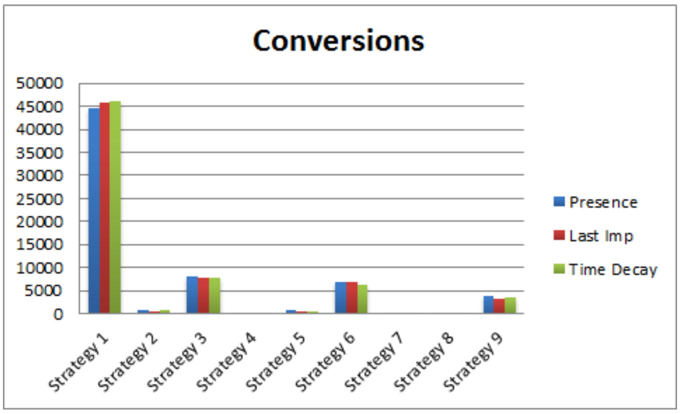

Figure 4.6: Conversions attributed.

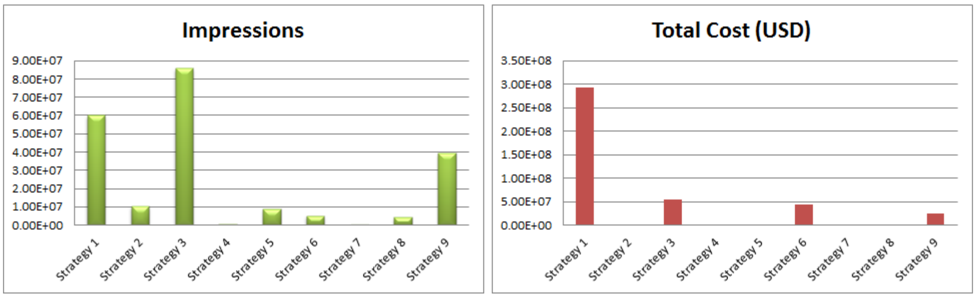

Figure 4.7: Total number of impressions and total cost.

Finally, according to the graphics, we might think at first sight that the models are very similar. However, this is still far from truth as shows figure 4.10. The difference between Shapley Presence and Last Impression rise to 60%.

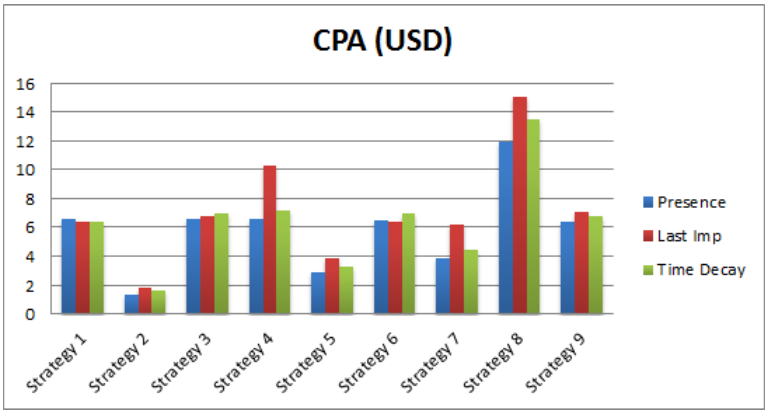

Figure 4.8: CPA.

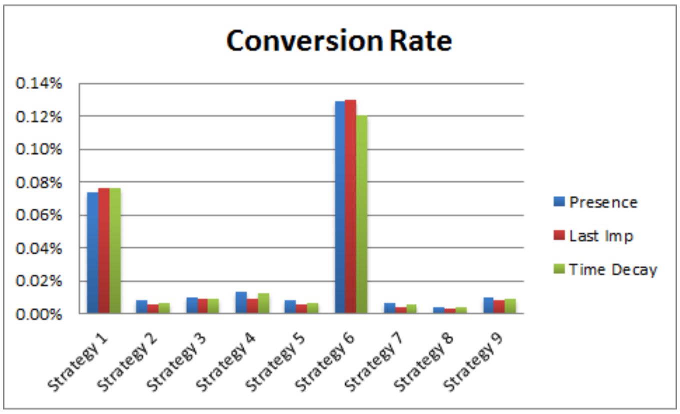

Figure 4.9: Conversion Rate.

4.3 By Day Time

As it has been said previously, we can evaluate the number of conversions achieved depending on the moment of the day. As can be seen in figure 4.11, there is an increasing trend of conversions in the beginning of the day.

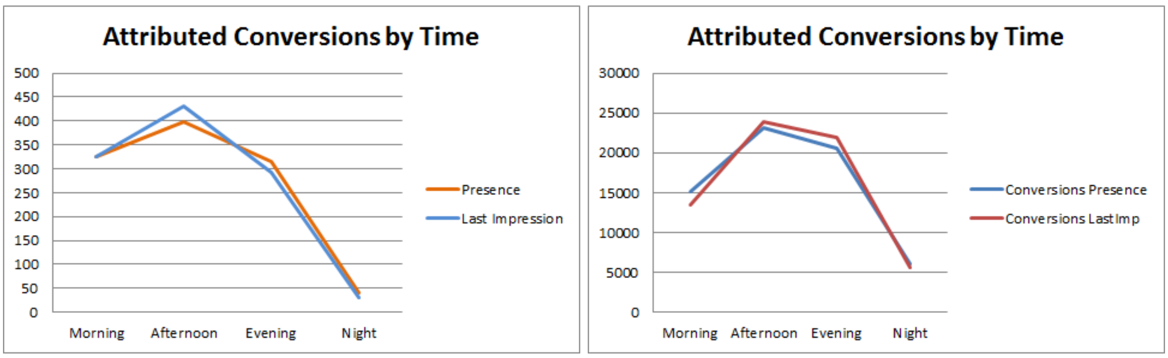

Figure 4.11: Distribution of conversions achieved by time for the insurance campaign (first plot) and for the automobile campaign (second plot).

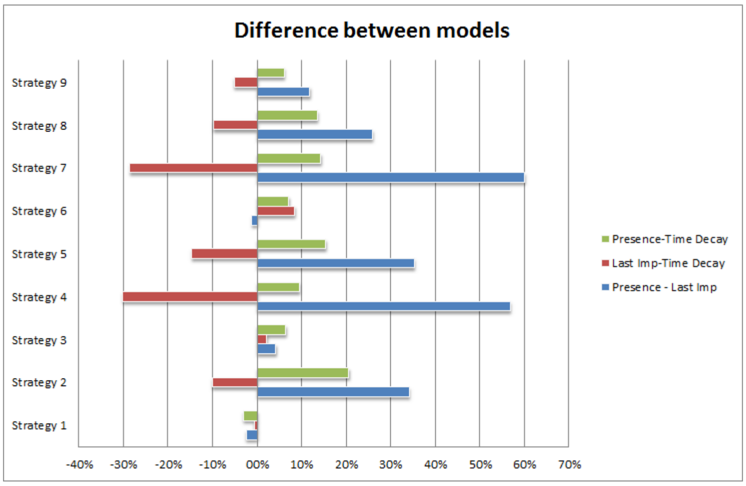

Figure 4.10: Difference between models.

However, we have to take into account the number of impressions as shown in figure 4.12. Note they have similar shapes to the conversions graphs except morning in the insurance campaign.

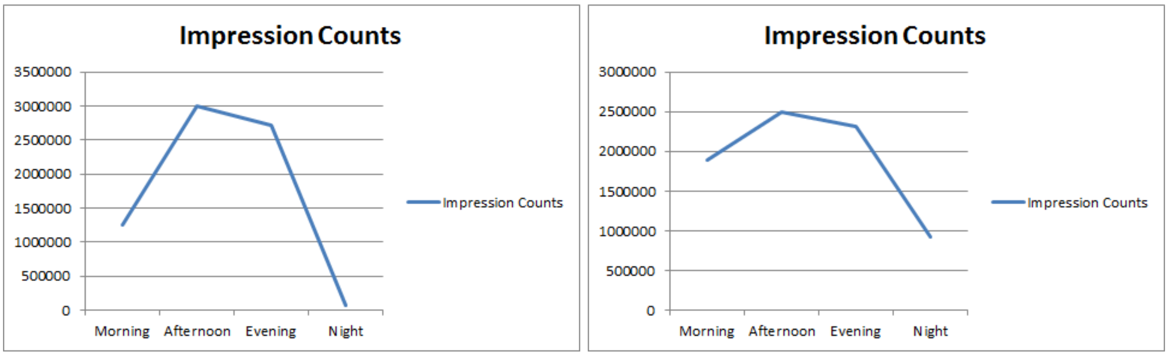

Figure 4.12: Distribution of impressions by time for the insurance campaign (first plot) and for the automobile campaign (second plot).

4.4 By Format

With the definition of channels as any feature that influences the conversion, we can analyze the impact of different variables such as the size of the ad:

In general terms, the largest an ad is, the most likely it achieves conversions. Conversion rate for the automobile campaign can be seen in 4.13 and CPA can be seen in 4.14

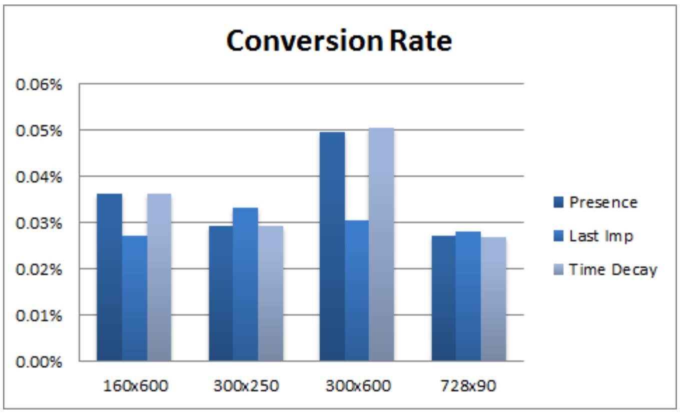

Figure 4.13: Conversion Rate.

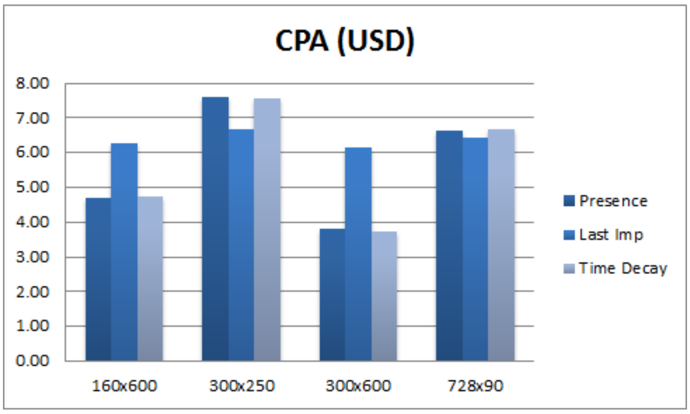

Figure 4.14: Cost Per Acquisition.

Chapter 5

Recapitulation

5.1 Summary of the work done

Digital publicity is a field in rise and omnipresent in our day to day. Diverse professionals of different disciplines, working for different companies, meet in this field to improve and discuss the current available tools. Among those disciplines, the most common ones are Data Science, Big Data, Informatics in general, Statistics and as seen in this work and Game Theory is very helpful when it comes to attribution models. Game theory (in particular, auction theory) can be also utilized to model and optimize the auctions that take place before the purchase of the space.

Regarding the development and completion of this thesis, much work has needed to be done. In the first place, understanding and going in depth in Online Publicity to understand better the problem. Once updated with respect to the situation of the problem, an introduction to Game Theory has been indispensable to investigate. In particular, some concepts of the Shapley Value which were first collected in the article of Lloyd Shapley ([8]). These concepts have been recalled in the Game Theory Unit in the Master.

The largest difficulty and also wealth of the thesis lies in the combination of two different disciplines: Online Publicity and Game Theory. The fact that with the attribution model we can prompt business KPI's of campaigns is an important step.

Moreover, there is a third difficulty that have been present throughout this entire job: the big volume of data and the complexity of calculations. In the beginning of this work, the Shapley Value was being computed in R storing all possible computations among channels in an inefficient manner. Soon, we realized that ways to focus the work needed to be changed, specially in computational terms. Then, it seemed suitable to work with python, in our opinion a very appropriate language in this case, given it is a lower level language, with comfortable access to work with data files. Also the package "itertools", which provides efficient iterators for permutations and combinations, has been very useful. The code in python takes more than 1000 lines and a schema of its functions and its relations can be seen in figure 5.1 in the annex.

Finally, it has been necessary to spend time choosing the most suitable campaigns and retrieving data, to make reports and plot descriptive graphs.

5.2 Conclusions

Let us recall in the introduction we fixed two main objectives for this work. One of them was building attribution models using Game Theory. Despite the fact barriers as complexity calculation of the Shapley value have been faced, the computational improve in 3.2 solve this problem because the interaction among channels is not excessively large. In this manner, we can say this goal has been achieved satisfactorily.

The other objective, inseparable from the first one, was building a more general attribution model than the last interaction one. Two of them have been created, the Presence attribution model and the Time Decay attribution model. The Presence model produces good results and it does not require hard computational work. As seen throughout this thesis, sometimes results of time decay presents large differences to Shapley Presence and Last Interaction or lies half way between them. However, often the Time Decay model does not produce meaningful differences with the Shapley Presence model. Therefore, it is generally preferable the Shapley Presence one, besides it is computationally cheaper. Future research lines might involve building improved time decay models and create customized models based on previous data.

Finally, we have to say that Cooperative Game Theory provides an effective and intelligible tool to rate channels and to choose suitable strategies. The calculation of the KPI's with three different models can be extremely helpful in industry to choose among tactics, impact timestamps, ad formats and so forth. Remark that the analysis done in this dissertation have been descriptive-type. The information provided opens new horizons for strategy optimization models.

Annex: programming

A set of scripts that follows the diagram of the figure 5.1 has been used throughout this dissertation. It consists of two scripts:

- ATMain: This is the main program. First of all, retrieves all the functions from "ATFunctions.py". Secondly, the main options such as graphics output, time modelling and so on are set up. Finally, with these options set up, ATMain initializes the class "cookies.py" with the name Paths.py and uses all the functions of that class as well as other functions of "ATFunctions.py".
- ATFuntions:
  - Cookies2path: Class with multiple options: Reads the raw data, creates a path of converters, calculates the Shapley value according to three models, etc.
    - _init_: Built-in python statement to initialize classes.
    - readInput: It reads the entire input file and saves the information in two dictionaries: conversor2path and nonconversor2path.
      - ProcessLine: Storage the information of every line keeping only the important fields.
      - dateTime2TimeStamp: Transform the dates to seconds.
      - parseDateTime: Needed by dateTime2TimeStamp.
    - shapleyFromConversorFile:
      - getUserChannels: For every cookie, creates a subdictionary with all the impacted channels .
      - calcIncrementalContributions: Previous to calculate the Shapley value
      - calcPermutationParticipation: Previous to calculate the Shapley value.
      - calcShapley: Calculates the Shapley Value.
      - standarizeShapley: Calculate Conversion Rates.

      * timeDecayEvents: Splits the events by time.
        · groupTimeDecayEvents: Aggrupates the events removing the time split.
      * Delnonconversors: Deletes the dictionary of non converters.
      * printConversors: Write all the converters in an output file.
  – logisticRegression: Computes a logistic regression.
  – treeClassifier: Computes a decision tree.
  – someGraphics: Depicts scatter plots.

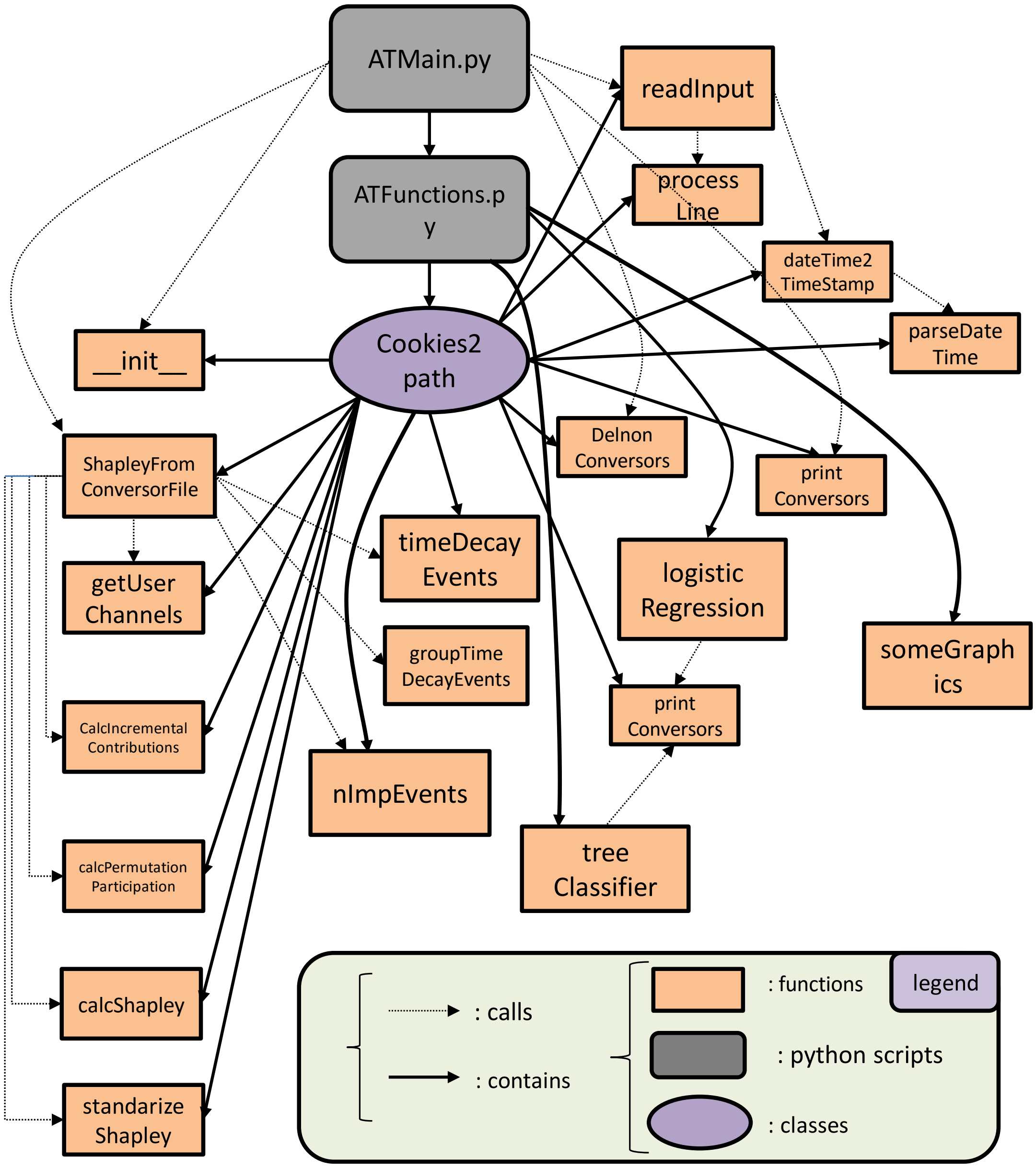

Figure 5.1: Function relationship diagram of the python code used.

Bibliography

[1] **Interactive Advertising Bureau**. *Libro Blanco de la Compra Programática*, 2014.

[2] **Media6Degrees**. *Bid Optimizing and Inventory Scoring in Targeted Online Advertising*. 2014

[3] **Robert Gibbons**. *Un primer curso de Teoría de Juegos*, 1993.

[4] **T. Hastie, R Tibshinari, J Friedman**. *The elements of Statistical Learning*, 2008, 2^{nd} edition.

[5] **Roger B. Myerson**. *Game Theory: Analysis of Conflict*. Harvard University Press, 1997.

[6] **John V. Neumann, Oskar Morgenster**. *Theory of Games and Economic Behavior*, 1944.

[7] **Albin E. Roth**. *The expecting utility of playing a game*, 1988.

[8] **Lloyd S. Shapley**. *A value for n-person games*, 1951.

[9] **Javier Castro, Daniel Gómez, Juan Tejada**. *Polynomial calculation of the Shapley value based on sampling*. Computers and Operations Research 36, 1726–1730, 2009.

[10] **Yuan, Shuai and Wang, Jun and Chen, Bowei and Mason, Peter and Seljan, Sam**. *An empirical study of reserve price optimisation in real-time bidding*, 2014.