ERASMUS UNIVERSITY ROTTERDAM

Erasmus School of Economics

Bachelor Thesis Programme BA&QM

July 4, 2021

Interpretable multi-touch attribution models

Student | *ID*
Saeed Rahimbaks | 484633

Supervisor: Kathrin Gruber
Second assessor: Wan P.

Abstract

With the growth of digital advertising, the understanding of the impact of media channels on revenue is a difficult task. The need for more accurate modelling techniques for multi-touch attribution is increasing, but with this, interpretability of these modelling techniques is decreasing. This paper aims to find a way to make use of newer accurate models that lack interpretation, while using a post-hoc interpretability method to retain interpretation. Besides the well-known Last-touch (LT) heuristic, we made use of three multi-touch attribution models: Bagged Logistic Regression (BLR), Support Vector Machine (SVM) and Random Forest (RF). In addition, we made use of a Shapley value (SHAP) implementation for post-hoc interpretability. In-sample and out-of-sample accuracy metrics were used to evaluate the predictive capabilities of the models, and the Mean Absolute Difference- and Mean Squared Difference metrics are used to evaluate the difference in attribution of two models. The RF model performed worst of all, and its attribution is largely different from the attribution of the other models and the LT heuristic. Both the BLR and SVM model determine the attribution of the media channels as expected, and can be used by companies for their specific multi-touch attribution problems. Furthermore, the results show that while the SVM model may be infeasible due to its large computation time, it does perform better in predictive accuracy than the other models. Despite its computation time, the SHAP method performed well and can be used as a post-hoc interpretability method.

Keywords: Multi-touch Attribution Model; Bagged Logistic Regression; Support Vector Machine; Random Forest; Shapley value.

Contents

1 Introduction **3**
1.1 Research problem . . . 3
1.2 Motivation . . . 3
1.3 Relevance . . . 3

2 Literature **3**

3 Data **4**
3.1 Data transformation . . . 5
3.2 Data analysis . . . 5

4 Methodology **6**
4.1 Last-touch Heuristic . . . 6
4.2 Preparation . . . 7
4.3 Bagged Logistic Regression . . . 7
4.4 Extensions . . . 8
4.4.1 Support Vector Machines . . . 8
4.4.2 Random Forest . . . 9
4.4.3 SHAP . . . 9
4.5 Model Comparison . . . 9

5 Results **10**
5.1 Last-touch Heuristic . . . 10
5.2 Bagged Logistic Regression . . . 11
5.3 Support Vector Machine . . . 12
5.4 Random Forest . . . 13
5.5 Model Comparison . . . 14

6 Conclusion **15**
6.1 Limitations and Future Research . . . 16

References **18**

List of Tables

1 Touch point statistics. 5
2 Last-touch attributions. 10
3 Bagged Logistic Regression marginal attributions. 11
4 Bagged Logistic Regression alternative attributions. 12
5 Support Vector Machine attributions. 13
6 Random Forest attributions. 14
7 Mean Absolute Difference. 15
8 Mean Squared Difference. 15

List of Figures

1 Bar plots of the media channels. 6
2 BLR Confusion Matrices. 12
3 SVM Confusion Matrices. 13
4 RF Confusion Matrices. 14

1 Introduction

1.1 Research problem

Digital advertisement continues to grow at a steady rate. According to RM (2020), we may expect the digital advertisement market to grow by 10% per annum between 2020 and 2026. Following this trend, the need for a better understanding of the impact of media channels is increasing for advertising companies. Previous models, such as first-touch, do not capture the importance of media channels well and are no longer sufficient. The simplicity of these models limits the accuracy. As such, the need for more accurate multi-touch attribution models is increasing. However, more accurate models often are more complex and consequently harder to interpret. The interpretability of a model is an important feature, as advertisement companies base their budget allocation decisions on these models. Therefore, the goal of this paper is to create a multi-touch attribution model, making use of highly accurate machine learning models while retaining interpretability through post-hoc interpretability methods. This trade-off between accuracy and interpretability is becoming a prevalent problem in machine learning as more intricate and complex black box models are being developed. As such, the findings of this paper may give insights regarding interpretation for companies that already make use of these complex machine learning algorithms. Additionally, advertisement companies who are new to the field may take interest in the results of this paper.

1.2 Motivation

Advertisement companies want to maximize customer conversions efficiently by allocating their budget according to the importance of their media channels. Thus, they need to know how to attribute importance to the many media channels correctly. Previous models often use heuristics that are too generalised and not data-driven. Other non-heuristic models tend to complicate interpretation, although they are more suited for prediction than heuristics. Therefore, more research needs to be done on this subject.

In this paper, we aim to incorporate complex black box classification models that increase accuracy, while using post-hoc interpretability methods to retain the user-friendly interpretation. As such, we tackle the issue of retaining interpretability and additionally make use of more accurate models.

1.3 Relevance

The correct or most efficient way of attributing importance to media channels can help advertisement companies greatly in maximizing customer conversions. Additionally, knowing which media channels contribute to conversions most can help companies reallocate their budget more efficiently. Therefore, the results of this paper are relevant to advertisement companies.

The findings of this paper are extremely practical, since the models are mainly data-driven. Companies and other advertisement agencies can incorporate these findings to their own market-specific data and use this to optimize profits. Therefore, the results can be tailored to the companies' marketing strategies and goals.

2 Literature

In this section we briefly explain the main findings and limitations of previous papers on this subject.

Shao and Li (2011) proposes to use a Bagged Logistic Regression (BLR) in addition to a simple probabilistic model. The simple probabilistic model is added to give advertisers the choice of

which model they see most fit to their data. This method has a simple deterministic structure, but still achieves accurate results. The Logistic Regression (LR) model is widely used in binary classification problems and boasts well interpretable parameters. Shao and Li (2011) adds on this method by including the bootstrap aggregating (bagging) method, which is designed to increase the accuracy of the model while also reducing variance. The results show that the BLR method outperforms the LR method in regards to their proposed V-metric. Furthermore, the BLR and simple probabilistic models perform similarly well. This indicates that both the probabilistic approach and the regression approach have potential in modelling multi-touch attribution. They argue that improvements can be made by controlling the dimensions of the dataset, as well as using tailored variables. By doing this, the model can become more compact and the amount of parameters compared to observations will decrease, making the model more accurate.

Kesteren (2015) compares eight different models that are interpretable. The well known first-touch and last-touch heuristics are used, as well as three variations on the LR model. Additionally, three Markov models (MAR) are used with orders 1, 2 and 3. Their findings show that, in line with expectations, the heuristic models perform least well since they are not data-driven. Furthermore, higher order MAR models outperform their lower order siblings. Kesteren (2015) proposes to incorporate timing differences by adding dummy variables to the already-known LR model. This novel model is found to perform best among all proposed models, and as such is worth investigating. However, this augmented LR model does not account for endogeneity issues that arise from the correlation between covariates and error terms.

Plas (2019) mainly focuses on the evaluation aspect of multi-touch attribution models. Four models are used, namely a heuristic models, a Shapley value model, the well-known Logistic Regression and a Markov model. Interpretability, accuracy and robustness were used as the evaluation metrics. The results show that no model is a definitive winner, as it seems there is a trade-off between the three evaluation metrics. The Shapley model scored best in regards to prediction accuracy, but is not robust. The LR model also has high accuracy, and opposed to the Shapley model it is robust. Markov chains were found to be robust, yet lacked in predictive accuracy. Furthermore, it is argued that the Shapley model and Markov model are decently interpretable while the LR model is not interpretable. Plas (2019) argues that improvements can be made by using several datasets opposed to just a single dataset. Furthermore, they argue that more information about revenues and costs of conversions and touch points respectively should be taken into account when creating a novel model.

3 Data

In this paper, we make use of the Altomare and Loris (2021) dataset. This dataset is obtained from the ChannelAttribution R-package. Consequently, we will retrieve and modify the data using the open-source coding language R. Further programming will be done in the well-known Python language, due to its abundant and highly optimised libraries. The dataset contains 4 variables, which in turn all have 10000 observations. The variables are *path*, *total_conversions*, *total_conversion_value* and *total_null*.

The *path* variable shows the customer journey. As an example, such a customer journey may look like $\alpha > \beta > \gamma > \alpha$, where the letters α, β and γ represent different media channels. In the dataset we find 12 distinct media channels. These media channels are all represented by Greek alphabet letters. However, moving forwards we will be using their full name in order to keep clarity. For example, α will become *Alpha* and κ will become *Kappa*. The variable *total_conversions* contains the total number of conversions for a given customer journey. A conversion has a broad meaning, as it represents a customer taking a specific desired action (e.g.

purchase, visit). The variable *total_conversion_value* contains total conversion value of a given customer journey, and will not be used in our analysis. Lastly, the variable *total_null* show the total number of paths that did not lead to a conversion.

3.1 Data transformation

The format in which the data is given cannot directly be used as input in the algorithms we will be using. The data needs to be transformed to fit our general notation, which we will further explain in Section 4.2. Twelve new variables are created by counting the total number of touches a media channel has had in a certain customer journey. These variables will then serve as the explanatory variables in the models we will use. Furthermore, we reproduce the customer journey rows as often as *total_conversions*, all leading to conversions. The same is done with *total_null*, leading to non-conversions. The newly created target variable, *conversion*, will display whether a customer journey lead to a conversion or not. As such, it is a binary dependant variable.

3.2 Data analysis

After the aforementioned transformations, some key descriptive findings of the dataset will be shown below.

Table 1: Touch point statistics.

Media channel	**Total touch points**	**Total conversions**
Alpha	162652	38019
Beta	45324	14079
Gamma	1938	568
Delta	50	17
Epsilon	6811	2070
Zeta	4794	1484
Eta	47135	13972
Theta	24584	6960
Iota	81556	23226
Kappa	3323	1068
Lambda	21527	6355
Mi	15	4

It is noticeable that channels *Delta* and *Mi* have significantly less total touch points than the other channels, especially considering the large amount of observations, namely 91989. We expect that channels with high total touch points will boast a higher attribution percentage. Because of this, we will remove the aforementioned variables from the dataset. As a result this decreases the amount of data we have by roughly 17%, which is beneficial for the machine learning algorithms we will use.

Furthermore, a remarkable observation is that there are eight consecutive customer journeys that have zero touch points. The reason for this is unclear, as the source of the dataset does not provide enough explanation. These eight observations will be removed as they can interfere with the machine learning algorithms we will be using. For example, divisions by 0 will occur if we leave the data as is.

The data shows a class imbalance in the target variable *conversion* of roughly $1 : 4$. This means non-conversions are found close to 4 times as frequent compared to conversions in the dataset. Class imbalance is an undesirable property for modelling, as machine learning models cannot cope with this issue automatically and will perform worse. A solution to combat this

issue is provided in the next section.

In Figure 1 we find the bar plots for all media channels. Do note that the y-axis is log-transformed. We find that some media channels do not occur as often in a customer journey as other media channels. For example, *Gamma* has no more than 10 touch points in a given customer journey, whereas *Alpha* occurs more than 50 times in some customer journeys. However, this discrepancy is in line with the distribution of total touch points, as seen in Table 1. As expected, all media channels show a downward slope, meaning that having more touch points of a given media channel in a customer journey occurs less frequent.

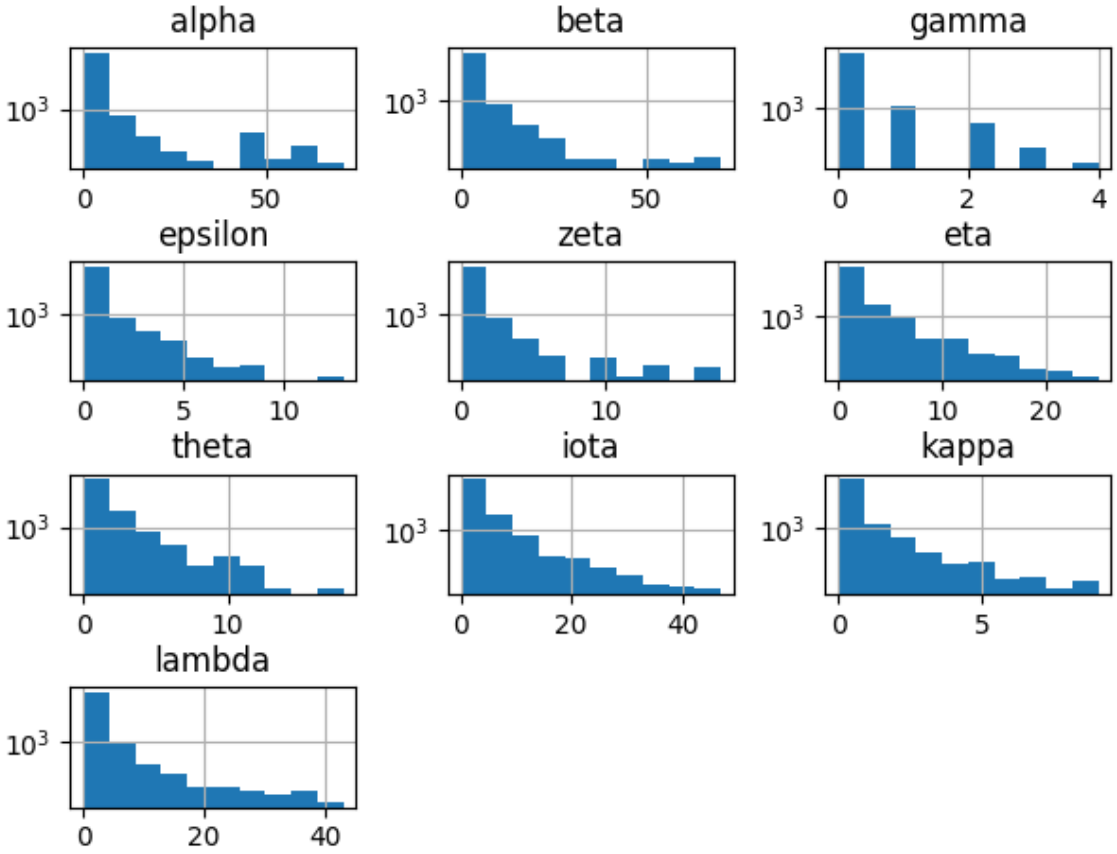


Figure 1: Bar plots of the media channels.

4 Methodology

4.1 Last-touch Heuristic

A common model used for attribution is called the last-touch (LT) heuristic. This method simply attributes 100% to the last touch point in a given customer journey. The idea behind this heuristic is that the last touch point is the final touch point for a customer before they either convert or not ($conversion = 1$ or $conversion = 0$), which may mean that this last touch point drove the conversion. In reality, this is not true at all. For example, you may imagine that a customer was attracted to a product on television, but only later bought this product after hearing a radio commercial. The radio commercial would then get all of the attribution, while in reality it was the television commercial that contributed most to the conversion or purchase. The LT heuristic completely ignores all previous information about a customer's journey, and

as such often performs much worse than the models we will propose further on. Additionally, it only uses information about customer journeys that end in a conversion. As such, a lot of information is lost. This heuristic is a single-touch attribution model, and will serve as a benchmark for the other models we use.

Multi-touch attribution models are preferred over simple heuristics such as the LT heuristic. The reason being that multi-touch attribution models paint a more complete picture of a customer's journey. Another advantage of multi-touch attribution models is that they give insights into non-conversions as well. These models attribute a percentage to all media channels instead of only one, and are therefore more accurate and more useful for interpretation.

4.2 Preparation

Preprocessing is needed for models to function as intended. In addition to the data transformation in Section 3, the following steps are taken.

The dataset will be split randomly into a train sample (or in-sample) and test sample (or out-of-sample). The proportions are 0.8 and 0.2 respectively. This is done to compute evaluation metrics for in-sample and out-of-sample observations. The in-sample observations will be used to train the models and compute its fit or accuracy, and the out-of-sample observations will be used to compute the forecast accuracy.

Additionally, all multi-touch models will have a modified objective function. Weights inversely proportional to class frequencies will be used in a weighted objective function, which replaces the original objective function. This is done because of the class imbalance described in Section 3. If this class imbalance is left as is, it is often the case that machine learning algorithms will disregard the minority class. Consider a class imbalance of $1:99$, then a machine learning model will have a 99% accuracy if it simply always predicts the majority class. The weighted objective function weighs the minority class more heavily and is thus penalised more if it disregards the minority class.

In this paper, we will mostly adhere to the notation used in Kesteren (2015) for convenience purposes. There are $i \in 1, 2, ..., N$ customer journeys, all having $j \in 1, 2, ..., J_i$ touch points in their respective journey. Let the j-th touch point of customer journey i be denoted as $v_{i,j}$. Each of the touch point is associated with a specific media channel denoted as C_k, with $k \in 1, 2, ..., K$. Let $C_{v_{i,j}}$ then denote the media channel associated with touch point j in customer journey i. Furthermore, let $y_i = 1$ denote a conversion in customer journey i and let $y_i = 0$ be no conversion in customer journey i. Shao and Li (2011) proposes to use explanatory variables, for example $x_{i,k}$, to represent the total number of touches a media channel k has had in customer journey i. That is:

$$x_{i,k} = \sum_{j=1}^{J_i} I\left[C_{v_{i,j}} = C_k\right], \tag{1}$$

where $I[z]$ is the indicator function which equals 1 if z is true and 0 otherwise.

4.3 Bagged Logistic Regression

The aforementioned explanatory variables are used in a LR model. This is a widely used binary classification method and is specified as follows:

$$P(y_i = 1) = \Lambda(\beta_0 + \sum_{k=1}^{K} \beta_k x_{i,k}), \tag{2}$$

where $\Lambda(x) = \frac{1}{1+e^{-x}}$ is the logistic function and $x_{i,k}$ represents the total number of touches a certain media channel k has in customer journey i. Shao and Li (2011) does not specify

how the estimated $\hat{\beta}_k$ are interpreted. However, Kesteren (2015) argues that a (naive) way of interpretation is through the usage of the (average) marginal effects. These marginal effects are specified as follows:

$$\frac{\partial P(y_i = 1)}{\partial x_{i,k}} = P(y_i = 1) * (1 - P(y_i = 1)) * \beta_k, \tag{3}$$

where $P(y_i = 1)$ is the logistic function used in Equation 2. We will refer to this as the marginal attribution approach. This method does not limit the attribution to be positive, and is therefore sub-optimal. However, Kesteren (2015) also proposes an alternative interpretation method. For each touch point $v_{i,j}$, calculate the estimated conversion probability $\hat{p}_{i,j} = \Lambda(\hat{\beta}_0 + \hat{\beta}_{C_{v_{i,j}}})$. These estimated probabilities are then normalized and directly used as attribution for each media channel. This approach will be named the alternative attribution approach.

We will be making use of bootstrap aggregating (bagging) in order to increase stability and accuracy and to decrease variability in the estimated coefficients β_k. This method samples m sub-samples from the original dataset, and performs the aforementioned LR algorithm on all of these sub-samples separately. In this paper, we create 10 sub-samples. It then aggregates the results and averages them to obtain more stable and accurate results. Note that the bagging technique increases the computation time of a machine learning algorithm, and should be used when the algorithm's complexity is not too large. For that reason, in this paper the relatively fast LR algorithm will be the only algorithm we use in combination with bagging. Henceforth we will refer to this bagging approach on the LR model as the BLR model.

4.4 Extensions

In this paper, we will add upon the previously mentioned classification method by making use of more sophisticated and complex machine learning methods. The methods we will use are: Support Vector Machine (SVM) and Random Forest (RF). We will compare the results with the aforementioned LR method. We expect that the SVM and RF methods will outperform the LR method with regards to accuracy, but will lose interpretability. To combat this, we will make use of a post-hoc interpretability method named SHapley Additive exPlanations (SHAP). This methods are computationally intensive, but may prove to be worth the effort.

The SHAP method is not model-specific, and as such is available for implementation in all models. This means that we can use these methods to retain interpretability while being able to make use of complex black box models. Consequently, this theoretically achieves our goal of finding an accurate model with interpretable results.

4.4.1 Support Vector Machines

The exact mathematical formulation of the SVM algorithm is very in depth and is therefore out of the scope of this paper. However, we will give a brief explanation in order to understand the basics for this method.

A SVM is a machine learning algorithm that is used for classification. It constructs so-called hyperplanes that separate the data points. This separation is optimised so that the distance between the hyperplanes and their closest data points is large. Afterwards, a new data point will be classified by considering on which side of the hyperplane it resides.

This method is very tuneable, as it makes use of a kernel function that can take many forms, in addition to having many regularisation parameters (such as the squared l2 penalty). In our case, we will be using a linear kernel. This choice helps reduce the incredibly large computation time and may perform similarly well as non-linear kernels.

The SVM method is not scale-invariant, which means it benefits from scaling down the data. Additionally, the SVM algorithm often performs worse when the data is not scaled, since it maximises distances. Large valued observations will then dominate over smaller valued

observations. As this is not desirable, we will be scaling the data. An additional benefit is that scaling will decrease the large computation time, which is a common issue with SVM models.

The estimated parameters for the SVM method are difficult to interpret, and even differ depending on the choice of kernel function. The parameters capture information about the hyperplanes that separate the data points. These hyperplanes can be interpreted well in a 2-dimensional feature space with a linear kernel function, as it then represents a simple linear equation. However, for higher-dimensional feature spaces the parameters become less simple to interpret. Although this method is well used in prediction, it is less useful for inferring relationships within the data compared to, for example, linear regression.

Therefore, we propose to use the SHAP algorithm to serve as our interpretation for the SVM model. Specifically, we will be using the KernelSHAP (Scott M Lundberg and Lee 2017) implementation in the open-source coding language Python to compute the Shapley values for the SVM model.

4.4.2 Random Forest

The RF algorithm is a tree-based machine learning method. It is a so-called ensemble method, meaning it uses aggregated results of several individual machine learning algorithms. This is quite similar to the bagging method described earlier.

A RF is an ensemble of several decision trees, hence its name. In our case, we will make use of 100 individual decision trees and aggregates their results. For our purpose, RF classifiers have an additional advantage over other machine learning algorithms. That is, tree-based models have a more tailored implementation of the SHAP algorithm, greatly decreasing its already large computation time.

The SHAP method will be applied on our RF model, as it is commonly regarded as a black box model. For this model we will be using the tailored TreeSHAP (Scott M. Lundberg et al. 2020) implementation in the open-source coding language Python to compute the Shapley values.

4.4.3 SHAP

The SHapley Additive exPlanations method, or SHAP method, is a post-hoc interpretability method that originates from game theory. SHAP explains the prediction of whether a customer journey results in a conversion by calculating how much each media channel contributes to the prediction. Game theory teaches us that the Shapley value shows us the importance of a player to the payout in a game. In our case, the media channels represent the players and the payout is the prediction of conversion.

The SHAP method trains a linear model on top of a given machine learning model. As such, it approximates the prediction of the machine learning model. This linearity provides the much needed interpretation for models that have little interpretation themselves.

4.5 Model Comparison

For each model, except the last-touch heuristic, we will be using two metrics in order to evaluate the models. We will use the in-sample accuracy score of the models to determine how accurate the models are. Furthermore, the out-of-sample accuracy scores will be used to evaluate the forecast performance of the models. These two metrics will be displayed in a confusion matrix. The confusion matrices in our paper will be 2 by 2 matrices as we have 2 classes for the variable *conversion*. The matrices are read as follows:

- Row 1, Column 1: Correctly predicted non-conversion (True Negative)
- Row 1, Column 2: Incorrectly predicted non-conversion (False Negative)

- Row 2, Column 1: Incorrectly predicted conversion (False Positive)
- Row 2, Column 2: Correctly predicted conversion (True Positive)

Additionally, as Kesteren (2015) proposes, the Mean Absolute Difference (MAD) of the attribution percentages of the models will be used to evaluate the difference in attribution each model has compared to another. This metric will be used on all models. The MAD between two models i and j is calculated as follows:

$$MAD(i,j) = \frac{1}{K}\sum_{k=1}^{K} |Attribution_{k,i} - Attribution_{k,j}|, \tag{4}$$

where we note that $MAD(i,j) = MAD(j,i)$, which we will call its symmetric property.

Another approach would be to calculate the Mean Squared Difference (MSD) between two model attributions. This method weighs larger differences between attributions more heavily. Therefore the MSD approach is more sensitive to outliers. This may give useful insights, as larger differences between the attributions indicate a discrepancy in the way two or more models calculate the attributions. A large MSD suggests that a company should closer inspect the models and decide which is best for their purpose. The MSD between two models i and j is calculated as follows:

$$MSD(i,j) = \frac{1}{K}\sum_{k=1}^{K} (Attribution_{k,i} - Attribution_{k,j})^2 \tag{5}$$

This equation also displays the same symmetric property as the MAD.

5 Results

5.1 Last-touch Heuristic

The last-touch heuristic model is fairly easy to compute. It simply aggregates the last visits of all customer journeys. Then we normalize these results to compute the attribution percentages. The results are displayed in the table below.

Table 2: Last-touch attributions.

Media channel	**Total last-touches**	**Attribution**
Alpha	8447	42.71%
Beta	989	5.00%
Gamma	92	0.47%
Epsilon	531	2.68%
Zeta	107	0.54%
Eta	4167	21.07%
Theta	653	3.30%
Iota	3355	16.96%
Kappa	230	1.16%
Lambda	1207	6.10%

Percentages may not add up to 100% due to rounding.

In this case, the attribution percentages are simply the proportion of total last-touches. Although it is a naive approach, it gives decent insights into the distribution of touches for the media channels. As this is a rule-based heuristic, its predictive capabilities are limited.

5.2 Bagged Logistic Regression

The results for the BLR model are displayed in the table below. As discussed, there are two attribution approaches for this model. These will be compared to each other and the LT heuristic model.

The attributions of the naive approach for the BLR model are displayed in Table 3. As explained in Section 4.3, the average marginal effects do not limit the attributions to be positive. This does not guarantee attributions will be negative, but as seen in the table below, in our case the attribution percentage for media channel *Alpha* is -3.39%.

Another remarkable finding is that the other attributions do not seem to remotely line up with the LT heuristic. Even though the LT heuristic is a simple approach, it does show the distribution of the attribution of media channels. We would expect the multi-touch attribution models to have improved results, yet still line up quite well with the LT heuristic. The MAD between the marginal approach and the LT heuristic is equal to 132.44%, which is extremely large. This shows that the marginal approach of attribution is not suited for the BLR model.

Table 3: Bagged Logistic Regression marginal attributions.

Media channel	**Attribution**
Alpha	-3.39%
Beta	8.11%
Gamma	16.73%
Epsilon	19.75%
Zeta	5.57%
Eta	14.67%
Theta	5.90%
Iota	5.95%
Kappa	19.92%
Lambda	0.00%

Percentages may not add up to 100% due to rounding.

The alternative attribution approach is displayed in Table 4. This approach does pose that attributions must be positive. The results for this approach are more in line with expectation, and will thus be used in further comparison instead of the marginal attribution approach.

A notable observation is that attribution percentage for media channel *Lambda* is equal to 0.00%. As seen in Table 1, *Lambda* has a large amount of total touch points (21527) compared to the other media channels. As such, the attribution for this media channel is quite unexpected. This suggests that *Lambda* is a media channel that is frequently visited, yet does not contribute as much to a customer's conversion.

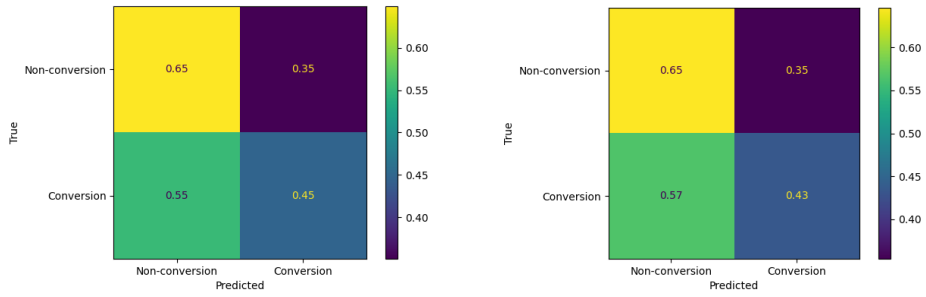
Furthermore, the media channels *Gamma*, *Epsilon*, *Zeta*, *Kappa* are also attributed quite low. However, these findings are more in line with the frequency of their visits, as seen in Table 1. A company's specific purpose may decide whether to include the aforementioned variables in future investments or to disregard them due to their low attribution.

Table 4: Bagged Logistic Regression alternative attributions.

Media channel	**Attribution**
Alpha	39.37%
Beta	10.55%
Gamma	0.79%
Epsilon	2.06%
Zeta	1.12%
Eta	19.24%
Theta	6.57%
Iota	19.35%
Kappa	0.95%
Lambda	0.00%

Percentages may not add up to 100% due to rounding.

The in-sample and out-of-sample evaluation metrics are found in the figure below. The confusion matrices show similar results for in-sample and out-of-sample. This suggests that the model is not affected by overfitting, which is a desirable characteristic. The model seems to predict non-conversions better than conversions. This can be a good- or bad property, depending on whether a company would like to interpret which features to stop investing in or which features to increase investments in.



(a) In-sample (b) Out-of-sample

Figure 2: BLR Confusion Matrices.

5.3 Support Vector Machine

The results for the SVM model are denoted in Table 5 below. The media channels *Theta* and *Lambda* are attributed less than expected based on their total touch points in customer journeys (24584 and 21527 respectively). The SVM model gives further reason to believe that *Lambda* is not a good indicator for conversions, despite its large number of touches. The media channel *Theta* may also suffer from this, although it does gain more attribution by the other models. *Epsilon* also has an unexpectedly low attribution for its total touch points (6811), but this is not as surprising as the aforementioned findings.

The MAD between the SVM and LT models is equal to 2.28%, and the MSD is equal to 9.86%. This indicates that the SVM model is quite in line with the LT heuristic model.

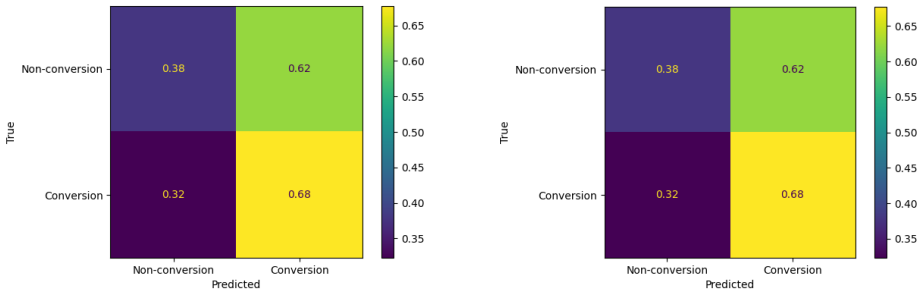
Table 5: Support Vector Machine attributions.

Media channel	**Attribution**
Alpha	47.54%
Beta	11.56%
Gamma	0.29%
Epsilon	0.61%
Zeta	0.25%
Eta	16.96%
Theta	2.99%
Iota	16.59%
Kappa	0.25%
Lambda	2.96%

Percentages may not add up to 100% due to rounding.

The confusion matrices of the SVM model are found in Figure 3. We notice a higher accuracy for predicting conversion, as opposed to non-conversions. This is quite the opposite of the findings for the BLR model. Companies that would prefer to predict conversions instead of non-conversions may benefit from making use of the SVM model compared to the BLR model.

A very remarkable finding is that the in-sample and out-of-sample have equal metrics (after rounding). This is most often not the case, but a good property to have nonetheless. It indicates that the model is not affected by overfitting in the slightest and is additionally quite consistent with out-of-sample forecasts. This model may then be the best choice for companies that have a larger focus forecasting conversions.



(a) In-sample (b) Out-of-sample

Figure 3: SVM Confusion Matrices.

5.4 Random Forest

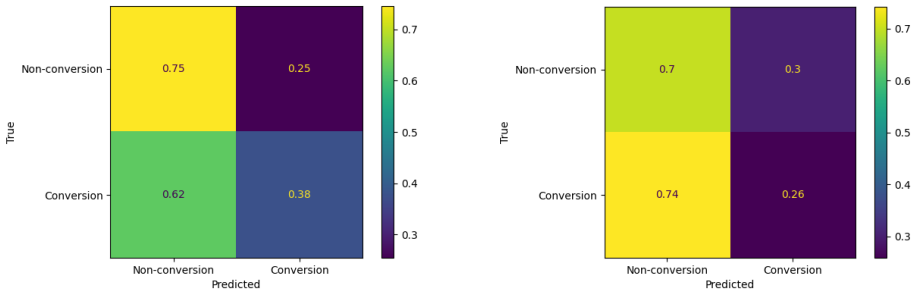
The results for the RF model are displayed in the table below. At first glance, the RF model's attributions are quite different from the BLR- and SVM model. This may be due to the large difference in the algorithms of the models, as the RF model is the only model that is tree-based. The attribution for media channel *Alpha* is lower than expected, given its total touch points of 162652. This is the most occurring touch point and is therefore expected to have the highest attribution percentage. However, occurrence is not necessarily a guarantee for attribution as we have seen in the results of the other models. The other attributions are quite in line with expectations based on the LT heuristic and previous results and findings.

Table 6: Random Forest attributions.

Media channel	**Attribution**
Alpha	21.16%
Beta	14.54%
Gamma	1.70%
Epsilon	4.11%
Zeta	2.53%
Eta	23.45%
Theta	6.91%
Iota	18.52%
Kappa	1.83%
Lambda	5.24%

Percentages may not add up to 100% due to rounding.

In the figure below we find the confusion matrices of the RF model. There is some discrepancy between the in-sample and out-of-sample results, which indicates that this model may suffer from overfitting issues. Additionally, we find that for both in-sample and out-of-sample the prediction accuracy for conversions is quite low. In fact, it is the lowest among all other models. It seems that the RF model may still undervalue conversions compared to non-conversions, despite the weighing techniques described in Section 4. Companies that have more need for predicting non-conversions might want to consider the RF model, but in general it seems to underperform compared to the other models.



(a) In-sample (b) Out-of-sample

Figure 4: RF Confusion Matrices.

5.5 Model Comparison

In this subsection we briefly look at the models compared to another.

In Table 7 we find the MAD percentages for all model pairs. The similar MAD percentages for LT, BLR and SVM suggests that the attribution of media channels by these models are quite in line with each other. This is not a guarantee that the true attribution must be equal to or close to their attributions, but it may provide an indication or baseline. As we have found in the previous subsection, the RF model attributed the media channels quite differently compared to the other models. This is further substantiated by the larger MAD percentages between the RF model and all other models. For this reason alone, the RF model may be disregarded as it does not provide the desired results.

Table 7: Mean Absolute Difference.

Model	*LT*	*BLR*	*SVM*	*RF*
LT	–	2.42%	2.28%	4.48%
BLR	2.42%	–	2.43%	3.81%
SVM	2.28%	2.43%	–	5.28%
RF	4.48%	3.81%	5.28%	–

The MSD percentages are found in the table below. As expected from the MAD percentages, the LT-, BLR- and SVM models have quite similar MSD percentages. Similarly, the RF model has extremely high MSD percentages. This is due to the fact that the MSD metric weighs larger differences much heavier compared to the MAD metric. This gives further indication that the RF model does not perform as good as the other models.

Table 8: Mean Squared Difference.

Model	*LT*	*BLR*	*SVM*	*RF*
LT	–	9.98%	9.86%	58.50%
BLR	9.98%	–	10.57%	40.13%
SVM	9.86%	10.57%	–	79.29%
RF	58.50%	40.13%	79.29%	–

6 Conclusion

In this paper we have used three methods to model customer journey data. Namely, these methods were used to calculate attribution percentages for all the features of the dataset. The attributions are expected to be somewhat similar to a Last-touch (LT) heuristic, which we use as a benchmark. These methods include a Bagged Logistic Regression (BLR), Support Vector Machine (SVM) and Random Forest (RF).

The BLR model used a model-specific approach to compute the attributions for the different media channels, performing quite fast and reliable. Its attributions were, as expected, quite in line with expectations. However, the model did not perform as well with regards to prediction and forecasting. Its accuracy for predicting non-conversions was higher than its accuracy for predicting conversions, which is usually not desired by the companies that use these models. The main advantage of this model is thus its very fast computation time compared to the other models.

The SVM model performed best compared to the other models. The attributions were in line with the LT heuristic. Furthermore, the SVM model boasts the best prediction capabilities. Opposite to the other models, it had a higher accuracy for predicting conversions than for predicting non-conversions. Most often this is a desired property of a model, as conversions are what drives profit in most cases. Furthermore, the SVM model had very similar in-sample and out-of-sample metrics, which indicates that this model is very consistent with predictions and is not affected by overfitting. The main disadvantage of this model is its very slow computation time. In our case, it was approximately 4000 times as slow as the other models. In this paper, we used a linear kernel function, which is one of the fastest kernels provided. This time scales non-linearly with the amount of observations and amount of features. For companies this is a

trade-off between performance and time or costs. Nonetheless the SVM model is most likely their best choice, especially for companies with little sample size.

The RF model is the only model we used that is part of the tree family. It underperformed compared to the other models. Its attributions were not as expected, and as such the Mean Absolute Difference (MAD) metric and Mean Squared Difference (MSD) metric were quite large for the RF model. There was discrepancy between the in-sample and out-of-sample prediction accuracy, which indicates that this model may suffer from overfitting. All in all, it seems that this model may not be up to par with the other models discussed and should not be considered by most companies for multi-touch attribution modelling.

The SVM- and RF models did not have their own attribution methods. For this reason, we made use of the SHapley Additive exPlanation (SHAP) method as a post-hoc interpretability method. For its purpose, the SHAP method performed very good. Apart from the RF model, the SHAP method computed attributions for the SVM model that were up to expectation. It is therefore our conclusion that the SHAP method is a good post-hoc interpretability method for black box models such as the SVM model. Its only disadvantage, and at that a very prominent one, is its computation time. On top of the already large computation time of the SVM model, the SHAP method for interpretation made this overall model very slow.

In general, a company must decide whether they want to focus on prediction of new instances or the attribution of media channels or a combination of both. For prediction purposes, the SVM model works best both in-sample and out-of-sample. For attribution purposes, the BLR and SVM model both work well and do not differ as much in their attribution ($MAD = 2.43\%$). Due to its fast computation time, the BLR model is then preferred for attribution. The RF model does not have an advantage over the BLR and SVM model in prediction and attribution, and should be left unconsidered for most companies. The SHAP method does improve interpretability for black box models and is suitable to use for multi-touch attribution, provided that the computation time is manageable.

6.1 Limitations and Future Research

This paper does not provide an all-encompassing model to be used for any multi-touch attribution task. Companies should choose models based on their own data and purposes. Improvements can be made in the selection of appropriate models, the encoding of data points and the computation time.

In this paper we chose the BLR, SVM and RF models. These models are well-known but are not necessarily the best model for multi-touch attribution modelling. More tailored models can be constructed, and more advanced algorithms can be used to produce better results. A family of models we did not use are the (Hidden-) Markov Models, which is out of the scope of this paper. Further research should be done into incorporating these models depending on the dataset. Other machine learning models should also be considered, as each has their own advantages and disadvantages. For example, some popular machine learning models include but are not limited to: K-Nearest Neighbours, Neural Network, Gradient Boosting and Boltzmann Machine.

Furthermore, other post-hoc interpretability methods should also be considered. The SHAP method works as intended, but its drawback of large computation time may be less ideal for companies that make use of large datasets. Global interpretation methods can be considered for companies that only need to explain feature importance opposed to explaining individual observations, which can save computation time. Other local interpretation methods such as the Local Interpretable Model-agnostic Explanations (LIME) method may be considered, as this method is considerably faster in computation than SHAP.

Another improvement can be made in the encoding of the data points. Incorporating carry-over effects add information about the past effects of the media channels, and may prove to be effective in increasing understanding of the underlying model and the prediction accuracy of the model. Furthermore, shape effects can be implemented in addition to the carry-over effects to incorporate diminishing returns on several consecutive media channel touch points. This, too, may prove to be worth researching. Lastly, as Kesteren (2015) proposes, dummies for first- and last-touches can be added as explanatory variables. As seen in Section 5, the LT heuristic does provide insights despite its simplicity. It seems as though last-touches carry more weight than the rest of the touch points in a customer journey. Therefore accounting for this may improve the model.

References

[1] Davide Altomare and David Loris. *ChannelAttribution*. `https://CRAN.R-project.org/package=ChannelAttribution`. Accessed: 13-05-2021. 2021.

[2] Joël Kesteren. "Searching for the Best Attribution Model". In: (2015).

[3] Scott M Lundberg and Su-In Lee. "A Unified Approach to Interpreting Model Predictions". In: *Advances in Neural Information Processing Systems 30*. Ed. by I. Guyon, U. V. Luxburg, S. Bengio, H. Wallach, R. Fergus, S. Vishwanathan, and R. Garnett. Curran Associates, Inc., 2017, pp. 4765–4774. URL: `http://papers.nips.cc/paper/7062-a-unified-approach-to-interpreting-model-predictions.pdf`.

[4] Scott M. Lundberg, Gabriel Erion, Hugh Chen, Alex DeGrave, Jordan M. Prutkin, Bala Nair, Ronit Katz, Jonathan Himmelfarb, Nisha Bansal, and Su-In Lee. "From local explanations to global understanding with explainable AI for trees". In: *Nature Machine Intelligence* 2.1 (2020), pp. 2522–5839.

[5] Joep van der Plas. "Evaluating attribution models on predictive accuracy, interpretability, and robustness". In: (2019).

[6] RM. *Global Digital Advertising Market 2020-2026 by Platform, Ad Format, Industry Vertical, and Region: COVID-19 Impact and Growth Opportunity*. Accessed: 10-03-2021. 2020.

[7] Xuhui Shao and Lexin Li. "Data-driven multi-touch attribution models". In: (2011).