Optimizing customer response to direct marketing initiatives

Francisco António Dias Amorim

Master's Dissertation

Supervisor: Prof. Vera Lucia Miguéis Oliveira e Silva

U.PORTO

FEUP FACULDADE DE ENGENHARIA
UNIVERSIDADE DO PORTO

Mestrado Integrado em Engenharia e Gestão Industrial

2018-07-02

Abstract

Players in retail, telecommunications and banking sectors quarrel for retention and development of their customer base in highly competitive markets, fact that steered companies towards targeting customer experiences. Direct marketing has been proven to increase customer's engagement and prolong their relationship with the companies, leading to higher customer equity over mass-marketing approaches. Despite the attention given by researchers to predicting individual's response to direct marketing initiatives and to the optimization of campaigns' specific goals, these two issues have been treated as solo acts. We propose a methodology that mitigates this gap. Firstly, state-of-the-art predictive techniques are applied and benchmarked to model customers' response to direct solicitations. The output of this first phase is then fed to a second stage prescriptive tool combining simulation and a sorting heuristic guided by the minimization of a regret function that, given capacity limitations and other business rules, maximizes the overall campaign response over a restricted number of solicitations. The approach takes into consideration the response elasticity of customers to the timing of contact.

The effectiveness of this combined tool is tested on a real-life case study through an end-to-end process on a company performing an upsell telemarketing campaign. The field experiment conducted yielded significantly higher response and sales rates for the treated group, while limiting the number of calls addressing the same client and keeping operator idle time at a low level.

The contribution of the present work to the state-of-the-art is twofold. At first, the benchmark made at various stages of the machine learning exercise has several practical implications that go beyond this specific exercise, many of which contest previously presented research. The results obtained, for instance, do not support the thesis of performing calibration on probabilistic outputs to improve error metrics and they show that tree based boosting algorithms are effective in response modeling. Then, the 27% increase in sales volume in the treated group, generated from an initial pool of 23% less call attempts, gives a practical validation to the proposed unified prediction-optimization framework. Furthermore, it showcases the unexplored potential of information lurking within companies' databases that can, with minor resource allocation, provide significant boosts to campaign profitability.

Resumo

Empresas em setores como o retalho, telecomunicações e banca debatem-se pela retenção e desenvolvimento da sua base de clientes em mercados altamente competitivos, facto que tem remetido as empresas a estratégias que direcionem as experiências ao consumidor. O *marketing* direto aumenta o envolvimento do consumidor com a empresa e prolonga a relação entre ambos, superando estratégias de *marketing* massificado neste aspeto. Pese embora a atenção que os investigadores têm direcionado à predição do comportamento do cliente individual no que toca à resposta a iniciativas de *marketing* direto e à otimização de objetivos específicos das campanhas promocionais, estes dois problemas têm sido tratados separadamente. É proposta uma metodologia que venha mitigar este desfasamento. Em primeiro lugar, técnicas preditivas avançadas são aplicadas na modelação da resposta dos clientes, sendo posteriormente comparadas. O resultado desta primeira fase é alimentado a uma segunda camada prescritiva que combina técnicas de simulação com uma heurística de ordenação guiada pela minimização de uma função de arrependimento que, dadas restrições de capacidade e outras regras de negócio, maximiza a resposta global à campanha promocional sobre um número limitado de tentativas. A abordagem toma em consideração a elasticidade de resposta dos clientes ao momento de contacto.

A eficácia desta ferramenta combinada é testada num caso de estudo real numa empresa que realiza campanhas de *upsell* em *telemarketing*. A experiência de campo realizada demonstrou que o grupo que sofreu tratamento teve comportamentos significativamente melhores, em termos de taxa de resposta e vendas, face ao grupo de controlo. Tudo isto foi alcançado com um menor número de tentativas endereçadas a cada cliente e mantendo um nível baixo de ociosidade nos operadores.

A contribuição do presente trabalho para o estado da arte versa dois pontos-chave. Em primeiro lugar, a comparação feita ao longo dos vários estágios do exercício de *machine learning* tem implicações práticas que extravasam esta aplicação específica, contestando em alguns casos investigação até então realizada. A título de exemplo, os resultados obtidos não suportam a tese de que a calibração de *outputs* probabilísticos melhora as métricas de erro e mostram que algoritmos de árvores de decisão sequenciais (*boosting*) são eficazes a modelar a resposta. Adicionalmente, o incremento de 27% em volume de vendas registado no grupo tratado, gerado a partir de uma base de tentativas 23% menor, dá uma validação prática à metodologia simbiótica de predição e prescrição. Além disso, demonstra o potencial desaproveitado da informação que reside no seio de bases de dados detidas pelas empresas e que, com uma aplicação limitada de recursos, pode trazer incrementos consideráveis na rentabilidade das campanhas promocionais.

Acknowledgements

This dissertation marks an important milestone in my academic career, thus concluding my Masters degree in Industrial Engineering and Management at FEUP. The chapter that now comes to close was marked by a treacherous path, whose traverse was untroubled for me, due to the contributions of countless men and women.

I would like to start by acknowledging the team at LTPlabs for firstly providing me with an interesting and impactful challenge and for, then, following that up by making me question my boundaries daily. I am grateful for the tireless contributions of Eng. Paulo Pereira, Bruno Batista e Pedro Campelo, with whom I worked closely. Being a careful observer, I recognize that I picked up a lot from following your stellar example. Besides, I am thankful for the opportunity to work under the guidance of Dr. Pedro Amorim, whose contagious energy never ceases to amaze me. Appreciation is also due to João Alves who demonstrated otherworldly patience while giving me support over the more technical issues I faced. Lastly, to the remaining junior team at LTPlabs that shared much of the same challenges as I did, I am certain that the long talks we shared will propel into a far greater future.

I would also like to extend my regards to Prof. Vera Miguéis who was originally assigned a supervising position over my work and ended up redefining that concept. I would place it closer to mentoring, or even beyond that. In fact, I would like to expand that gratitude towards all those who elevate FEUP and the Masters degree in Industrial Engineering and Management to the highest standard. Undecidedness marked my academic path, but being in the latter stage of my four-year run at FEUP, I am profoundly grateful to not have missed this opportunity.

Lastly, lacking a better way to shine a spotlight on the unsung heroes whose wise advice, kind words or mere example landed me here, I dedicate to you the efforts I placed on this work:

Ana, Bruna, Cátia, Diogo, Eduardo, Francisco, Gonçalo, Hugo, Inês, João, Leonor, Margarida, Nuno, Óscar, Paulo, Quitério, Rafael, Simão, Tiago, Uma, Vera, Xavier, Zoraida.

There. One name per letter of the alphabet - a small set representative of a much larger and fully contained story. Family, friends, mentors, ordinary people whose trajectory has only find mine once. My path is, most certainly, the result of the added contributions of all of you.

Francisco Amorim

E eu, semente, comecei a germinar.

Contents

Acronyms and Symbols

aCRM	Analytical customer relationship management
ALIFT	Area under the lift curve
ANN	Artificial neural networks
AUC	Area under the receiving operating characteristic
BA	Boruta algorithm
CA	Customer account
CART	Classification and regression trees
CHAID	Chi-squared automatic regression
CLTV	Customer lifetime value
CN	Customer number
CRM	Customer relationship management
DT	Decision tree
FN	False negative
FP	False positive
GBM	Gradient boosting machine
HMM	Hidden Markov model
KDD	Knowledge discovery in databases
KPI	Key performance indicator
LIME	Local interpretable model-agnostic explanations
MCM	Markov chain model
mrMR	Minimum redundancy maximum relevance
NP	Nondeterministic polynomial time
PAC	Probably approximately correct
PCC	Percentage of correctly classified
POMDP	Partially observable Markov decision process
RF	Random forest
RFM	Recency, frequency and monetary value
ROC	Receiving operating characteristic
SVM	Support vector machines
TN	True negative
TP	True positive

List of Figures

List of Tables

Chapter 1

Introduction

1.1 Motivation

The retail, telecommunications service providers, banking and other sectors alike saw a huge surge in competitiveness, forcing companies to upscale their promotional activity. The traditional approach towards marketing focused on targeting the masses, conveying a general message that often times failed to comply with the individual expectations of customers. Although this approach is still in vogue nowadays, mostly to build brand recognition, it has been losing weight to the alternative of tailoring the message to the recipient (Ling and Li, 1998). Direct marketing is, thus, the alternative data-driven process focusing on customer-first experiences to build long-lasting mutually beneficial relationships between customer and company (Miguéis et al., 2017).

In general, targeting communications to the individual customer, besides requiring a more careful approach in designing the message, also implies a cost higher than that of traditional broadcast vehicles like outdoors, television commercials, radio spots and so on. Mistargeting individuals when directly addressing them can severely degrade their loyalty (Bickert, 1997). So can a policy of contacting too frequently with offers that fail to meet customer's expectations. Thus, a far from optimal direct marketing initiative may be very costly to the company deploying it.

On the other hand, when properly targeting, direct marketing initiatives can yield significant benefits. Baesens et al. (2002), when analyzing campaign profitability, showed through a practical experiment that a 1% increase in response rate to direct mailings can generate an additional cash inflow of 500 000 euros. Besides, it is well established in literature that direct marketing, when adequately deployed, can strengthen the loyalty of customers and can more easily attract prospects, benefiting both customer retention and attraction strategies.

The investments made by companies in gathering information about their customer base allows for a rich environment within which data mining tools can gather valuable knowledge. The premise of the work that unfolds in this dissertation rests on developing a prescriptive system that chases an optimal campaign design. Assisting that task will be the knowledge extracted from databases.

To build on that premise, a case study is brought along. The company at stake is a major telecommunication service provider that has reason to believe its outbound telemarketing operations, responsible for about 35% of all sales, are running in far from optimal conditions. Current practices disregard the individual customer's preferences of contact, leading to attempts being made at inopportune moments, severely harming the sales conversion rates. Being a major contender in the market and having a customer base with high average length of relationship, its databases are rich with information that can effectively feed prediction engines aimed at predicting response to telemarketing calls. Moreover, the volume of customers (in the hundreds of thousands) that qualify at any given moment for its upsell, cross-sell and prospecting initiatives ensures that even slight increases in the response rate can generate significant boosts in campaign profitability.

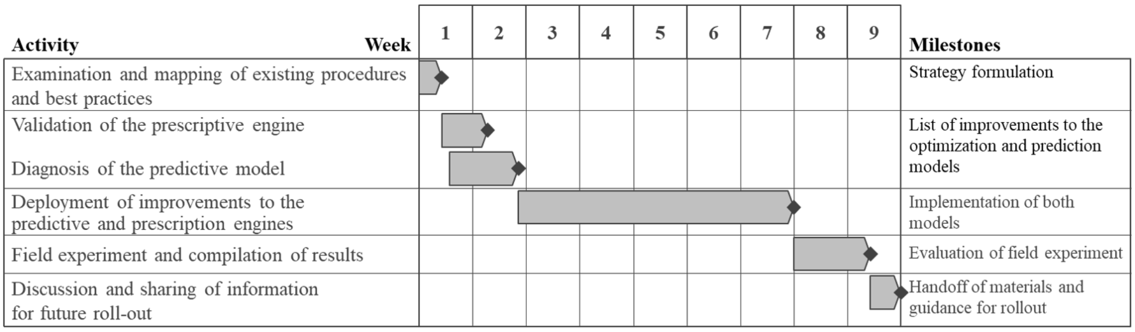
1.2 The project

Its large volume information panel, along with management teams' rising concern over leveraging analytics to improve operations, make the company at stake an ideal test bin for the prediction-prescription combo proposed. As stated, the consulting project conducted arose not from a problem, but rather from an improvement perspective. The aim was set at improving response to the outbound calls made, without burning though most of the customer base to do so.

Overseeing the progress of the project were two main stakeholders: the customer relationship management (CRM) team, responsible for defining the ideal characteristics of the campaign, and the operational team, responsible for following the guidance given by the CRM team and conducting the campaign itself. CRM teams have interest in maximizing the global value that each customer brings to the company (customer lifetime value). Operational teams placed more emphasis on short term objectives, like sales conversion rates. As such, a key requirement for the success of the project rests on complying with both requisites, that is, in helping to improve sales without degrading the future value of the customer base.

The end goal of the project is, thus, on demonstrating the effectiveness of a symbiotic view of prediction and prescription to meet the requirements defined. The performance will be assessed in a field experiment conducted on a large portion of customers during a one week span. Proven successful, the methodology will witness a rollout for all telemarketing campaigns done by the company.

As portrayed in figure 1.1, the project undergoes an initial phase of mapping the procedures and best practices currently applied, so that the strategy defined chases a congruent goal. Afterwards, a list of improvements to both prediction and prescription models is compiled. Within a five week window those improvements are embedded in the tools and preparations are made in order to ensure a smooth field experiment. At a final stage, the results of the field experiment are compiled. Ensues a discussion over those results and guidance for the rollout phase.



Activity Week	1	2	3	4	5	6	7	8	9	Milestones
Examination and mapping of existing procedures and best practices	▨◆									Strategy formulation
Validation of the prescriptive engine	▨	◆								List of improvements to the optimization and prediction models
Diagnosis of the predictive model	▨	▨◆								
Deployment of improvements to the predictive and prescription engines			▨	▨	▨	▨	▨◆			Implementation of both models
Field experiment and compilation of results								▨◆		Evaluation of field experiment
Discussion and sharing of information for future roll-out									▨◆	Handoff of materials and guidance for rollout

Figure 1.1: Project's timeline.

1.3 Thesis outline

The structure of the present dissertation is as follows. In chapter 2 a broad view of the research efforts in the area of analytical direct marketing is presented. Chapter 3 ensues with a more technical look at the multitude of concepts applied further along in this dissertation. A brief description of the case study, brought along as trial run, is provided in chapter 4. The following chapter is dedicated to showcasing the prediction/prescription methodology, firstly in a broader sense, and then in its adapted format, as applied to the case study. Chapter 6 displays some preliminary results, as well as the outcome of a pilot test ran on a real telemarketing operation. Ultimately, in chapter 7, some conclusions of the work are presented, along with a discussion on what that work adds to the current state-of-the-art. Some concerns over the applied methodology are raised and some future enhancements are catalogued.

Chapter 2

Literature review

The current chapter has the goal of presenting the research efforts made in the area of analytical direct marketing. Section 2.1 emphasizes the awareness that individual/segment targeting has raised over mass-marketing accompanying the large growth in individual consumer information availability, over the last decades. Then, section 2.2 brings to the discussion the areas that analytical direct marketing addresses, presenting some problem solving strategies explored in the literature. The chapter concludes with an allusion to the gap between the predictive and prescriptive streams within response modeling, with the current research acting to bridge them closer.

2.1 Current trends in Customer Relationship Management

As per mentioned in the introductory chapter, players in sectors like grocery retail, telecommunications and banking quarrel for retention and development of their customer base in highly competitive markets. Focusing on retail, Li and Feng (2017) noticed that the continuous drop in switching costs between badges in recent years, both in monetary and psychological terms, has shortened customer's life-cycle and has made some companies embrace loyalty programs.

Loyalty programs, binding contracts and other information vehicles alike allow companies to gather heaps of data about their customers, providing a view into their consumption behavior, demographic characteristics and lifestyle habits. This exponential increase in data availability allowed the surge of what Blattberg et al. (2008) call "database marketing" - the use of predicted and prescriptive analytics to improve the relationships between companies and their customers.

To enhance marketing productivity, firms quickly realized they could leverage data to address segments of customers which had similar needs. Focusing on marketing communication, rather than broadcasting a general message, advertisement could be targeted to specific customer classes (Bose and Chen, 2009). Dissociating themselves from mass-marketing, Blattberg and Deighton (1991) coined the concept of "addressable customer", a notion that is tied to the inception of Direct Marketing.

Bult and Wansbeek (1995) made one of the primary efforts in incorporating database information to enhance customer experience, introducing the RFM[1] model for segmentation purposes. The body of work that followed is quite extensive and using ever more sophisticated techniques: Minghua (2008) showed promising results applying a multilayer feed forward neural network to split retail customers on their consumption behavior; Kuo and Chen (2017) developed a customer segmentation model using particle swarm optimization (PSO) and artificial immune network; Nakano and Kondo (2018) explored a customer partition strategy according to purchase habits and channel (online/offline) selection. Miguéis et al. (2011) leveraged customer segmentation and market basket analysis to design segment tailored promotions for an European retailing company.

Some researchers, however, saw the increase in individual consumer panel data as an opportunity to dig even deeper. Rossi et al. (1996) started by quantifying the value that individual-level information can have in direct marketing endeavors. The observed 250% net gain in revenue achieved by personalizing coupons over blanket couponing[2], made them argue in favor of that specific approach. Building on this premise, Khan et al. (2009) investigated the value of one-to-one marketing in relation to segment-level and mass-marketing. In sync with Rossi et al. (1996), they showed that personalizing solicitations lead to a quantifiable increase in campaign profitability over uniform promotions. Venkatesan and Farris (2012) added to this argument by stating that customized coupons drop the perceived costs of redemption since they lower searching costs and, hence, can lead to purchase acceleration. Those conclusions receive further support from Mahar et al. (2017) who stated that introducing individual customer preference estimates boost campaign profits by 8.4% - 9.1%.

Contrastingly to what these findings lead to believe, empirical validations of the impact of direct one-to-one targeting, especially in the offline channel, stopped surfacing. Efforts have, instead, been focused on online channels, since e-mail communications and website recommendations are more readily customizable and cost-effective. The work of Wattal et al. (2012), for instance, found that propensity to redeem coupons in the online market is related to the inclusion of individual's characteristics upon designing the promotions.

2.2 Quantitative challenges of direct marketing

Nash (1984) identifies four main drivers of targeted campaign's success: i) decide the ideal moment to present the offer (Ching et al., 2004); ii) determine the remainder optimal communication characteristics, such as channel (Freitag, 2016); iii) elect the customers to focus on (Ma et al., 2016); iv) consider the right set of offers (Osuna et al., 2016).

[1] the customers' future behavior is mapped according to the recency (R), frequency (F), and monetary value (M) of their previous transactions

[2] promotional campaign not tailored for customers' specific needs

Most previous work developed in analytical customer relationship management (aCRM) falls strictly into one of the previous four categories. It is very seldom that one research team transversely attempts to look at more than one of these dimensions at once.

Adding to this argument, Talla Nobibon et al. (2011) structures the problem of searching for campaign optimality within direct marketing in two major steps, whose direct influence on one another calls upon a symbiotic view of both, which has hardly ever been explored in research. Those two steps comprise:

1. a data mining layer, where the response models are embedded;
2. a problem formulation and solution step where some marketing objective is mapped and optimize, restricted by a set of business rules materialized by campaign limitations.

2.2.1 Data mining applications in analytical CRM

The prediction of consumer behavior has been one of the cores of data mining applications in practical settings. Association rule mining, clustering, classification and forecasting/regression are the four main groups of aCRM techniques, according to Ngai et al. (2009).

Classification algorithms are very common in customer attrition prediction (Tsai and Lu, 2009; Larivière and Van den Poel, 2005); clustering techniques are popular in segmentation exercises (Li, 2013; Cao et al., 2009); association rule mining is the clear prevailing tool for market basket analysis and to feed recommender systems in next-product-to-buy models (Valle et al., 2018). Forecasting/regression models have an array of applications: Lismont et al. (2018) used them to predict inter-purchase time in a retail environment, while Larivière and Van den Poel (2005) used regression trees to predict a profit evolution function.

A stream of data mining applications attracting growing attention within aCRM is tied to response modeling - accurately predicting the outcome (be it participation, revenue generated, ...) of a campaign targeted at a specific customer (Haughton and Oulabi, 1997). Classical approaches, like the one presented by Asllani and Halstead (2015), make use of Markov chain models (MCMs) with states defined by RFM segmentation to predict customer response upon a solicitation. RFM response modeling, however, has received some criticism from Rhee and McIntyre (2009) and Blattberg et al. (2009) who argue it may not accurately depict consumer behavior.

More recent studies leverage the analytical power of machine learning algorithms to overcome the limitations mentioned above. Sacrificing some interpretability over discriminatory power (Olson and Chae, 2012), we find a plethora of work documenting successful applications of machine learning to response modeling, providing a rationale for its use (Moro et al., 2014; Javaheri et al., 2013; Miguéis et al., 2017). Illustrating this perspective, Li et al. (2016) applied an ensemble model using several well-known level-one learners like support vector machines (SVM), chi-squared automatic regression (CHAID), neural networks and logistic regression to predict customer response to telemarketing and emails. The achieved 74% to 132% lift that the treatment offered over the control group illustrates the predictive power of such approach. Furthermore, some work goes as far as benchmarking different algorithms for the same application (Asare-Frempong and Jayabalan, 2017).

Notwithstanding, the problem space for predicting consumer behavior upon a direct marketing solicitation is so vast, that research has only scratched the surface. For instance: forecasting models to predict promotional effects on sales of a product/category seem to receive far more attention in literature than the impact they generate on the customer lifetime value[3] (CLTV); response models to telemarketing initiatives are circumscribed to the financial sector, providing little proof that those approaches hold in other service sectors.

2.2.2 Seaching for optimality within response modeling

Talla Nobibon et al. (2011), having formally described the optimization problem into two main streams (section 2.2), proceed to focus their efforts on the latter point, applying linear programming as well as heuristics to determine the set of clients to be contemplated with promotional offers. Keenly, they point out that heuristics should be favored whenever dealing with large instances and bounded by time constraints.

The quest for optimality is again versed by Asllani and Halstead (2015). Customer's past purchase data is used to create a RFM segmentation. This RFM state coding is then worked on through a goal-programming approach, ultimately deciding on which customer segments to target to maximize campaign profitability. The multi-objective oriented solution is an evolution from their original work with linear programming (Asllani and Halstead, 2011).

Freitag (2016) looks at a slightly different problem. His concern is to devise a decision support system that, in a omni-channel environment, advises marketeers that have to decide which customer segments to assign to which communication channel, taking into account the business process and channel capacity. Again, mathematical linear programming was used to determine the solution.

In short, two points shine bright when assessing the content of the work available in academia: a wide variety of problems is tackled, albeit centered around segments of customers. At the individual client granularity level, there are very few widespread approaches outside next-product-to-buy recommender systems.

2.3 Contribution to the state-of-the-art

The previous sections make one thing clear: analytical marketeers are leveraging both predictive techniques and optimization engines to make direct marketing content into a vehicle to increase companies' customer equity [4]. Strikingly, research on these topics has been polarized, in the sense that there are very few documented approaches where a predictive model is embedded within an optimization platform.

[3] sum of the predicted net future cash flows between the customer and the company

[4] sum of the predicted net future cash flows between a company and its customer base. Equals the sum across the customer base of the individual's lifetime values.

One of those sporadic attempts to bridge the gap between prediction and prescription is provided by Ma et al. (2016). Their efforts were aimed at improving mailing decisions to optimize total accrued benefits (a proxy to customer lifetime value). The first step was devoted to forecasting response through a hidden Markov model (HMM), which then fed a second stage partial observable Markov decision process (POMDP) to derive optimal mailing decisions. Additionally, they support the view that traditional response forecasting models (logistic regression, decision trees, support vector machines, and so on), although great for pattern recognition, suffer from practical inadequacy as they focus on one selection period at the time, neglecting the multi-period response dynamics.

Reutterer et al. (2017) devised a stepwise approach with two main stages: a data mining layer, where they made use of clustering techniques and association rule mining to uncover frequent itemsets bought by different customer segments; and an optimization layer to filter those itemsets and select product categories to promote to specific segments to maximize spillover effects over non-promoted items. Their contribution goes beyond the boosts between 15% and 128% in campaign profitability, it stands out as lone effort to encompass this symbiotic view of prediction and optimization. I maintain, however, that the lack of available evidence supporting this hybrid approach is mostly due to corporate confidentiality, rather than overall system under-performance.

Considering the literature on direct marketing response, the contribution of the research uncovered in the next chapters to the state-of-the-art is threefold:

- First and foremost, this research aims at measuring the effectiveness of a unified predictive-optimization framework and its ability to perform adequately on a scalable business model;
- It expands the available body of knowledge on the application of machine learning algorithms to model response of direct marketing endeavors;
- The methodology is applied in a end-to-end process to a real-world problem, with a pilot being conducted on actual users to ensure that the theorized benefits are achievable in a practical setting.

Chapter 3

Theoretical background

The aim of the unfolding chapter is to acquaint the reader with some techniques and jargon used in machine learning (sections 3.1 through 3.5) and optimization exercises (section 3.6) with the sole purpose of improving the reading experience over the following chapters, where there is the assumption that that knowledge has, to some degree, been absorbed. The first section (3.1) is devoted to enumerating classification problems (many of which are transverse to all machine learning exercises) and the strategies to tackle them. Then, in section 3.2, different algorithmic choices are presented. Some further machine learning concepts are introduced from sections 3.3 through 3.5. Finally, a brief overview of optimization in general, and multi-objective optimization in particular, along with a hint of Decision Theory and simulation ensues in section 3.6.

3.1 Classification issues

A widespread problem faced within classification emerges from imbalanced proportions of observations in each class. In binary classification, it is common for the balance to tilt in favor of the negative class. The implications of training/testing a model over imbalanced datasets have been thoroughly identified in literature. Cieslak and Chawla (2008) point out that traditional learning algorithms may perform poorly when addressing such sources of data, as they tend to favor the larger, but less important classes. Subtle and rare observations become much harder to identify. Although this issue has been pinpointed, the approach to tackle it is still up for debate. Lin et al. (2017) purpose four main directions to follow: i) induce some form of algorithmic-level adaptation to handle rare events; ii) perform random over and/or undersampling before the training stage; iii) apply cost sensitive methods that more fiercely penalize misclassification on the rarest class; iv) make use of an ensemble of classifiers (see section 3.2).

Chawla (2009a) provides a complete overview of data mining over imbalanced datasets and makes a clear point – although there is a lot to be gained by tweaking the framework, there is no one-size-fits-all solution. Furthermore, Chawla (2009b) introduces the idea of pursuing alternative performance goals when training the model. Percentage of Correctly Classified (PCC, equation A.1), and other metrics alike, overrule the importance of rare cases. The evaluation criteria guides the learning process and must not ignore the importance of the minority class. The precision and recall measures (equation A.2) attend to the importance of the positive class, allowing for rules that chase rare events to be formulated. The F-measure (weighted harmonic mean of precision and recall- equation A.3) enables the fine tuning of the importance given to the false positive (FP) and false negative (FN) recognition rates. However, Powers (2011) reiterates that information retrieval metrics that ignore the performance on the more frequent class are inherently biased and do not take into account the chance level performance. Cohen's Kappa (equation A.4) addresses the latter point, but still fails on the former.

Notwithstanding, when the desired output is not to classify instances into black and white collections, but rather the probability forecast made of the binary event, an adequate metric for assessing the results is the Brier score (Brier, 1950). The mathematical formulation of the Brier score (equation 3.1) measures the effectiveness of the model by comparing the forecasted probability f_i with the true posterior probability o_i and it is by all means similar to the mean squared error metric used in regression exercises, but with the estimates and true values within $[0,1]$. Precedent for measuring performance of classifiers through the Brier score can be found in Providencia et al. (2018) and Jolliffe (2017).

$$BrierScore = \frac{1}{n} \cdot \sum_{i=1}^{n} (f_i - o_i)^2 \tag{3.1}$$

Whenever there are capacity limitations to perform the direct marketing contact, practitioners might be interested in evaluating the performance of their models for a strict portion of the observations. The lift of the *nth* percentile is calculated as the recall achieved on the top *nth* percent observations, regardless of how the model behaves (erratically or accurately) on the remaining cases. If, however, the ranking capacity of the model over the whole dataset is a measure of interest, then Li et al. (2016) explored a plausible solution. Their approach was to perform a weighed sum of the recall rate at each decile (lift index, equation A.6). This metric was proven to converge towards to the value of the area under the cumulative lift curve (ALIFT). The same researchers allude to the resemblance between ALIFT and the Area Under the Curve (AUC, equation A.8) of the Receiving Operating Characteristic (ROC), which displays the discriminatory power of a classifier in terms of false positive (FP) and true positive (TP) recognition rates over the whole spectrum of possible thresholds for class assignment. This threshold independence nature makes AUC desirable for performance assessment.

3.1.1 Over and underfitting

In machine learning the ultimate goal is to accurately map the input space into the output one through a mathematical formulation. The left-most portion of figure 3.1 illustrates the notion of underfitting. When underfitting, the mathematical formulation achieved by the model is extremely general and fails to capture intricacies that the algorithm is expected to pursue. At this stage, the model is agnostic to fluctuations in the training data and, thus, the prediction error on both the training and the test datasets is high.

In contrast, overfitting is the jargon used for the phenomenon of over training a model past the point when all significant and generalizable information is depleted. Under such conditions the model becomes highly volatile to changes in the input, even if those changes are mainly due to noise. The outcome is that, although the training error is minimal, the underlying truth has failed to be captured, causing the test metrics to have unsatisfactory results (right portion of figure 3.1).

The correct fit is somewhere in between the two extreme cases, where there is a fine equilibrium between the general perception and the capture of details, leading the prediction error on the test set to fall into a global minimum (Michailidis, 2018).

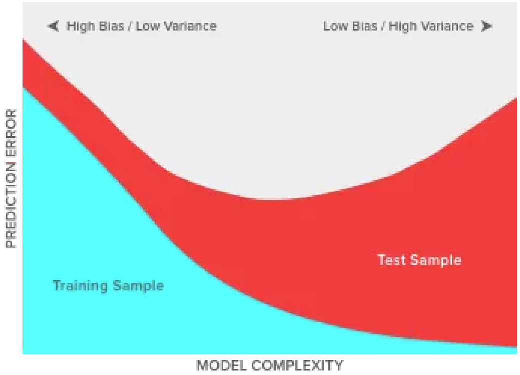


Figure 3.1: Finding the correct fit and avoiding under and overfitting (source Michailidis (2018))

The notions presented above explain why measuring the performance of the model on the training dataset can result in the ill-founded perception that, upon deployment, the model will have a far greater predictive power than what it is actual capable of. Thus, there is the need to split the available observations into training and test sets. The holdout method, using random or stratified sampling, rests on the assumption that both samples accurately depict the population's behavior. That assumption, however, seldom holds, especially for small subsets. The test partition drawn might be abnormally well suited for the model (lucky sampling), or the opposite might occur.

It it consensual in the current state-of-the-art that cross-fold validation provides an accurate picture of the model's performance upon deployment (Kohavi, 1995). The main caveat of k-fold validation is to partition the original data into k bins and iteratively, over k iterations, train the model on $k-1$ of those bins and calculate the performance/error measures on the holdout set. The overall result is achieved through averaging the scores of the k iterations. Besides providing a far more realist picture of the model's behaviour upon deployment than the holdout method, it might also boost overall performance as none of the available information is set aside to perform the test. The setback comes from requiring heavier computational loads. The path is, then, one of decreasing gains: as the available dataset increases in size, the simple holdout method yields less biased samples and the computational demand of cross-fold validation increases exponentially.

3.1.2 Feature selection

The under versus overfitting equilibrium does not rest solely on algorithmic choices made during the training phase. There is a certain ceiling to the model's performance once the features have been selected. Once again, failing to identify key predictors will cripple the robustness of the model. However, introducing several non-generizable variables that only pick up random deviations in the data (noise) might be equally harmful. Gauging relevance in advance might be impossible. Traditional approaches to feature selection include correlation analysis and chi-squared tests.

Nevertheless, some of these techniques are based on statistical hypothesis testing which tends to always reject the null hypothesis as the sample size (N) becomes larger. Furthermore, they often have severe scaling problems and focus mainly on eliminating redundancy rather than irrelevance from the dataset. In light of these shortcomings, several alternatives arose, like the Maximum Relevance Minimum Redundancy principle (Kamandar and Ghassemian, 2010), and the Boruta algorithm introduced by Kursa and Rudnicki (2010). In short, the Boruta Algorithm (BA) rests on random forest classifiers (see section 3.2.2). In its vanilla form, current RF algorithms provide variable importance estimates using the permutation accuracy importance principle, that is, by comparing variable importance before and after a permutation destined at removing the correlation between the predictor and the response variable (Strobl et al., 2007). To decide on the truly significant attributes, it is critical to ensure that the importance score is higher than that expected from random fluctuations. Hence, in a step-wise manner, the BA iteratively tries to discard variables that are statistically proven not to provide more insight than random attributes introduced artificially (shadow features), while confirming the relevance of those who do. Besides solving the all relevant problem, BA was shown to have robust scalability. Refer to appendix B for a complete overview of BA's pseudo-code.

3.1.3 Concept drift

Models can, to some extent, effectively approximate the true phenomenon that guides the behavior of the dependent variable as a function of the predictors. The issue arises whenever that same phenomenon is non-stationary in time, that is, whenever the approximated mathematical formulation uncovered by the model correctly identifies the underlying truth for some period of time and behaves erratically outside of it.

In direct marketing, this change can be induced by altering one of the pillars of campaign success: frequency, timing and other contact characteristics (channel); changing the value proposal or targeting different subsets of customers. Changes in customers' response dynamics may also be outside corporate control. Those are materialized in seasonality fluctuations and customers' preference adjustments.

Žliobaitė (2010) proposes some measures that, when taken during the training phase, might help mitigate such issues. To begin with, the practitioner is asked to make some assumptions on how the underlying truth might change over time and how that change can be correctly mapped in the inputs given to the model. Seasonality indexes are a great example of how one can anticipate future data drift. Furthermore, he adds some advice on other training procedures: retraining the model ever so often with recent data to allow adaptation to current conditions and applying a rolling window for training by either erasing old data or assigning heavier weights to fresher information.

3.2 Classification techniques

Still on the subject of finding the most accurate mathematical formulation, there are several algorithmic options that chase it. Typically, algorithms are classified into two major groups: single models and ensembles. Single models are a self-explanatory term illustrative of a unique instance of an algorithm that is trained to generate predictions. When the contribution of several single models is combined into a more powerful unique prediction engine, we enter the domain of ensembles. Generally speaking, ensembles are grouped into two main categories: homogeneous ensembles, whose single contributors are all identical, in the sense that they derive from the same algorithm; and heterogeneous ensembles, whose single contributors come from a diverse set of algorithms (as is the case of **stacking**). Within homogeneous ensembles, a further subdivision is found between parallel ensembles (like **bagging**) and sequential ones (such as **boosting**). The use of ensembles intends to improve predictions and decrease the variance and bias of single models.

3.2.1 Single classifiers

Decision trees (DT)

Systematize by Breiman et al. (1984), the classification and regression tree (CART) algorithm is a tree shaped set of exhaustive and mutually exclusive rules that can tackle both classification and regression problems. Given that soon thereafter other tree based learning algorithms emerged, like the C4.5 proposed by Quinlan (1986), with a very similar learning procedure, we will, from now on, address this family of learners as Decision trees. Decision trees' guiding heuristic aims at recursively partitioning the dataset (branching) trying to improve some cost function (impurity or entropy measure). In doing so, it follows a greedy approach, always selecting the best feature and splitting point for that effect (recursive binary splitting). The predictions assigned on each end of the tree ramifications (leafs) will be the average or the most frequent value of the response variable within the observations that fall into that partition.

Without a stopping criteria, decision trees are encouraged to grow until all observations within each node are perfectly homogeneous. That overfitting behavior is undesirable since the ability to generalize beyond the training instances is limited. Controlling the growth of the tree, there are some stopping criteria like the minimum count of training instances falling in each leaf, the maximum length between root and leaf, among others. To these pre-pruning strategies, one can add post-pruning heuristics aimed at trimming the nodes that are increasing the generalization error.

Feedforward Artificial Neural Networks (ANN)

ANNs mimic the nature of human though processing by connecting several nodes (neurons) each one processing the information received from several precedent nodes, applying a mathematical formula and sending the output to all the subsequent nodes. In its purest form, an ANN has two layers of neurons: an input layer with the same number of nodes as the dimensionality of the feature space; and an output layer with as many nodes as the dimensionality of the predictive space. This schema only allows for the capture of linear relationships between the dependent and independent variables. To induce non-linearity, a common practice is to introduce additional (hidden) layers of nodes between the two previously mentioned.

At a very high level, ANN work by learning out to tweak the weights assigned to the connections between nodes in different layers, so that the activations of neurons in the input layer translate to an activation of the neurons in the output layer that maps a response that is the closest to the instance being observed (Nielsen, 2017). This is done through calculating the gradient of a cost function for misclassifying training examples, expressed in terms of weights and activations, and incrementally taking steps in the direction where the gradient function becomes more negative (gradient descent).

Each layer's output ($\mathbf{a}^{(i)}$) is expressed as the weighted sum all the activations of the previous layer ($\mathbf{a}^{(i-1)}$) and the bias[1]($\mathbf{b}$) to which is then applied an activation function (typically sigmoid - $\sigma(x)$). In matricial form:

$$\mathbf{a}^{(i)} = \sigma(\mathbf{W}\mathbf{a}^{(i-1)} + \mathbf{b}) \tag{3.2}$$

3.2.2 Ensemble classifiers

Bagging

Breiman (1996) layed the foundations for bootstrap aggregating (bagging), an ensemble learning principle that induces diversity in the base learners by training them on a randomly (with replacement) drawn portion of observations from the original population. Bagging exploits the independence quality of the algorithms, given that they are trained in parallel, to reach estimates with lower variance.

The bagging principle's most notorious application is the **Random Forests** (RF) algorithm, introduced by Breiman (1996) himself, in an attempt to overcome the overfitting limitations that conventional CART displays whenever careful pruning is not applied. When an unseen observation is fed to the forest of DTs, the final score attributed comes from majority or weighted voting of the predictions of each level-one learner.

Additionally, it has been empirical proven that inducing algorithmic randomness on the individual decision trees contributes to reducing the variance and bias of the bagging algorithm. At each tree node, selecting the best attribute over which to split the observations from a random subset of the available predictors, although detracting from the performance of the individual model, has been shown to significantly boost the predictive power of the ensemble (Breiman, 2001). This perception led to some unconventional and rather extreme trials of inducing randomness into decision trees, with moderate success: Geurts et al. (2006) went one step further and besides choosing the attribute for splitting from a random subset, also selected the splitting point in a purely arbitrary fashion.

Boosting

Boosting algorithms used in current machine learning applications are some iteration of the original Probably Approximately Correct (PAC) learning framework showcased by Schapire (1990).

[1] the bias can be seen as some indication that the specific neuron tends to be active or inactive

The guiding heuristics used is to train a series of base learners over the complete training dataset in a step-wise manner, intensifying the costs of misclassifying an observation over each iteration, while decreasing the weights of correctly classified ones. Each learner is then given a voting right proportional to the accuracy achieved over the validation set. The score of a newly appointed observation is, thus, given by the weighted average votes of the level-one models.

Gradient Boosting Machines (GBMs) are tree based ensembles just like RFs, but rather than growing the trees in parallel, they are grown sequentially with each tree narrowing down and improving on mistakes made by the preceding ones.

Stacking - the Super Learner

In the realm of ensemble methods, traditional approaches (bagging and boosting) use several similar (often times weak) learners whose averaged contributions allow for the assembly of a far better performing model. In recent times, the preference has shifted towards performing stacking - the notion of starting of from a diverse set of strong single learners whose cross-folded predicted results feed a metalearning algorithm that optimally combines them to produce a more accurate output. The Super Learner, a creation of Laan et al. (2007), was proven to, under most situations and for the metric being optimized, perform at least as well as the strongest single model that fed it. Hence, it has been "theoretically proven to represent an asymptotically optimal system for learning" (LeDell, 2015).

3.3 Hyperparameter tuning

In their vanilla form, machine learning algorithms have a core configuration that allows for several problems to be addressed. It's easy to conceive that such configurations can have sub-optimal conditions to handle diverse sets of specific problems. For this reason, most algorithms allow for the learning process to be tweaked by actuating a set of levers - the **hyperparameters**.

Common approaches to define the set of hyperparameters that optimizes the learning behavior include an exhaustive (cartesian) search, where all combinations of factor levels being considered are trained, with the best set being chosen. The combinatory nature of this approach can easily drive the problem into computationally inconceivable sizes. A more suitable method comprises training a restricted number of models, selecting the levels of the hyperparameters through random sampling with replacement - random grid search (Bergstra and Bengio, 2012). While the former method guarantees that the optimal set of hyperparameters (within the search space) is found, the latter only assures a good step in that direction. The benefit comes from significantly lower computational times.

To ensure that the model tuning is done in ways that favor the improved generalization capacity of the model, rather than its ability to exploit statistical peculiarities of datasets, the performance assessment should be done over statistically pure (unseen) observations. As such, there is a need to introduce an additional step of validation upon deciding the ideal set of hyperparameters. The test metrics are then computed over a separate dataset, disjoint of the training and validation ones (Cawley and Talbot, 2010).

3.4 Extracting probabilities from supervised learning

Although AUC and lift-based metrics provide good insight into the discriminatory capacity of a classifier, they shed a dim light into how the output probabilities match the true posterior probabilities, since they focus mainly on ranking observations. Niculescu-Mizil and Caruana (2005) conjecture that, by their algorithmic limitations, Naives Bayes tend to over-push probabilities towards 0 and 1, while tree based boosting algorithms tend to do the opposite, applying a sigmoid distortion. Several alternatives extend the output of SVM and ANN (which is typically a score) into a probabilistic estimate (Bravo et al., 2010). Niculescu-Mizil and Caruana (2005) showed that, when applied, those alternatives output calibrated probabilities.

As a corrective measure to improve performance under these conditions, Platt (1999) proposes a transformation of the output - the Platt's scalling - that involves fitting a logistic regression to the output, effectively rescalling the probabilities without altering their rank. This ensures that rank-dependent metrics do not suffer modification, while the Brier score is, generally, improved. Niculescu-Mizil and Caruana (2005) showed empirically that for boosting trees the benefits of using Platt's scaling can be substantial. However, there is conflicting evidence on the success of such approach (Nee, 2014).

3.5 Uncovering the mysteries of black-box models

An increasing concern of practitioners dealing with ever more complex models is validating that the decisions made by the algorithm have rational backup. Realizing that a stacked ensemble has exceptional performance due to data leakage[2] might be a arduous task, especially considering that the tracking of how the prediction came to be is extremely complex and hidden from sight. The same holds true when an identification field is wrongly passed to the training dataset.

Fortunately, some tools that allow for trust to be built on the models created have been developed. Outputs such as variable importance and sensitivity analysis of the impact of major predictors help achieve this goal. It was with this ambition that Ribeiro et al. (2016) engineered the Local Interpretable Model-Agnostic Explanations (LIME) framework. It builds on the premise that any explainer should: provide at the very least local fidelity[3] in order to ensure global fidelity; be model-agnostic, that is, it should not rest on assumptions that hold true only for a selective group of classifiers; and yield an intelligible (interpretable) connection between the input variables and the output generated.

Let's assume a black-box model has found a complex and highly intricate(non-linear) function $f : I\!R^n \to \{0,1\}$ (unknown to LIME) that maps the input variables into predictions. Assuming there is interest in understanding the prediction $y \in \{0,1\}$ made on observation $X \in I\!R^n$, LIME starts by sampling the neighborhood of X and mapping those observations into the output space through f. It then weights each output of the sampled neighbours according to the proximity to

[2] allowing information known only a posteriori to be fed into the training observations

[3] ability to provide accurate explanations within a neighbourhood of an observation

X, ultimately fitting a linear model. That linear model is locally faithful, but not globally so, and is easily explainable. A visual explanation of this principle can be found in figure 3.2. The main premise behind LIME is that by explaining a set of individual representative instances, one can expect to globally understand the model.

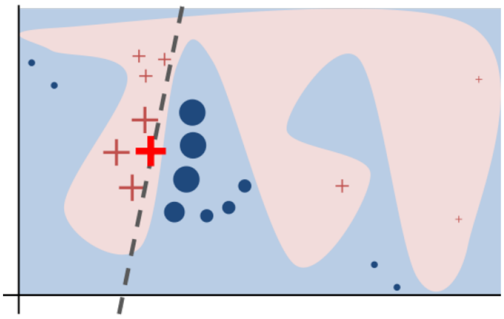


Figure 3.2: Graphical illustration of the LIME principle (Source: Ribeiro et al. (2016)). The background represents the function f as mapped by the black-box model; the dashed line showcases the local faithful approximation that LIME performs; the bold cross illustrates the observation X being explained; the dots and crosses represent instances of positive and negative classes with the respective size mapping the proximity to the observation being studied.

3.6 Prescriptive tools

3.6.1 Optimization: a high-level look

Optimization is concerned with finding the values to be taken by decision variables to best meet a certain objective, without violating the defined constrains (find the best feasible solution).

When complexity of the problem being dealt with escalates quickly with the size (such as nondeterministic polynomial time (NP) problems[4]), heuristic algorithms search for good (not necessarily optimal solutions) within a conceivable time frame. Greedy heuristics, for instance, always choose, at every step, the move that grants the largest improvement to the objective function, disregarding the consequences that act might generate later on (Maringer, 2005). The easiness of implementation is paid for by a greater risk of being stuck in local (not-global) optima.

Multi-objective problems

Realistic problem formulations, however, require several objective functions to be considered simultaneously, often times with conflicting goals. Under such conditions, directing the optimization at fulfilling one objective can lead to unacceptable solutions in regards to the others (Konak et al., 2006).

General approaches to tackle multi-objective problems comprise the combination of individual objectives into a single composite function by computing an utily function or performing the weighted sum method. These pondering objective methods generate a single objective function, to which are then applied common optimization strategies. An alternative is the application goal

[4] problems for which there is no known deterministic algorithm that extracts the optimal solution in polynomial time

programming. In goal programming, the decision taker sets out a prioritized list of goals he/she is set on achieving (Lee, 1972). The best solution is the one to minimize the deviance between the objectives established and their fulfillment, weighted by the priorities given. More intricate alternatives (Konak et al., 2006) comprise calculating the set of solutions that are non-dominated[5] with respect to each other (Pareto optimal set). The Pareto front translates the cost of edging one objective as a function of the prices paid in terms of the remaining objectives. The trade-off capability is generally preferred by practitioners given that it resembles more closely the reality of decision making. However, computational requirements escalate and the interpretability of the Pareto front becomes fuzzy as the number of objectives increases.

Savage's regret criterion

The aforementioned optimization techniques assume that consequences of alternative decisions are known within a reasonable degree of certainty. When taking decisions over uncertain outcomes, one enters the domain of Decision Theory (Hillier et al., 2004). An important notion within Decision Theory is the state of nature: a stochastic occurrence that will determine the state found once the decision is taken.

Savage's rationale to tackle decisions under uncertainty proposes not to take the route that leads to the best outcome regardless of the risk, but rather chose the path that leads to the minimal opportunity loss. This entails the need for computing a regret matrix, where the maximum opportunity costs associated with each alternative decision path are mapped, calculated relative to each state of nature observed. The decision is then taken in the direction of minimizing the maximum opportunity cost (Minimax). In academia, this approach is classified as moderately pessimistic, since the decision taker assumes that the best outcome from the action will not occur (Ballestero, 2002).

3.6.2 Monte Carlo simulation

Deterministic mathematical models might have some degree of fidelity when representing reality. Nevertheless, they rest on the assumption that there is no variability affecting the inputs in previously not-accounted-for ways. Inputs of realistic models, however, are impacted by external factors that introduced variability in the outcomes. Highly unpredictable environments might, thus, require simulation exercises.

Monte Carlo simulation (Raychaudhuri, 2008) tries to account for stochastic events through repeated random sampling of each of the inputs following statistical distributions. These sources of variability are propagated to the outcomes. To create robustness, several runs are sequentially performed and then the outcomes are scrutinized.

[5] a feasible solution is non-dominated when the are no another feasible solutions that are better than the current one in some objective function without worsening other objectives

Chapter 4

Problem description

To enrich the thesis that there is value in taking an holistic view of prediction and prescription, a case study in a telecommunications company performing upsell telemarketing campaigns was devised. The goal of the ongoing chapter is to first characterize the current operating conditions of the outbound telemarketing (section 4.1), initially at a high-level and with general considerations, and drilling down afterwards into the specific details of the operation run by the aforementioned company (section 4.2). Then, this chapter evolves into explaining the objectives that the management team desires to pursue, from which the as-is methodology is out-of-sync (subsections 4.2.1 and 4.2.2). Afterwards, the challenging nature of the problem is examined (section 4.3) and improvement opportunities are catalogued (section 4.4).

4.1 Current state of outbound operations

Traditionally, improvements to the telemarketing outbound operations have been focused on the predictive properties of an automatic dialing machine (named the dialer from now on) to keep workers as busy as possible, while complying with a set of business rules and legal restrictions. In these exercises, queueing theory and simulation are combined with the objective of anticipating new call answers and schedule them just after an operator becomes available (Samuelson, 1999).

The business rules referenced can be seen as a set of best practices - often called hygienic rules - defined by the CRM teams with the hope of not degrading CLTV due to the intrusive nature of the outbound contacts. Under this light, Grig (2005) alludes to the shrinkage of public's tolerance towards telemarketing due to years of irresponsible contact policies. Typically, the restrictions mentioned act on the customer account level, but they can dig deeper, reaching the individual customer contact information. Furthermore, they generally act by imposing a limit on the number of attempts made and that limit is a function of the outcome of the call: rejected calls have a ceiling lower than that of machine answered ones. Table 4.1 compiles the common restrictions found.

The predictive behavior of the dialer entails that some calls will be fired prior to the capacity to take them is freed. Thus, there is a chance a customer will answer a telemarketing call only to be met by a silent interlocutor. Silent calls are taken very harshly by respondents, reason why contemporary regulations enforce a maximum of 3% silent calls and no more than one per phone number targeted, per day (Ofcom, 2008).

Table 4.1: Hygienic Rules (not-exhaustive)

Restriction	Action level	Restricted on
Maximum attempts to a customer account	CA	Attempts
Maximum silent calls to a specific phone number	CN	Attempts
Maximum rejected calls to a specific phone number	CN	Attempts
Minimum time period between two consecutive attempts	CA	Time-slot
...	...	...

where CA is the customer account, and CN is a specific number within a customer account

Current procedures do not go much further than exploring the predictive behavior of the dialing software to keep the operators on low idle time. A customer-centric view is completely disregarded, as customers are treated like an homogeneous pool of individuals for whom the solicitation features (timing and contact chosen) are irrelevant. Kolar (2006) concluded that taking a systems' view and handling outbound call centers like production grounds, where throughput is all that matters, is large part of the reason why consumer's express distrust on telemarketing.

4.2 The specific operation at hand

The company subject of the case study is a major European telecommunications service provider, with millions of revenue generating units, a considerable portion of which are enrolled in loyalty contracts. The outbound channel represents about 35% of all sales generated and a large potion of those relate to upsell campaigns. Having described generally the governing thought behind the use of predictive dialers, it now calls upon a more in-depth look at the specific operation at hand.

A starting pre-defined set of customers to address (a **batch**) is fed to the dialer in the form of a randomly ordered list. The allocation of a customer to a campaign is the responsibility of CRM teams and it is outside the scope of the current project. Complying with the hygienic rules, the customer list is followed sequentially, with calls being fired whenever the automatic machine finds suitable. From the set of calls attempted, only a small portion (around one fifth) converts into **answered calls**. The remaining either end up in answering machines, are rejected or try to reach a discontinued number. From the answered calls pool, a good portion is assigned to an available operator while the reminiscent constitutes silent calls.

Narrowing on the calls transferred to operators, only a selective subset of respondents (about 40%) shows itself available to hear the sale pitch. The other portion was either called at an inopportune moment and asks for a reschedule, flat out denies any attempt of telemarketing or has been mistargeted by campaign managers and fails to meet the criteria of that specific campaign. In short, less then a tenth of all calls performed reach a point where the message has been successfully conveyed to the customer and he/she is asked to accept or deny the offer. That state of meaningful conversation is referred to as a **useful contact**. The outcome of a useful contact is threefold: acceptance is granted and the sale is closed (**sale**); the offer is denied, but the customer shows openness to other products or services and he is, thus, transferred to agents waiting in a second line[1] (**transfer**); the sale is denied and the customer refuses to be transferred to the second line (**refusal**). A visual aid of the narrowing effect from attempted call to sale is provide in figure 4.1.

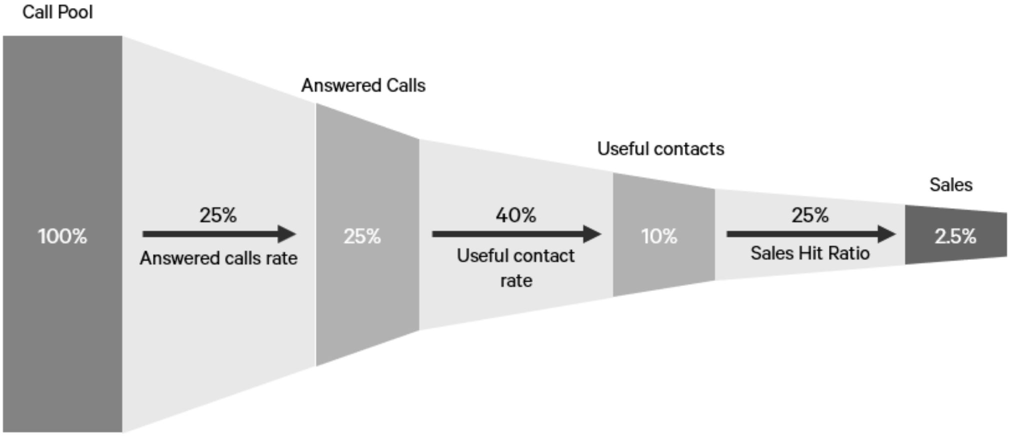


Figure 4.1: Narrowing effect acting on the call chain (not at scale)

The hygienic rules further refrain the dialer from unceasingly firing calls to the same account by applying a quarantine rule: if a previous call was made, but did not convert in a useful contact, then the account enters a quarantine mode, within which no solicitations are allowed. Figure 4.2 shows a condensed view of the the intricate flow of the dialing operation, described above.

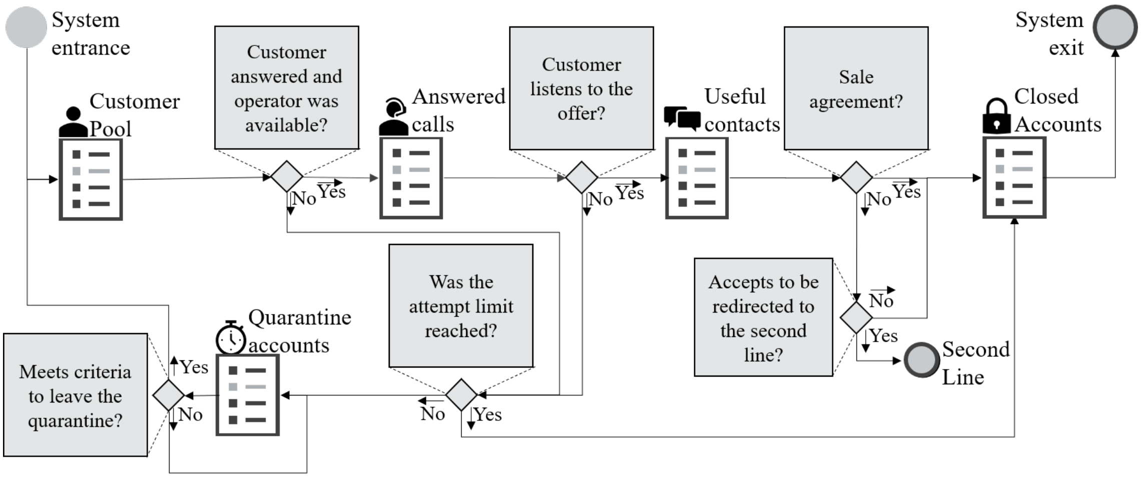

System entrance
Customer Pool
Customer answered and operator was available?
No
Yes
Answered calls
Customer listens to the offer?
No
Yes
Useful contacts
Sale agreement?
No
Yes
System exit
Closed Accounts
Accepts to be redirected to the second line?
No
Yes
Second Line
Was the attempt limit reached?
No
Yes
Quarantine accounts
Meets criteria to leave the quarantine?
Yes
No

Figure 4.2: Flow of outbound operations

[1] the second line is dedicated to selling products/services with low value added and it is outside the current scope of the project

4.2.1 Customers: a scarce resource

When dealing with cross or upsell campaigns the pool of available customers that qualify for a given offer is limited. Hence, the focus is on converting as much clientele as possible from the available set. Having a contact policy that generates sales quickly, but burns through a heavy portion of the batch to do so, is unsustainable in the long run. For that reason, current best practices in the outbound operation imply that small batches are created and introduced periodically. Within each batch, customers are assigned a state: **open** - customers for whom no offer was formally made and have yet to break any of the customer account level restrictions - and **closed** - if any of the former conditions is true. Customers from previous batches (legacy batches) that are still in an open state when a new batch is launched, remain visible to the predictive dialer, fighting for position in the randomly sorted lists. As an enforcement, reschedules always have priority over regular contacts, since their sale conversion rate is much higher than that of regular calls.

4.2.2 The shortcomings of the as-is situation

The far-from-optimal operating conditions of the current methodology for assigning calls are a concern within the organization. The high levels of rescheduling requests[2] raised a red flag in the eyes of the management teams. There is a general worry that contacting a customer at an inopportune moment might harm the conversion chance of not only that call, but all subsequent calls to the same individual - effect known as **response decay**. This line of thought implies that, at an aggregate level, the degree of exploitation of the batch, that is, the total number of clients reached usefully over the total batch size, becomes far from ideal.

Within each customer account, the effect of missing the timing is exacerbated by failing to identify the ideal contact to ring. This compounded mistargeting effect is believed to erode customer loyalty, which materializes in a loss in CLTV and a greater share of customers who subdue to prospecting initiatives from competitors.

4.3 Additional challenges

In addition to all the operationally intricacies and to the specificity of the objective being chased, there is the difficulty adjacent to the fact that the company runs the outbound calls through several service providers, each with their own particularities in operating and reporting results. Thus, the solution chased must allow for some level of customization, all the while providing a general framework that can be applied to all campaigns and service providers.

The premise presented in section 2.3, that building an optimization heuristic on top of predicted outcomes is seldom applied, helps back the claim that no off-the-shelf solutions are available. Moreover, the challenging nature of the problem is uplifted by previously failed pilot runs of projects chasing the same purpose on the same company.

[2] historically, around 10% of all respondents request a reschedule

4.4 Improvement opportunities

Despite having heaps of data about their customers, ranging from the complete history of outbound/inbound calls, to measures taken directly from interactions with equipments stored in households, the company subjected to the case study does not leverage this information towards personalizing communications. A purely arbitrary way of conducting campaigns, like the one in place, does not suffice in today's competitive direct marketing environment.

The predictive capacity of machine learning algorithms fed with such rich data is expected to yield a significant lift in the answered calls rate. Additionally, there is a sound belief that contacting clients at an opportune moment and to the phone number more prone to answer is somewhat correlated with a higher availability to hear and evaluate the offer. In short, by improving the answered calls rate, one expects a positive spillover to the useful contact rate. In absolute terms, that indicates that, even if the sales conversion rate is kept, or slightly decreases, the sheer volume of useful contacts generated will ensure a boost in sales.

The methodology that unfolds in the subsequent chapter has the answered call as the modelling object. By pursuing willingness to answer, all the subsequent states - useful contact and sale agreement - are expected to follow the same direction.

Switching the focus to the operation itself, if the positive spillover between willingness to answer and willingness to listen and assess the offer is verified, then the average call duration will increase. Having a higher number of respondents that stay on the line, on average, for longer will put mounting pressure on the operators, further reducing their idle-time.

Moreover, there is one unexplored axle of improvement, hinted by the previously failed projects. One of the failure reasons concerned the methodology of contacting, for each time slot, the customers more prone to answer. This greedy approach meant that the same group of customers - that were more willing to answer in general - were hammered by requests, many of them at times that did not match their personal preference. In addition, individuals that had poorer response rates were all grouped together and contacted at a later point of the batch life, which meant that the answer rates dropped significantly with time.

In short, the methodology proposed in the following chapter exploits the two main axis of improvement pinpointed so far. An initial stage is devoted to incorporating the richness of customer information available in the company's database into a solid predictive model for assessing the right timing for contact. A second stage, of a more prescriptive nature, builds on top of those predictions to decide, in a not-so-greedy approach, which customers to contact within each time-slot.

Chapter 5

Solution approach

5.1 Methodology overview

The premise of the thesis builds on the idea that prediction and prescription should be thought of together, in a holistic view. Good prediction engines sometimes fail to impact business decisions given that they lack a prescriptive nature. Prescribing tools are often built on top of flawed input, ultimately giving poor advice on what the next move should be.

Figure 5.1 illustrates, at a very high level, the methodology advocated to fight those two analytical shortcomings. At first, the business goal is identified and the key inputs for a prescription engine are mapped. Those key inputs are then chased through some knowledge extraction framework, like the Knowledge Discovery in Databases (KDD) overviewed in Fayyad et al. (1996). After reaching quality predictions, with the aid of state-of-the-art machine learning algorithms, the prescription tools come into play. Those take the shape of simulation engines, heuristics or exact algorithms, and they supply the advisory nature that predictions alone lack.

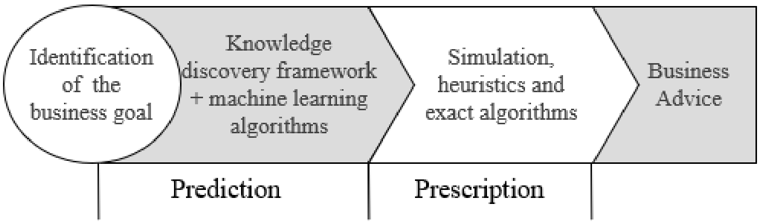

Identification of the business goal
Knowledge discovery framework + machine learning algorithms
Simulation, heuristics and exact algorithms
Business Advice
Prediction
Prescription

Figure 5.1: High-level blueprint of the methodology

The approach to the case study at hand can be seen as a practical application of this high-level framework. As stated in section 4.4, the solution proposed sits on two cornerstones: the first regards the inclusion of the available data about the individual consumer to develop response models that accurately predict the best time to contact; the second concerns an heuristic that can arrange these probabilistic outputs of prediction models into a ranked list, aiming at maximizing the cumulative response probability, over a limited number of attempts, while ensuring that the capacity of the call center is fully exploited. Sections 5.2 and 5.3 describe in-depth both cornerstones and how the first link intertwines with the second.

5.2 Prediction

The prediction endeavor followed the Knowledge Discovery in Databases (KDD) framework. With it, several sequential strides of an ever growing complexity are taken towards achieving the goal of extracting valuable information from stored data.

Data selection and consolidation

The first step to take in the direction of accurately inferring the willingness to answer a contact is in identifying the drivers of the answering behavior. That behavior will, undoubtedly, be affected by the characteristics of the customer addressed. Under this umbrella, one finds the length of relationship, general demographic features, the competitive profile and the proximity to contract expiration date as possible explanatory variables. Besides, past interactions with outbound calls might provide a window for predicting the future behavior. Stating, for instance, that a customer that never had successful interactions before will, most likely, decline the next attempt is a compelling assumption. Moreover, the propensity to answer might be explained by the degree of activity demonstrated by the customer in his/hers interactions with all equipments related to his account. The belief, in this case, is that if a customer usually zapps through his/hers cable TV subscription at a particular time of day, then he/she is likely to be available to hear a sale pitch or he/she will be more tempted to answer the land phone rather than the mobile number, at that time.

The first step of variable identification led to a subsequent move of data gathering, coming from the three main sources identified: dialer logs, customer profile and interaction "metadata" records. The lengthy list of all predictors considered is provided in appendix C. Albeit available, the information was dispersed through several sources and spanned multiple database systems. Data gathering and consolidation was, thus, a first and tumultuous step to ensure quality predictions. The effort, however, proved fruitful, as the consolidated data was dense and ended up requiring only minor data cleaning efforts.

Feature engineering and pre-processing

The density and quality of the consolidated information, however, did not exempt some data transformation efforts, mainly in constructing new features based on the available information. The step taken was with the aim of better capturing the intricacies of human response towards the unsolicited contacts. The feature engineering phase was paramount in building the set of variables that mapped the complete customer response history. Taking inspiration in the RFM analysis (see Birant (2011), for further details on this model), these variables included time elapsed since the last contacts and past outcomes[1], the frequency of contacts made, the relative frequency of each

[1] outcomes include: silent call, machine answered, invalid number, handled call, among others

outcome, the average call duration, just to name a few (consult table C.2 in appendix C for a more extensive overview).

In a preventive effort to mitigate the effects of concept drift, the features were calculated using a rolling window approach. In general, two time windows were used to capture the behavior: within 30 days or within one year of the record date (figure 5.2). The latter intended to capture the general conduct of the customer, while the former seized recent behavioral changes. When determining, for example, the number of telemarketing attempts historically done to one phone number, the records gathered from the dialer logs did not date back more than one year. The rolling window, however, raised some sporadic null values within the dataset. Thus, most data cleaning efforts were focused on performing minor imputations.

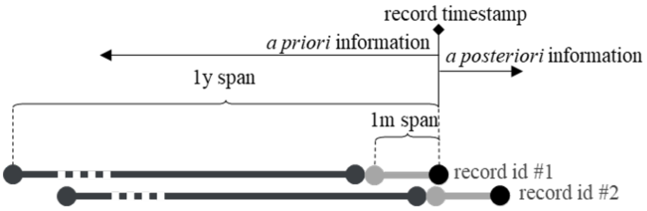


Figure 5.2: Rolling window applied to feature computation

Feature selection

The assumption that all the predictors considered (appendix C) were, firstly, relevant and, secondly, non-redundant had to be determined with certainty. The Boruta Algorithm (subsection 3.1.2) was the selected tool to assist this determination.

The iterative nature of the BA makes it computationally demanding. Thus, a dimensionality reduction is required to ensure results are obtained within a reasonable time frame. The importance of predictors is tested against the shadow feature introduced, through a two-sided test of equality. Given that a stopping criteria on the maximum number of iterations is considered, it is likely that a statistical decision might not be reached for all predictors. As such, their inclusion will rely on a second criteria: those variables that show a median z-score importance, over all iterations, higher than that of the most important shadow feature will be deemed important, with the rest being dismissed.

Training

As mentioned before, the modeling object was the answered call, mapped in the dataset through a binary variable. Although the target was identified, the granularity of the prediction had to be determined. Since within each customer account there are several affiliated phone numbers, often with very diverse response performances, the decision was settled on predicting the response behavior of a specific phone number of a given account, at a particular hour of the day. Furthermore, the problem was handled as a binary classification one, rather than multinomial. A multinomial formulation, where one would predict the hour of the day a specific contact is be more willing to respond, would yield predictions of how likely is that specific time frame to be the best for that particular number. That differs from the desired output, which is to predict, as assertively as possible, the probability that a fired call will be answered. The latter option allows for a meaningful comparison between multiples records' probabilities, the former does not.

A third decision to be made, prior to the full machine learning implementation begins, concerns the spectrum of predictions the model will be asked to forecast. To clarify, there were several starting options:

- Train a single model that would forecast the dependent variable across every hour of the day;
- Train 12[b] models, each predicting the dependent variable for an hour of the day, with visibility limited to training observations of that particular hour;
- Train 12 models, relaxing the visibility constraint, but evaluating the performance only over the predictions of a certain hour of the day.

Since training 12 models instead of one is more computationally demanding, that route would only be taken if a significant boost in performance would be observed. To determine the best path, a preliminary assessment was constructed by training gradient boosting machines (GBMs) over data within the same time-window and with the specifications aforementioned.

Algorithmic selection

Upon beginning the data mining quest, there is no suitable approach for all circumstances. The algorithm selection is no exception to this rule. Although there are a few guidelines governing what is to be expected from the training exercises, there is no certainty that given algorithm will outperform another, as that is case dependent. Once again, there's the need for an empirical benchmark of all the algorithms considered.

To ensure variety in the mix, four algorithms were equated: two of which were single models (logistic regression and artificial neural network), one bagging ensemble (random forest) and one boosting ensemble (gradient boosting machines). Besides, an AUC maximizing stacked learner, with contributions of all four, was trained. The benchmark was made over a sample of the data (300K observations), using a 5 fold cross validation. All algorithms were implemented through the R interface of H2O's machine learning platform written in Java (H2O, 2018) and they were allowed to perform a hyperparameter optimization through random grid search over 16 possibilities. Table 5.1 summarizes all the hyperparameters reasoned and appendix D expands that information to contemplate the alternatives considered, along with a short explanation of each hyperparameter's purpose.

Evaluation and tuning

A careful evaluation procedure is critical when there's the need to perform the comparisons and selections between models and when assessing what is to be expected once the predictive

[b] the outbound operation ran for 12 hours daily

Table 5.1: Hyperparameters considered for search

Model Familiy	Parameters tuned
RF	number of trees, number of bins, max depth, column sample rate
GBM	number of trees, learn rate, max depth, sample rate, column sample rate
Logistic Regression	alpha and lambda
ANN	activation function, hidden layers' configuration, epochs, l1, l2

engine is applied in a practical setting. To that end, the sequence of training and evaluation must reproduce, as closely as possible, the reality found once a model goes live. To meet this goal, several measures were taken:

- A train/test time-window approach was used. With data available since the January 2017, the training/validation partition was made for observations between January 2018 and March 2018 (inclusive). Records from the month of April 2018 were used as a test set. This partition is illustrated in figure 5.3;
- The computation of the different variables disregarded observations less than one day old, since it is unrealistic to conceive that the model would have a streaming[3] behavior. A daily refresh rate is the best that can be expected;
- Following remarks made by Schuller (2018) on how to properly engineer cross validation, observations of the same batch[4] were forced to be grouped together in the same validation fold (figure 5.4). This consideration avoids having two observations of the same customer, on the same day, in the training and validation sets simultaneously and provides a more realistic estimate of the model performance when forecasting for a new batch.

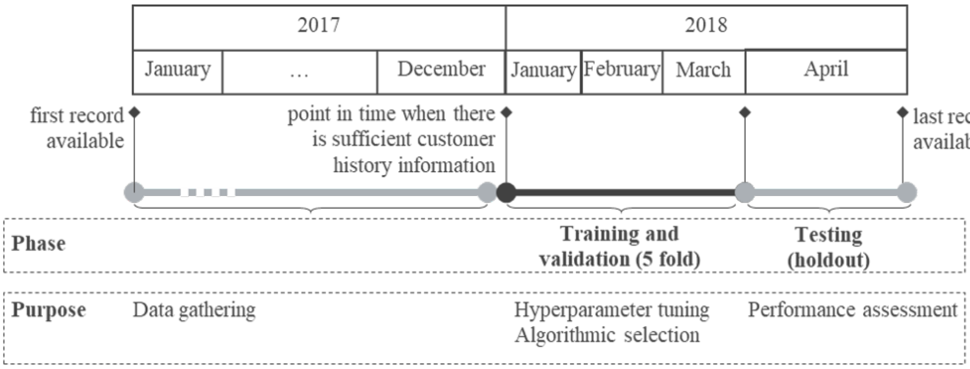


Figure 5.3: Training, validation and testing time windows

An additional concern, advocated by Chawla (2009b) and referenced in section 3.1, is that the performance metric needs to be aligned with the model's purpose. Since the desired output is probabilistic, the Brier score guided the hyperparameter optimization. For selecting the best performing algorithm, although the Brier score had prevalence, other metrics like AUC, F_2 measure,

[3] streaming relates to receiving and processing data in real-time

[4] recall that a batch comprises the set of customer accounts assigned simultaneously for an outbound campaign

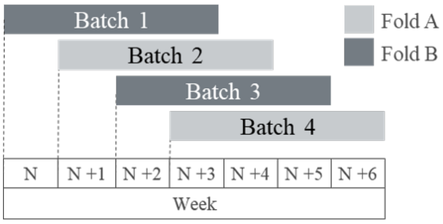


Figure 5.4: Illustrative example of fold allocation. Observations belonging to the same batch are always assigned to the same fold. Each fold is composed of several batches.

bias and lift index were looked at to validate the decision made.

Notwithstanding, at every decision point, the combination picked is the best performing one. The end result is that the cross validated performance estimates become biased towards an optimistic view of the model's real performance. To correct for this effect, the final evaluation was made over the test set for each of the 12 picked models.

Scaling predictions and interpreting the output

A critique often raised against tree-based boosting methods is that they apply a sigmoid distortion to probabilistic outputs. Thus, the methodology accounted for the correction of such behavior through Platt scalling (section 3.4). A calibration set, distinct of the training and test sets, would be used to scale the outcomes of the models, improving the look of the reliability plot[5], while, hopefully, generating better Brier score results. If that was not the case, that option would be shelved.

Besides, machine learning models' output, specially that generated from ensembles like GBM's, have a low grade of interpretability. Operational teams, particularly those without a strong quantitative vein, raise concerns when they are called to act over predictions that lack an intelligible backing. As such, to raise credibility in the models' output, a sensitivity analysis was conducted over the most important predictors and the LIME framework was used to provide explanations over a selective set of observations (section 3.5).

5.3 Prescription

Up until this point, we have been looking on how to generate reliable predictions. The probabilistic output, however, is in itself deprived of an advisory nature. For that, there's the need to include it as the input to a prescription engine. The section that unfolds is devoted to describing the methodology conceived to address the objectives that the studied company desires to pursue. Prior to presenting the solution approach, it might be helpful to introduce the problem through a mathematical formulation.

[5] obtained by firstly binning the probabilistic outputs and then plotting the predicted probabilities within each bucket against the average real response rate. A calibrated output should follow the diagonal as closely as possible

5.3.1 Formulation

Let $\mathscr{C}$ bet the set of customer accounts (CAs) available at any given time for telemarketing attempts and $\mathscr{J}_i$ the set of distinct individual phone numbers affiliated with each account (from now referenced as customer numbers, CNs). Besides, let $\mathscr{L}$ be the set of possible time-slots for contacting, typically the different hours of the day. Let's further define: Y_{ijk} as the binary decision of firing a call to CN j of CA i during time-slot k; p_{ijk} as the probability of having a response to the call; d_{ijk} as the duration of said call and X^m_{ijk} as the flag that indicates whether outcome $m \in \mathscr{M} \cup \mathscr{N}$ occurred for that call. Treating the useful contact as a special type of outcome, let's define U_{ijk} as the flag that maps it. Moreover, some outcomes have imposed maximums. Since those imposed maximums can be indexed to the CA or to the CN, let's define $\mathscr{M}$ as the set of outcomes subject to customer account restrictions and $\mathscr{N}$ as the set of outcomes subject to individual contact restrictions. Although $\mathscr{M}$ and $\mathscr{N}$ have many intersecting elements, they do not match precisely. A rejected call is an outcome subject to CN-level restrictions (and, thus, belongs to $\mathscr{N}$), but has no global customer account limit (and, thus, is not included in $\mathscr{M}$).

Sets :

$\mathscr{C}$ the set of customer accounts

$\mathscr{J}_i$ the set of contacts within each customer account i

$\mathscr{L}$ the set of possible time-slots for contacting

$\mathscr{M}$ the set of outcomes subject to CA restrictions

$\mathscr{N}$ the set of outcomes subject to CN restrictions

Parameters :

a^m number of maximum requests allowed, at the CA level, with outcome $m \in \mathscr{M}$

b^n number of maximum requests allowed, at the CN, with outcome $n \in \mathscr{N}$

q quarantine span

s^+ silent call factor allowed per time slot

C_k the capacity available during time slot k

$$X^m_{ijk} = \begin{cases} 1 & \text{if the call made to customer } i\text{, contact } j\text{, on time slot } k \text{ had outcome } m, \\ 0 & \text{otherwise.} \end{cases}$$

$$U_{ijk} = \begin{cases} 1 & \text{if the call made to customer } i\text{, contact } j\text{, on time slot } k \text{ was useful,} \\ 0 & \text{otherwise.} \end{cases}$$

d_{ijk} the duration of said call

Variables :

$$Y_{ijk} = \begin{cases} 1 & \text{if customer } i \text{ was called through contact } j \text{ on time slot } k, \\ 0 & \text{otherwise.} \end{cases}$$

From an optimization perspective the problem emerges as multi-objective (equation 5.1). For once, there's the desired to maximize the cumulative response probabilities for all the calls made (equation 5.1b), in an attempt to extract as much value from the batch as possible. Then, there is interest in maintaining operators busy, ensuring that the positive difference, in time units, between capacity on demand and the duration of made calls, over the same time slots, is minimized (equation 5.1a). The latter objective is, by company policy, mandatory and should, thus, be given higher priority.

$$\text{Minimize} \sum_{k \in \mathscr{L}} g(k) \tag{5.1a}$$

$$\text{Maximize} \sum_{i \in \mathscr{C}} \sum_{j \in \mathscr{J}_i} \sum_{k \in \mathscr{L}} p_{ijk} \cdot Y_{ijk} \tag{5.1b}$$

$$\text{where} \quad g(k) = \begin{cases} C_k - \sum_i \sum_j (d_{ijk} \cdot Y_{ijk}) & \text{if } C_k - \sum_i \sum_j (d_{ijk} \cdot Y_{ijk}) \geq 0, \\ 0 & \text{otherwise.} \end{cases} \quad ,\forall k \in \mathscr{L}$$

The set of restrictions follows company's best practices: each outcome $m \in \mathscr{M}$, subject to account level restrictions, must not take place more than a^m times for each customer account (equation 5.2) and each phone number of the account cannot be exploited more b^n times with the same outcome n (equation 5.3). Both a^m and b^n are parameters applied indistinctly for all customers. Besides, if a call is made ($Y_{ijk} = 1$), indifferently of the outcome, the account enters a quarantine mode within which no solicitations are allowed and that spans between $]k, k+q[$, with k being the timing of the failed attempt and q the length of the quarantine (equation 5.4). Restriction 5.5 does not allow for the duration of all calls made during time-slot k to overshoot the capacity available by more than the allowed silent-call factor (s^+). Finally, condition 5.6 indicates that a useful contact requires, firstly, a made call, while restriction 5.7 ensures that an account evolves to the closed state once a useful contact is established.

$$\sum_{j \in \mathscr{J}_i} \sum_{k \in \mathscr{L}} (Y_{ijk} \cdot X_{ijk}^m) \leq a^m \qquad ,\forall i \in \mathscr{C}, m \in \mathscr{M} \tag{5.2}$$

$$\sum_{k \in \mathscr{L}} (Y_{ijk} \cdot X_{ijk}^n) \leq b^n \qquad ,\forall i \in \mathscr{C}, j \in \mathscr{J}_i, n \in \mathscr{N} \tag{5.3}$$

$$\sum_{j \in \mathscr{J}_i} \sum_{k \in \mathscr{L}} (Y_{ijk} + Y_{ijk+1} + ... + Y_{ijk+q}) \leq 1 \qquad ,\forall i \in \mathscr{C} \tag{5.4}$$

$$\sum_{i \in \mathscr{C}} \sum_{j \in \mathscr{J}_i} (d_{ijk} \cdot Y_{ijk}) - C_k \leq s^+ \qquad ,\forall k \in \mathscr{L} \tag{5.5}$$

$$U_{ijk} \leq Y_{ijk} \qquad ,\forall i \in \mathscr{C}, j \in \mathscr{J}_i, k \in \mathscr{L} \tag{5.6}$$

$$Y_{ijk} \leq 1 - \sum_{j \in \mathscr{J}_i} \sum_{t \leq k \in \mathscr{L}} U_{ijt} \qquad ,\forall i \in \mathscr{C}, j \in \mathscr{J}_i, k \in \mathscr{L} \tag{5.7}$$

$$Y_{ijk}, U_{ijk}, X_{ijk}^m \in \{0, 1\}$$

5.3.2 The heuristic

The problem at hand cannot be approximated successfully in a deterministic way. Firstly, the nature of several variables is stochastic: the probability to answer, the outcome and the duration of the call are inputs whose real values are not known *a priori*. Besides, the evolution of the system is highly dependent on past outcomes. For instance, a call, within a time-window, will only be fired if the duration of all previously listed calls over the same time frame does not surpass the capacity available and if the account is not in quarantine mode, due to previously failed attempts.

To meet the mandatory objective of guaranteeing an operator idle-time as low as possible (eq. 5.1a), the dialing system must have visibility across all CAs available for contact and, if need be, fire calls to all of them within the same time-window. As such, a sorting heuristic is required and its governing logic must be thoughtful, so that response can be maximized (eq. 5.1b).

In section 4.4, an overly-greedy sorting approach was identified as a root cause that spelled failure for the previous pilot experiments. That knowledge steered the solution towards a rationale of minimizing some materialization of an opportunity cost: a regret.

Taking inspiration in the Savage's formulation of the Minimax criterion, the regret associated with firing a call (R_{ijk}) was modelled as the response probability loss between the hour equated to make the contact (p_{ijk}) and the best call that could be made with that attempt slot ($\max_{h,t}(p_{iht})$), for that day. This view maps the regret as the daily maximum opportunity cost. Objective 5.1b is, thus, translated into the objective function 5.9. Neglecting the capacity limitations and the minimal idle-time objective, the minimization of the regret (eq. 5.9) ensures that all attempt slots are used only for the preferred hours of each customer and, thus, an optimal allocation is reached.

$$R_{ijk} = \max_{h \in \mathscr{J}_i, t \in \mathscr{L}} (p_{iht}) - p_{ijk} \quad , \quad \forall i \in \mathscr{C}, j \in \mathscr{J}_i, k \in \mathscr{L} \tag{5.8}$$

$$\text{Minimize} \sum_{i \in \mathscr{C}} \sum_{j \in \mathscr{J}_i} \sum_{k \in \mathscr{L}} Y_{ijk} \cdot R_{ijk} \tag{5.9}$$

The issue with this approach arises whenever there are conflicting needs during a time slot, and the necessity to select customers arises. In some cases, the predictive model is incapable of discerning a different response behavior across the different hours of the day. The implication is that some CAs, which we will classify as **flat customers**, will have a stable response rate, translated into an opportunity cost close or equal to zero at all times. Resting solely on the opportunity cost to perform an hourly ranking of CAs to address would push flat customers towards the list top across several hours. Besides, the best slot for inelastic[6] CAs would be treated the same way as for elastic ones. With this construction, the sorting heuristic would be biased towards contacting inelastic CAs. Figure 5.5 presents an extreme case of this bias: customer A is clearly elastic, while customer B is flat. For both the maximum opportunity cost is minimal during hour 13. However, the realities of the two are very different: if the opportunity of calling A during hour 13 is missed, the subsequent better options imply taking a big hit to the response rate. For B, those options are nearly as good as the best. Therefore, the two must not be handled equally.

[6] CAs for which changes in the timing of contact have relatively small effect on the propensity to answer

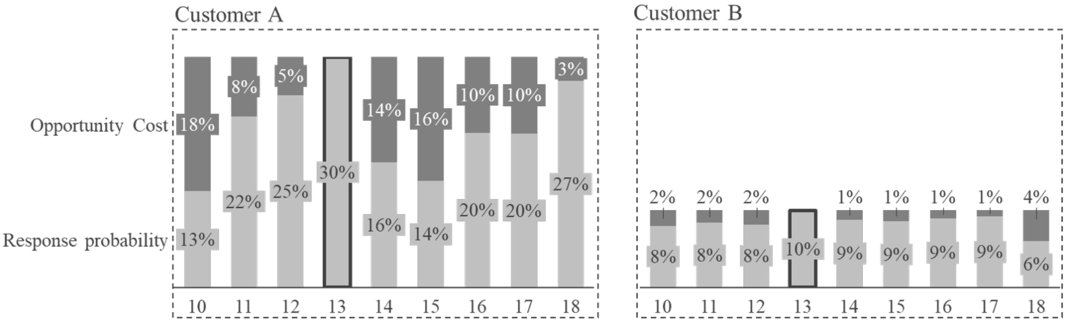


Figure 5.5: The issues of response elasticity in prioritization.

To overcome the elasticity limitation, a conjunction of two principles was used: the previously explained regret would have prevalence, but the descending response probability would be used as a tie breaker. Furthermore, the relative weight given to each criteria would be tweaked by adjusting the number of decimal places that the regret would be rounded to.

5.3.3 Simulation

Despite having a qualitative support, the heuristic lacked quantitative backing. Given the stochastic nature of several inputs, that quantitative analysis required simulation.

The simulation engine constructed was based on the Monte Carlo principle and rested on several assumptions:

- The predicted response probabilities were considered as true probabilities;
- It was assumed that no correlation existed between willingness to answer and willingness to listen, implying that the answered call duration would follow the distribution of call duration found in the dialer's history;
- Operators would be on no idle time, since an unanswered call was modelled as having a null duration;
- The useful contact conversion was presumed to be constant and, thus, independent of the response probability;
- The available capacity of the outbound call center would remain constant throughout the day.

The simple assumptions listed might deviate from reality. Nevertheless, the tool build allowed for a relative comparison of the performance of different sorting principles when subject to the same conditions. Besides the benchmarking capabilities, the simulation provided a way to align expectations about the gains that could be observed once the methodology went live and a way to realize which were the drivers of said gains. The prescriptive role of the simulation was, then, on determining the ideal combination of sorting principles to use.

The following chapter will look in detail into the outcomes of the prediction and simulation, as well as provide an in-depth look at how a field experiment, aimed at assessing the combined performance of the prediction-prescription ensemble, was design and the results it led to.

Chapter 6

Results

The current chapter compiles the results achieved through the application of the methodology described in chapter 5. The first section is dedicated to exhibiting preliminary results from the machine learning and simulation exercises. Section 6.2 discusses the details of the field experiment designed to assess the performance of the prediction-prescription ensemble and section 6.3 elaborates on the results it led to.

6.1 Preliminary results

6.1.1 Prediction

The prediction exercise was comprised of several sequential steps, within which several options are faced against each other. The current subsection is dedicated to showcasing the results that supported the decisions made throughout the knowledge discovery process.

Feature Selection

The application the Boruta Algorithm for feature selection in the configuration referenced in section 5.2, with one tenth of all training observations as the input and a limit of 30 iterations, lead to nearly 90% of the proposed variables being deemed statistically important or unimportant, through a two-sided test of equality, when compared with the shadow features introduced (please refer to figure 6.1). A statistical decision could not be reached for the remaining 10% of predictors. The second criteria of applying a threshold based on the median z-score importance helped achieve a decision over the predictors still in the undecided condition. Ultimately, five variables were excluded (grey patch of figure 6.1), either by being irrelevant or redundant. This move slightly decreased the dimensionality of the problem while also improving the error metrics. The brier-score, for instance, suffered a 3% decrease, hinting that the variables eliminated picked up on mostly noise signals. Again, appendix C extends the information already provided.

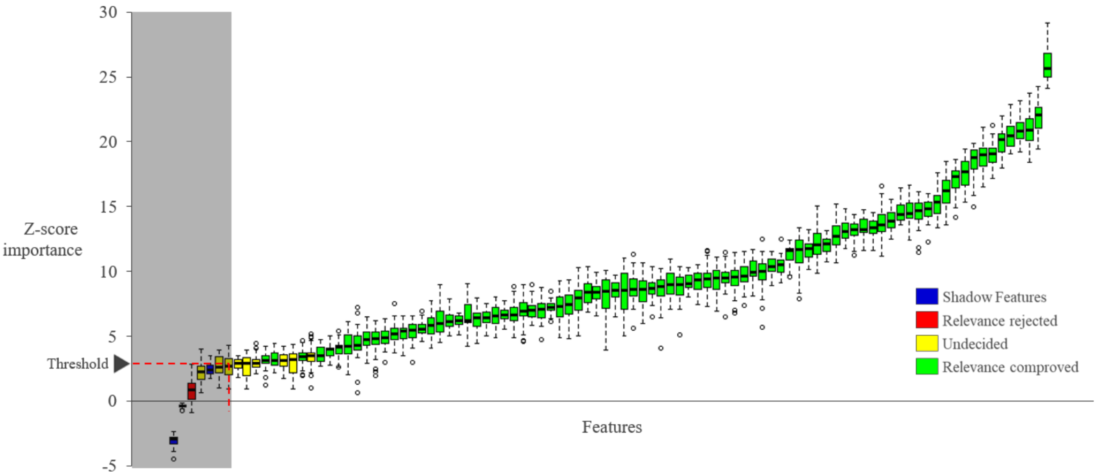


Figure 6.1: Boruta algorithm's output

Algorithmic choice and parameter tuning

In chapter 5, a mention was made over the necessity to perform an algorithmic benchmark, in a similar fashion to the one employed by Niculescu-Mizil and Caruana (2005). Figure 6.2 condenses, in a radar chart, the comparison over six different dimensions of all five algorithms tested. Uncalibrated GBMs (red color coded) yielded a better performance over three of the six metrics evaluated. The ordering metrics (AUC and lift index) indicate that GBMs are exceptional at ranking the cases, independently on where those cases lay[1]. The benefits are also achieved over the predominant metric - the Brier score - which aims at interpreting the quality of the posterior probabilities. The only threshold-dependent metric, the max F_2, also reveals a better performance. Nevertheless, boosting trees present some shortcomings. The higher predictive capacity is paid by in two fronts: the first regarding a high computational time (on average 36 times higher than training logistic regression); the second is related to the sequential nature of boosting that, by iteratively intensifying costs of misclassified observations, appears to introduce a significant positive bias in the predictions. Out of the five algorithms tested, GBM was the only to overestimated the answered call rate, all the others had a negative bias. In short, GBMs are not only more capable to provide quality predictions in their uncalibrated format, but they complement this ability with a higher capacity to rank observations. Thus, the decision was settled on applying GBMs as the algorithm to generate predictions.

It should be mentioned that all algorithms were deployed with the third configuration conjectured in section 5.2, that is, from each algorithm 12 models were equated, one for each hour of the day, all with visibility over the entire dataset, but evaluated only on observations belonging to the hour at stake. The decision was reached after a preliminary run, where that option revealed the be the most adequate. That was to be expected since, with all training observations, it enabled a better grasp of each customer's behavior, while allowing for some specific tuning to capture the

[1] those metrics make no distinction if predictions fall just within]0.1;0.2[, or if they cover a much wider range of]0.01;0.99[

nuances of each hour.

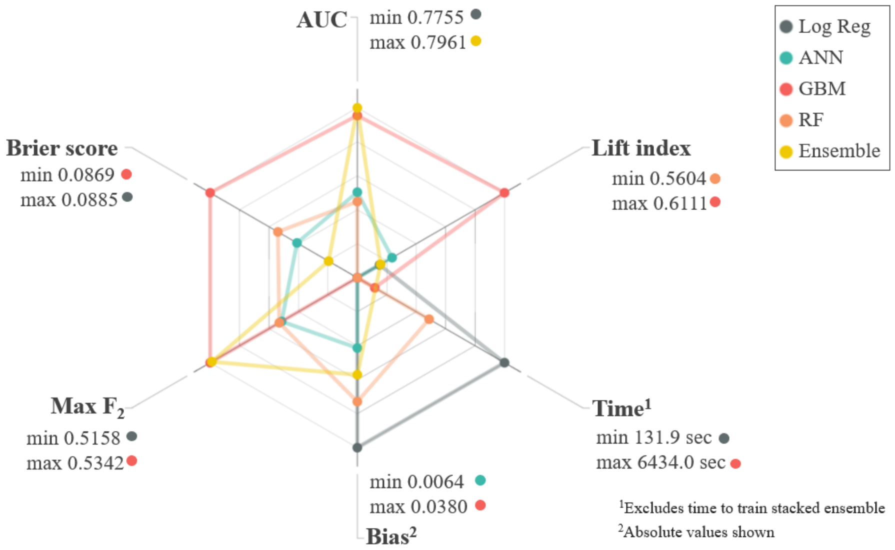


Figure 6.2: Algorithmic benchmark: average of each algorithm's performance over the 12 hour period (cross-fold validated). Each metric suffers a min-max normalization. Outermost points demonstrate desirable qualities of the metric: for the Brier score that is a minimal value, for the AUC that is the maximum.

Another conclusion to be drawn from the benchmark conducted is that good performance on threshold or ordering metrics provides no absolute assurance that the probabilistic metrics will follow the same trend. ANNs have a better score in lift index and AUC when compared to RFs and have very similar behavior over the max F_2. Yet, this reality is mirrored when looking at the Brier score. The same happens to the stacked ensemble. Since the governing heuristic behind its training was AUC-maximizing, unsurprisingly, the stacked ensemble was the top performer for AUC. Still, it showed mediocre results for the brier score.

Regarding the hyperparametrization, since the test conducted yield a total of $16 \cdot 12 = 192$ trials per algorithm considered, and in order to keep the body of this thesis within a reasonable length, the cross validated results are only displayed for four distinct hours of the day (out of the 12 computed) and are compiled in appendix E.

Concerning the calibration exercise, the Platt scalling failed to push the reliability diagram towards the diagonal, as hypothesized in Niculescu-Mizil and Caruana (2005). As figure 6.3 illustrates, the scaling introduced additional errors within probability bins where most observations laid. Hence, the calibrated probabilistic output was more inaccurate, overall, than the uncalibrated predictions. Since the calibration introduced an additional level of complexity, doing so without improving the probabilistic metrics, a decision of not pursuing it became trivial. The results push a conclusion in the opposite direction as the argument raised by Caruana and Niculescu-Mizil (2006), who defend that scaling significantly boost probabilistic measures, particularly for tree-based boosting algorithms. The dipping of both curves beneath the desirable diagonal are indicative of the aforementioned positive bias.

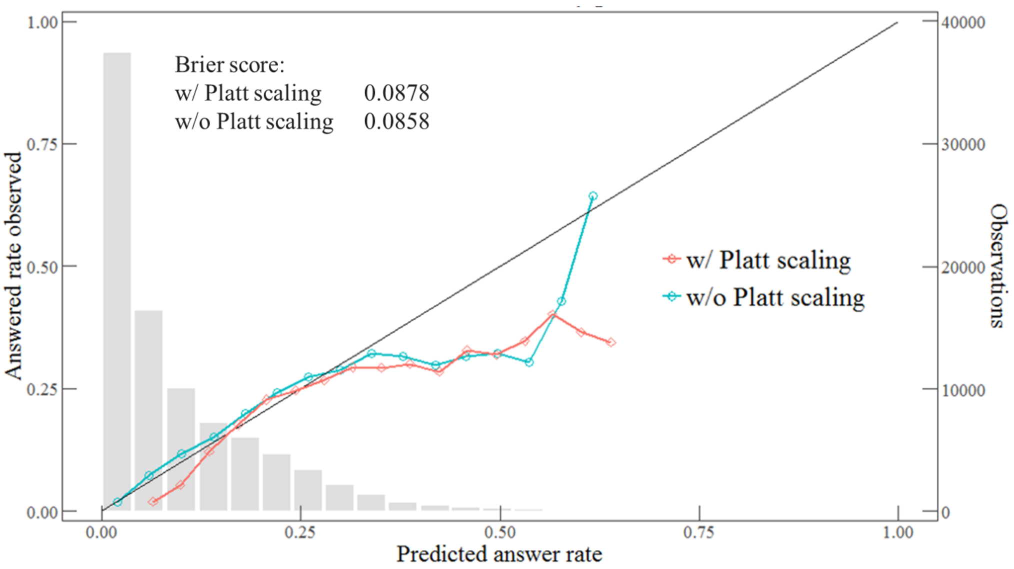


Figure 6.3: Predictive reliability plot (scaled vs non-scaled predictions). The values of both curves are read in the left vertical axis and it is desirable that they follow the diagonal as closely as possible. The histogram shown displays the absolute frequency of observations falling within each of the buckets and should be read on the right vertical axis. Buckets of observations that go beyond the ones presented were excluded since the amount of observations that fell on them was so residual that the evolution of the curves became erratic and noisy.

The evaluation procedure concluded with an assessment of performance metrics over the test set. The cross validated error predictions, conjectured to provide an optimistic view of model performance, demonstrated to converge towards the values extracted from the test evaluation. Table 6.1 encapsulates that comparison.

Table 6.1: Test results.

Estimates	**Brier score**	**AUC**	**Lift Index**	**Max F2**	**Bias**
Cross-validated	0.0869	0.7946	0.6111	0.5342	0.0380
Test set	0.0858	0.7778	0.6317	0.5297	0.0371

Trusting the model

As mentioned, calling for action over predictions made by black-box models can easily lead to apathy in the operational teams since they do not trust the information that they are given. As such, interpretability is certainly a big driver of machine learning applications' success. To build trust in the models developed, three different exercises were undergone. The first, and certainly more traditional, regarded analyzing the variable importance (figure 6.4). Then, a simple sensitivity analysis of the impact of the most important variables was done through a top 10% versus bottom 20% comparison (figure 6.5). Both analysis showed that predictors indexed to the individual contact had far more impact on the outcomes than the ones related to the timing of the contact.

The frequency of answer of a customer number (freq_CN[2]) was far more determinant than the frequency of answer of a particular hour (freq_hour[2]), for instance. The predictive engine was firstly addressing the general behavior of each customer number before narrowing down on intricacies of specific contact timings, which constituted a reasonable and expected behavior.

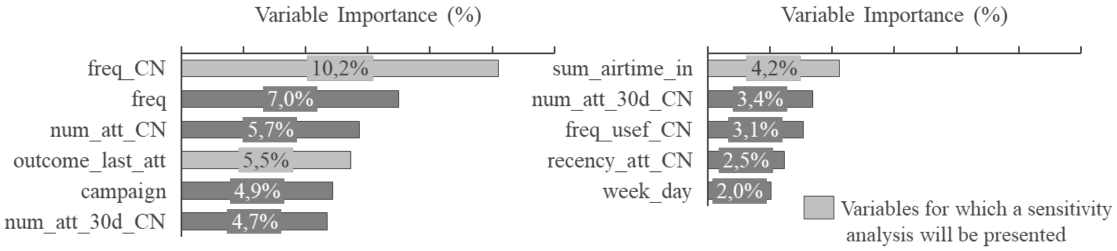


Figure 6.4: Variable importance outcome for the predictors with a percentage importance greater than 2%.

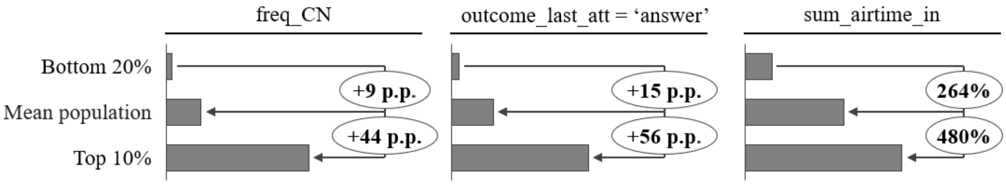


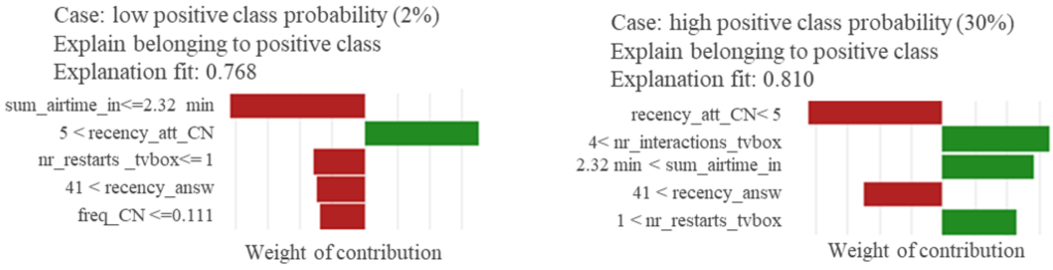
Figure 6.5: Sensitivity analysis (bottom - mean - top) of three top predictors.

Finally, the LIME framework was applied. The premise of LIME is that validating the explanations for predictions within a diverse subset of observations, even if those explanations are merely local, provides a window to the global behavior of the model. Several explanations were scrutinized, but the focus will be on two of them: one with a relatively high[3] response probability (30%), and another whose expected response rate is just 2%. LIME's output on figure 6.6 should be read as follows: the magnitude of the bars indicates the weight that the specific variable has on the prediction made, the color is indicative on whether, globally speaking, the value observed for that feature is supportive of assigning that observation to the positive class, or if it contradicts it. Finally, the explanation fit provides a quantitative evaluation of the explanation achieved by LIME.

For the case presented in 6.6(a), four out of the five most explanatory variables, around the neighborhood of the observation, are rooting against assigning that observation to the positive class. The low average time spent on calls received during that hour (*sum_airtime_in*), along with a residual interaction with the tv box (*nr_restarts_tvbox* ≤ 1) and the low historical frequency of answer of outbound calls for that particular phone number contribute to the almost certain decision of assigning it to the no-answer class. On the other hand, for observation 6.6(b), the higher interaction with the tv box and higher mobile phone activity, for that hour of the day, contribute to the higher predicted probability of response. According to the explanation, the fact

[2] recall that appendix C provides a description of all predictors used

[3] note that a response probability of 30% falls within the 15th highest percentile



(a) Low probability of response

(b) High probability of response[2]

Figure 6.6: LIME's output for two very distinct observations (features' description in appendix C).

that less than five days passed since the last attempt to that contact is preventing the probability from being even higher.

LIME's output proves that the model has a sensible behavior when reaching predictions: it benefits observations of customers that interact more with the devices of their account and have a more active historical outbound behavior. The explanations matched, for the most part, the intuition that operational teams had, helping to generate consensus over the predictive capacity of the model.

6.1.2 Simulation outcomes

A second dimension to control for resides in the expected benefits of the sorting heuristic, built on top of predicted outcomes. The simulation proved that having hygienic rules is critical in ensuring that the sorting heuristic outperforms the random allocation. If no boundary is set, then the sorting applied as little to no effect on the batch's degree of exploitation. This is noticeable in the plateau formed in figure 6.7 and on the convergence of the curves, as the number of attempts allowed per customer rises.

The prescriptive role of the simulation was on determining the ideal combination of sorting principles to use. Taking only the minimum regret would not be ideal due to the flat behavior of some customers. Going for descending response probability alone, as aforementioned, was too greedy. A combination of ascending maximum regret with the descending response probability as a tie breaker proved to be the option that maximized the response. The simulation chased the right balance of the two, ultimately steering the decision towards using the regret rounded at two decimal places.

Using the last configuration, the benefits of the sorting heuristic to the response were expected to fall between 8 and 22 percentual points on top of the current assignment rules. The lower bound is found assuming that the random allocation always contacts the phone number, within the CA, more prone to answer at that time. The upper bound relaxes that condition. The true behavior of the dialer falls between those two bounds: its nature isn't purely arbitrary, given that it assigns priority to the preferred contact within each CA, but it is hardly ever right. Aligned with the conclusions of the variable sensitivity exercise of the previous section, the gap between the lower and upper bounds is indicative that most of the benefit lays in identifying the right contact to ring within each customer account, rather than in recognizing the ideal hour to call.

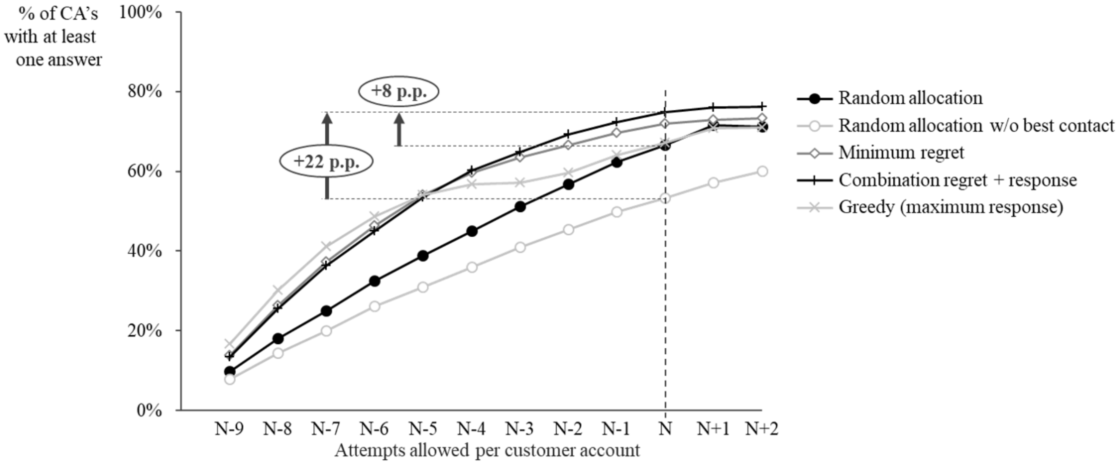


Figure 6.7: Simulation's output: average batch cumulative response, over 20 runs, by sorting heuristic applied, where *N* is the current number of attempts allowed per customer.

6.2 Experimental design

The machine learning model's performance upon deployment is known through measures taken over the test set. The sorting heuristic benefits over the random allocation were approximated through the simulations performed. Yet, the combined performance of the prediction-prescription ensemble was never assessed. To that end, an experiment was designed.

Experiments can be extremely powerful in diagnosing the impact that factors have on the behavior of certain populations. Nevertheless, the assessment made is only as good as the care taken in designing the experiment. As seen, several factors influence the outbound operations' success. However, the study wants to measure the benefit that the prediction coupled with the sorting heuristic have on the response. Therefore, that effect must be isolated from the all remaining controllable inputs.

To that extent, a pilot test was designed. A batch of 7750 customers (the population) was divided equally into two groups: one destined for control, while the other received treatment[4]. Given that the starting population is large, by performing a randomized sampling, the two groups ended up with extremely similar behavioral measures across the board (in appendix F, backing for this claim is presented). This procedure ensures that customer-related factors become irrelevant to the analysis.

Moreover, the company object of the case study showed availability to assign operators with similar contact history performance to handle calls for each of the groups, eliminating yet another source of noise. Besides, all addressed individuals were already customers of the company to

[4] treated observations were the ones subject to the methodology proposed

whom the same upsell offer would be presented. Since both groups would be tested simultaneously, any time dependent sources of noise would be cancelled.

In compliance with all these requirements, there was enough confidence that the methodology proposed in this thesis was the only assignable cause affecting the response behavior of the samples. Therefore, the stage was set for the six day[5] pilot test to begin.

The operation to handle the CAs in the control group remained unaltered. The treated group, however, was exposed to the methodology described in chapter 5. After the splitting, predictions for all combinations of customer account (CA), phone number and hour were generated through GBM models, without scaling. Those predictions were fed to the simulator to infer the magnitude of the sorting criteria to apply. With that information, 12 prioritized lists (one for each hour of the day) were fed daily into the dialing system. The dialer logs, collected daily after the operation closed, enriched the available dataset and affected, with one of delay, the predictions generated. Since no significant concept drift was expected to occur over such a short span of time, the models weren't retrained during the pilot's lifetime.

6.3 Experiment results

The logs collected daily fed a dashboard (appendix G) that showcased several KPIs, meaningful for management, and established comparisons between the progress of the treated group and the control group. Each of the stages of the funnel presented in figure 4.1 were analyzed. Measures were grouped into hourly, daily and weekly granularities. Besides, the evolution of the duration of answered calls, useful contacts and sales contacts was tracked.

Given the concern already mentioned over the progress of the batch consumption, aggregate results were also computed. These included the fraction of CAs with at least one answered call, percentage of customers contacted usefully, total sales achieved in the batch and number of total accounts pushed into the closed state.

Recalling the funnel previously presented, but this time introducing a comparison between the treated and control groups, it becomes evident that, applying the methodology proposed, a larger number of answered calls were generated from 23% less attempts. The useful contact conversion rate was slightly higher for the treated group, edging it ahead of the control by 5% when we compare the volume of useful contacts generated. Furthermore, from the customers who got to the decision point of having to accept or decline the offer, a significantly higher portion of them agreed with the sale. Combining all those effects, the amount of sales generated was 27% higher in the treated group from an initial 23% less attempts. As table 6.2 verifies, all three conversion metrics displayed a behavior statistically better when the methodology was deployed.

Although the focus was on the answered call, which did indeed rise significantly, there was the belief that the willingness to answer would be matched by a willingness to listen. That willingness to listen was conjectured to be materialized in a higher useful contact conversion rate and higher sales hit ratio. That suspicion was proven to be truthful in this analysis. The explanation

[5] from monday 2 p.m to saturday 2 p.m

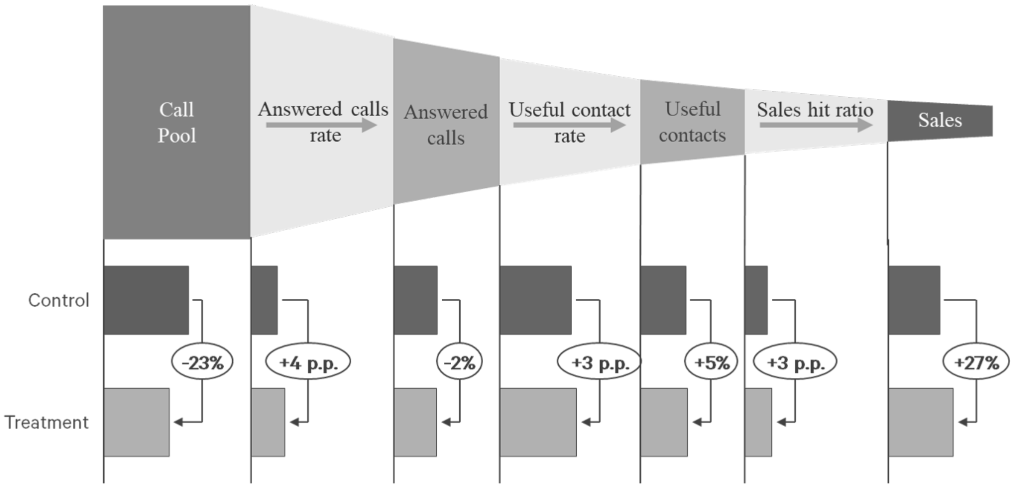
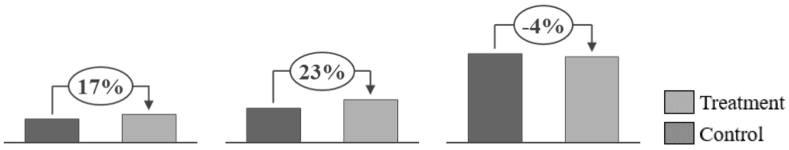


Figure 6.8: Pilot's results: The blocks "Call Attempts", "Answered Calls", "Useful contacts" and "Sales" are all in absolute values, thus the comparison is made in percentage gains. The remaining blocks represent rates of transformation in percentage and their comparison is made in percentual points.

for a higher sales hit ratio is found in the higher duration of answered calls and useful contacts (figure 6.9). On average, a call answered when the methodology was applied was 17% longer than without it (figure 6.9(a)). That effect was even more noticeable in the useful contact domain, where the increase was 23% (figure 6.9(b)). Customers spent more time on the phone, providing the operators with a larger window for pitching the sale.



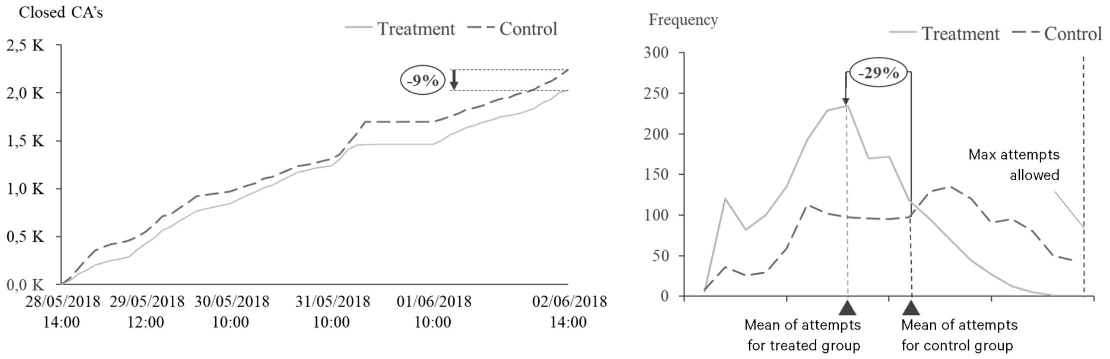
(a) Answered Calls (b) Useful contacts (c) Sales

Figure 6.9: Call duration

The contact effectiveness of the prediction/prescription engine is further demonstrated by the drop of 1.1 percentual points in rescheduling outcomes. Additionally, silent calls raised dramatically, from 0.80 % in the control to 1.26% in the treated group. That is indicative that the dialing software was underestimating the response arrival rate and the call duration that were being achieved with the framework in play.

Table 6.2: Hypothesis test results of proportion comparison (Z-test, Type I error: 5%)

Metric	Confidence interval	Proportion Difference	Z_observed	Z_critical	p-value
Answered Call Rate	]0,0357 ; 0,0493[	0,042	12,301	1,960	< 0,0001
Useful Contact rate	]0,0093 ; 0,0507[	0,030	2,849	1,960	0,004
Sales Hit Ratio	]0,0056 ; 0,0506[	0,028	2,035	1,960	0,014



(a) Closed accounts evolution

(b) Attempts made to open accounts

Figure 6.10: Batch state at the end of the experiment

Still, there is one unexplored dimension. Recall the mention in section 4.4 about the concerns raised over accelerated batch consumption to generate sales. In fact, basing the sorting heuristic on the minimization of an opportunity cost led to a balanced contacting behavior. The harassment that characterized the outbound operation was replaced by a more refrained and data-backed way of contacting. With less attempts made, there were fewer accounts closed due to hygienic rules' violation. At the end of the week, there were 9% less CAs closed in the treated group (figure 6.10(a)). The CAs that remained active in the treated group were further away from the threshold of maximum attempts allowed (figure 6.10(b)).

Chapter 7

Discussion and conclusions

7.1 Practical implications

We will begin by addressing the contribute of the research conducted in this dissertation to the knowledge discovery process in general, and to machine learning exercises in particular. At each stage of the KDD, there were several conjectured options that could be taken. Whenever appropriate, some options were benchmarked.

The algorithmic benchmark performed, albeit biased to the dataset used, steers towards the conclusion that ensemble methods have a higher capacity to capture the underlying truth of the signal than single models. From those, gradient boosting machines stand out as particularly well-suited to handle the unbalanced dataset presented. GBMs were prominent in adequately ranking the observations, while also providing probabilistic outputs that matched closely the real posterior probabilities.

At critical decision points, the research conducted seems to stride away from previous studies (Niculescu-Mizil and Caruana, 2005). Firstly, there is a general conception that tree-based boosting algorithms introduce sigmoid distortions to probabilistic outputs, with linear models and ANNs outshining them in that regard. Yet, the results obtained indicate otherwise. Moreover, an argument can be raised that the use of Platt scaling to improve model performance when chasing probabilistic metrics should be treated as case dependent, and never as a global truth. In the practical test conducted, it unsettled predictions even further than the uncalibrated algorithm.

The philosophy of stacking strong learners into an ensemble capable of superlative performance (Super Learner) complied with what it proposed on paper, but only just. The stacked ensemble, trained to be AUC maximizing, surpassed all other algorithms for that metric. Yet it provided mediocre results for the Brier score, validating the notion that optimizing for some performance metric does not ensure that all others follow. Having failed to come across a super learner aimed at optimizing probabilistic measures, we raise that concern into future work.

Additionally, there is lack of consensus in academia on how to perform the evaluation procedure when testing several algorithms and allowing for hyperparameterization. The route taken in this dissertation has some reasonable backing. Still, it can and should be contested. Furthermore, the hyperparametrization grid search conducted may be skewing the results of the benchmark, as it is restricted to searching within the boundaries defined initially. Alternatives like the adaptive range random grid search optimization strategy, introduced by Schuller (2018), should be considered in future implementations.

On a good note, the Boruta algorithm emerged as an adequate tool, adapted to the current demands of knowledge discovery over massive datasets, for selecting the relevant and non-redundant set of features that explains the response variable. Nevertheless, a comparison between this and other feature selection algorithms like mrMR is still absent and should be pursued.

Moreover, the effort undergone to give a view into the inner workings of the model constitutes a module than can be reproduced in any machine learning exercise. Besides comforting the non-expert with the predictions generated, it helps the model designer validate the decisions made by the algorithm, understanding if the behavior of the model is reasonable even when it misclassifies instances. For this particular application, this stage was key in uniting the interest of all parties involved, fact that later proved critical to the experiment's success.

The prediction-optimization combo

Glancing at the results showcased in the previous chapter, it becomes evident that the targets defined in the introductory motivation were met. Sales increased by 27%, doing so with a reduction on the number of attempts made. As such, both operational teams (guided by sales metrics) and CRM teams (guided by customer satisfaction) rallied around the proposed methodology, eager to perform rollouts for all telemarketing campaigns the company operates. Sustaining the 27% increase in sales would constitute a transformational leap forward for the company. Notwithstanding, the greater benefit arguably spawns from the increased customer satisfaction due to a more thoughtful targeting policy. To capture these mid-to-long term effects, however, a longer pilot run is required.

Regarding the optimization module, given the particularities of the case study, a simple sorting heuristic sufficed to generate significant performance increases. Nevertheless, other applications may be suitable for linear optimization, or require more sophisticated heuristics like evolutionary or insertion heuristics. Still on the case study at hand, other improvement opportunities, congruent with the described system, were identified. Current practices dictate that the capacity of the call center is maintained throughout the day. As seen, capacity constraints play a critical role in the solution found. Thus, a workforce planning exercise, to fine tune the available capacity at different moments throughout the day would be beneficial.

7.2 Closing statement

The work developed should not be seen as a tailor made solution applied to the intricacies of a case study. Alternatively, it should be considered as a proposal of a unified prediction-optimization framework that was validated on a practical setting. This approach, as mentioned in chapter 2, is unlike most academic exercises since it bridges the gap between prediction and prescription, rather than focusing on just one of them. Moreover, the exercise condensed in this dissertation showcases the potential benefits lurking within the myriad of information spread through companies' databases that can, with moderate resource allocation, provide significant boosts to campaign profitability.

Nevertheless, direct marketing is not solely comprised of telemarketing. As such, future steps should be taken in the direction of endorsing the prediction-prescription ensemble with other practical case studies. Direct mailing of individually tailored promotions issued by a retail chain or advisory systems that present the next best offer to convey to a customer once he steps though a retail store, just to name a few, constitute ideal proving grounds to put this framework to the test.

Bibliography

Asare-Frempong, J. and Jayabalan, M. (2017). Predicting customer response to bank direct telemarketing campaign. volume 2017-January, pages 1–4. Institute of Electrical and Electronics Engineers Inc.

Asllani, A. and Halstead, D. (2011). Using rfm data to optimize direct marketing campaigns: A linear programming approach. *Academy of Marketing Studies Journal*, 15(2 SI):59–76.

Asllani, A. and Halstead, D. (2015). A multi-objective optimization approach using the rfm model in direct marketing. *Academy of Marketing Studies Journal*, 19(2):65–80.

Baesens, B., Viaene, S., Van Den Poel, D., Vanthienen, J., and Dedene, G. (2002). Bayesian neural network learning for repeat purchase modelling in direct marketing. *European Journal of Operational Research*, 138(1):191–211.

Ballestero, E. (2002). Strict uncertainty: A criterion for moderately pessimistic decision makers. *Decision Sciences*, 33(1):87–107.

Bergstra, J. and Bengio, Y. (2012). Random search for hyper-parameter optimization. *Journal of Machine Learning Research*, 13(Feb):281–305.

Bickert, J. (1997). Cohorts ii: a new approach to market segmentation. *Journal of Consumer Marketing*, 14(5):362–379.

Birant, D. (2011). *Data mining using RFM analysis*. InTech.

Blattberg, R. C. and Deighton, J. (1991). Interactive marketing: exploiting the age of addressability. *Sloan management review*, 33(1):5.

Blattberg, R. C., Kim, B.-D., and Neslin, S. A. (2008). *Why Database Marketing?*, pages 13–46. Springer New York, New York, NY.

Blattberg, R. C., Malthouse, E. C., and Neslin, S. A. (2009). Customer lifetime value: Empirical generalizations and some conceptual questions. *Journal of Interactive Marketing*, 23(2):157–168.

Bose, I. and Chen, X. (2009). Quantitative models for direct marketing: A review from systems perspective. *European Journal of Operational Research*, 195(1):1–16.

Bravo, C., L'Huillier, G., Lobato, J. L., and Weber, R. (2010). Probability estimation for multiclass problems combining svms and neural networks. *Neural Network World*, 20(4):475.

Breiman, L. (1996). Bagging predictors. *Machine Learning*, 24(2):123–140.

Breiman, L. (2001). Random forests. *Machine Learning*, 45(1):5–32.

Breiman, L., Friedman, J., Stone, C., and Olshen, R. (1984). *Classification and Regression Trees*. Taylor & Francis.

Brier, G. W. (1950). Verification of forecasts expressed in terms of probability. *Monthey Weather Review*, 78(1):1–3.

Bult, J. R. and Wansbeek, T. (1995). Optimal selection for direct mail. *Marketing Science*, 14(4):378–394.

Cao, S., Zhu, Q., and Hou, Z. (2009). Customer segmentation based on a novel hierarchical clustering algorithm. pages 969–973.

Caruana, R. and Niculescu-Mizil, A. (2006). An empirical comparison of supervised learning algorithms. In *Proceedings of the 23rd international conference on Machine learning*, pages 161–168. ACM.

Cawley, G. C. and Talbot, N. L. (2010). On over-fitting in model selection and subsequent selection bias in performance evaluation. *Journal of Machine Learning Research*, 11(Jul):2079–2107.

Chawla, N. V. (2009a). Mining when classes are imbalanced, rare events matter more, and errors have costs attached. volume 3.

Chawla, N. V. (2009b). Mining when classes are imbalanced, rare events matter more, and errors have costs attached. volume 3, page 753.

Ching, W. K., Ng, M. K., Wong, K. K., and Altman, E. (2004). Customer lifetime value: stochastic optimization approach. *Journal of the Operational Research Society*, 55(8):860–868.

Cieslak, D. A. and Chawla, N. V. (2008). Learning decision trees for unbalanced data. In Daelemans, W., Goethals, B., and Morik, K., editors, *Machine Learning and Knowledge Discovery in Databases*, pages 241–256. Springer Berlin Heidelberg.

Fayyad, U., Piatetsky-Shapiro, G., and Smyth, P. (1996). From data mining to knowledge discovery in databases. *AI magazine*, 17(3):37.

Freitag, C. (2016). Modeling marketing effort in an omni channel world. pages 117–121. Association for Computing Machinery, Inc.

Geurts, P., Ernst, D., and Wehenkel, L. (2006). Extremely randomized trees. *Machine Learning*, 63(1):3–42.

Grig, R. (2005). Outbound calling — discredited or misunderstood? *Journal of Targeting, Measurement and Analysis for Marketing*, 13(4):295–298.

H2O (2018). H2o documentiation site. `http://docs.h2o.ai/h2o/latest-stable/h2o-docs/index.html#` [Accessed: 20th March 2018].

Haughton, D. and Oulabi, S. (1997). Direct marketing modeling with cart and chaid. *Journal of Direct Marketing*, 11(4):42–52.

Hillier, F., Lieberman, G., Hillier, F., and Lieberman, G. (2004). *MP Introduction to Operations Research*, book section 15. McGraw-Hill Science/Engineering/Math.

Javaheri, S. H., Sepehri, M. M., and Teimourpour, B. (2013). *Response Modeling in Direct Marketing. A Data Mining-Based Approach for Target Selection*, pages 153–180. Elsevier Inc.

Jolliffe, I. T. (2017). Probability forecasts with observation error: what should be forecast? *Meteorological Applications*, 24(2):276–278.

Kamandar, M. and Ghassemian, H. (2010). Maximum relevance, minimum redundancy feature extraction for hyperspectral images. In *2010 18th Iranian Conference on Electrical Engineering*, pages 254–259.

Khan, R., Lewis, M., and Singh, V. (2009). Dynamic customer management and the value of one-to-one marketing. *Marketing Science*, 28(6):1063–1079.

Kohavi, R. (1995). A study of cross-validation and bootstrap for accuracy estimation and model selection. In *Ijcai*, volume 14, pages 1137–1145. Montreal, Canada.

Kolar, T. (2006). Evaluating the performance of call centres from consumers'perspective: Marketing research industry example. *Management: journal of contemporary management issues*, 11(2):53–76.

Konak, A., Coit, D. W., and Smith, A. E. (2006). Multi-objective optimization using genetic algorithms: A tutorial. *Reliability Engineering & System Safety*, 91(9):992–1007.

Kuo, R. J. and Chen, S. S. (2017). Intelligent customer segmentation system using hybrid of artificial immune network and particle swarm optimization algorithm. *Applied Mathematics and Information Sciences*, 11(3):877–889.

Kursa, M. B. and Rudnicki, W. R. (2010). Feature selection with the boruta package. *J Stat Softw*, 36(11):1–13.

Laan, M. J., Polley, E. C., and Hubbard, A. E. (2007). Super learner. *Statistical applications in genetics and molecular biology*, 6(1).

Larivière, B. and Van den Poel, D. (2005). Predicting customer retention and profitability by using random forests and regression forests techniques. *Expert Systems with Applications*, 29(2):472–484.

LeDell, E. E. (2015). *Scalable Ensemble Learning and Computationally Efficient Variance Estimation*. University of California, Berkeley.

Lee, S. M. (1972). *Goal programming for decision analysis*. Auerbach Publishers Philadelphia.

Li, G. (2013). Application of improved k-means clustering algorithm in customer segmentation. volume 411-414, pages 1081–1084.

Li, X. and Feng, J. (2017). A model of cross selling based on association rules. *Boletin Tecnico/Technical Bulletin*, 55(12):256–262.

Li, Y., Murali, P., Shao, N., and Sheopuri, A. (2016). Applying data mining techniques to direct marketing: Challenges and solutions. pages 319–327. Institute of Electrical and Electronics Engineers Inc.

Lin, W. C., Tsai, C. F., Hu, Y. H., and Jhang, J. S. (2017). Clustering-based undersampling in class-imbalanced data. *Information Sciences*, 409-410:17–26.

Ling, C. X. and Li, C. (1998). Data mining for direct marketing: Problems and solutions. In *KDD*, volume 98, pages 73–79.

Lismont, J., Ram, S., Vanthienen, J., Lemahieu, W., and Baesens, B. (2018). Predicting interpurchase time in a retail environment using customer-product networks: An empirical study and evaluation. *Expert Systems with Applications*, 104:22–32.

Ma, S., Hou, L., Yao, W., and Lee, B. (2016). A nonhomogeneous hidden markov model of response dynamics and mailing optimization in direct marketing. *European Journal of Operational Research*, 253(2):514–523.

Mahar, S., Salzarulo, P. A., and Wright, P. D. (2017). Simultaneous use of customer, product and inventory information in dynamic product promotion. *International Journal of Production Research*, pages 1–17.

Maringer, D. (2005). *Heuristic Optimization*, pages 38–76. Springer US, Boston, MA.

Michailidis, M. (2018). How driverless ai prevents overfitting and leakage. `https://blog.h2o.ai/2018/03/driverless-ai-prevents-overfitting-leakage/` [Accessed: 12th May 2018].

Miguéis, V. L., Camanho, A. S., and Borges, J. (2017). Predicting direct marketing response in banking: comparison of class imbalance methods. *Service Business*, 11(4):831–849.

Miguéis, V. L., Camanho, A. S., and Cunha, J. F. E. (2011). Mining customer loyalty card programs: The improvement of service levels enabled by innovative segmentation and promotions design. In *Lecture Notes in Business Information Processing*, volume 82 LNBIP, pages 83–97. Springer Verlag.

Minghua, H. (2008). Customer segmentation model based on retail consumer behavior analysis. pages 914–917.

Moro, S., Cortez, P., and Rita, P. (2014). A data-driven approach to predict the success of bank telemarketing. *Decision Support Systems*, 62:22–31.

Nakano, S. and Kondo, F. N. (2018). Customer segmentation with purchase channels and media touchpoints using single source panel data. *Journal of Retailing and Consumer Services*, 41:142–152.

Nash, E. L. (1984). *Direct marketing handbook*. McGraw-Hill.

Nee, D. (2014). Calibrating classifier probabilities. `http://danielnee.com/tag/platt-scaling/` [Accessed: 5th May 2018].

Ngai, E. W. T., Xiu, L., and Chau, D. C. K. (2009). Application of data mining techniques in customer relationship management: A literature review and classification. *Expert Systems with Applications*, 36(2 PART 2):2592–2602.

Niculescu-Mizil, A. and Caruana, R. (2005). Predicting good probabilities with supervised learning. In *Proceedings of the 22nd international conference on Machine learning*, pages 625–632. ACM.

Nielsen, M. (2017). *Neural Network and Deeplearning*, chapter How back propagation algorithm works. published online.

Ofcom (2008). Ofcom fines barclaycard maximum amount for silent calls. `https://www.ofcom.org.uk/about-ofcom/latest/media/media-releases/2008/ofcom-fines-barclaycard-maximum-amount-for-silent-calls` [Accessed: 9th June 2018].

Olson, D. L. and Chae, B. (2012). Direct marketing decision support through predictive customer response modeling. *Decision Support Systems*, 54(1):443–451.

Osuna, I., González, J., and Capizzani, M. (2016). Which categories and brands to promote with targeted coupons to reward and to develop customers in supermarkets. *Journal of Retailing*, 92(2):236–251.

Platt, J. (1999). Probabilistic outputs for support vector machines and comparisons to regularized likelihood methods. *Advances in large margin classifiers*, 10(3):61–74.

Powers, D. M. (2011). Evaluation: from precision, recall and f-measure to roc, informedness, markedness and correlation.

Providencia, R., Marijon, E., Barra, S., Reitan, C., Breitenstein, A., Defaye, P., Papageorgiou, N., Duehmke, R., Winnik, S., Ang, R., Klug, D., Gras, D., Oezkartal, T., Segal, O. R., Deharo, J. C., Leclercq, C., Lambiase, P. D., Fauchier, L., Bordachar, P., Steffel, J., Sadoul, N., Piot, O., Borgquist, R., Agarwal, S., Chow, A., Boveda, S., and Investigators, D.-P. (2018). Usefulness of a clinical risk score to predict the response to cardiac resynchronization therapy. *International Journal of Cardiology*, 260:82–87.

Quinlan, J. R. (1986). Induction of decision trees. *Machine Learning*, 1(1):81–106.

Raychaudhuri, S. (2008). Introduction to monte carlo simulation. In *Simulation Conference, 2008. WSC 2008. Winter*, pages 91–100. IEEE.

Reutterer, T., Hornik, K., March, N., and Gruber, K. (2017). A data mining framework for targeted category promotions. *Journal of Business Economics*, 87(3):337–358.

Rhee, E. and McIntyre, S. (2009). How current targeting can hinder targeting in the future and what to do about it. *Journal of Database Marketing and Customer Strategy Management*, 16(1):15–28.

Ribeiro, M. T., Singh, S., and Guestrin, C. (2016). "why should i trust you?" explaining the predictions of any classifier. volume 13-17-August-2016, pages 1135–1144. Association for Computing Machinery.

Rossi, P. E., McCulloch, R. E., and Allenby, G. M. (1996). The value of purchase history data in target marketing. *Marketing Science*, 15(4):321–340.

Samuelson, D. A. (1999). Predictive dialing for outbound telephone call centers. *Interfaces*, 29(5):66–81.

Schapire, R. E. (1990). The strength of weak learnability. *Machine Learning*, 5(2):197–227.

Schuller, P. (2018). *A machine learning approach to promotional sales forecasting*. Thesis, Faculdade de Engenharia da Universiadade do Porto.

Strobl, C., Boulesteix, A.-L., Zeileis, A., and Hothorn, T. (2007). Bias in random forest variable importance measures: Illustrations, sources and a solution. *BMC Bioinformatics*, 8:25–25.

Talla Nobibon, F., Leus, R., and Spieksma, F. C. R. (2011). Optimization models for targeted offers in direct marketing: Exact and heuristic algorithms. *European Journal of Operational Research*, 210(3):670–683.

Tsai, C. F. and Lu, Y. H. (2009). Customer churn prediction by hybrid neural networks. *Expert Systems with Applications*, 36(10):12547–12553.

Valle, M. A., Ruz, G. A., and Morrás, R. (2018). Market basket analysis: Complementing association rules with minimum spanning trees. *Expert Systems with Applications*, 97:146–162.

Venkatesan, R. and Farris, P. W. (2012). Measuring and managing returns from retailer-customized coupon campaigns. *Journal of Marketing*, 76(1):76–94.

Wattal, S., Telang, R., Mukhopadhyay, T., and Boatwright, P. (2012). What's in a "name"? impact of use of customer information in e-mail advertisements. *Information Systems Research*, 23(3 PART 1):679–697.

Žliobaitė, I. (2010). Learning under concept drift: an overview. *arXiv preprint arXiv:1010.4784*.

Appendix A

Performance metrics

$$PCC = \frac{TP+TN}{TP+FP+TN+FN} \tag{A.1}$$

$$Precision = \frac{TP}{TP+FP} \quad \text{and} \quad Recall = \frac{TP}{TP+FN} \tag{A.2}$$

$$F_\beta = (1+\beta)^2 \cdot \frac{Precision \cdot Recall}{(\beta^2 \cdot Precision) + Recall} = \frac{(1+\beta)^2 \cdot TP}{(1+\beta)^2 \cdot TP + \beta^2 \cdot FN + TP} \tag{A.3}$$

$$kappa = \frac{PCC_{actual} - PCC_{expected}}{1 - PCC_{expected}} \tag{A.4}$$

$$BrierScore = \frac{1}{n} \cdot \sum_{i=1}^{n} (f_i - o_i)^2 \tag{A.5}$$

$$Liftindex = \frac{\sum_{j=1}^{10} (w_j \cdot TP_j)}{TP+FN} \tag{A.6}$$

$$Bias = \frac{1}{n} \cdot \sum_{i=1}^{n} (f_i - o_i) \tag{A.7}$$

$$AUC = \int_0^1 ROC(u)\, du \tag{A.8}$$

Where :

β	constant (typically $= 2$) to adjust the importance given to FP and FN
$PCC_{expected}$	PCC of a random guesser
f_i, o_i	forecasted and true probabilities of observation i
w_j, TP_j	weigth and recall of decile j
$ROC(u)$	receiver operating characteristic for threshold $u \in [0,1]$

Appendix B

Boruta algorithm - pseudocode

```
Input:  input dataset, **A**
        list of candidate variables, $\mathscr{G}$
        stopping criteria limiting the number of iterations, $nruns$
Output: list of variables shown to be explicative of the dependent variable, $\mathscr{L}$
$Undecided \leftarrow \mathscr{G}$
while $i \leq nruns$ & $Undecided \neq \emptyset$ do
    $\mathbf{B} \leftarrow \mathbf{A}$
    Shuffle B row-wise to remove correlations with the response variable
    $\mathbf{C} \leftarrow \mathbf{A} \cup \mathbf{B}$
    Train a random forest classifier on the extended dataset C
    $ZS \leftarrow$ variable importance Z scores
    $ZSA \leftarrow$ Z-score of shadow attributes
    $MZSA \leftarrow \max_i (ZSA_i)$
    foreach $variable \in \mathscr{G}$ do
        if $ZS[variable] > MZSA$ then
            $variable[hits] \leftarrow variable[hits] + 1$
        end
        if $variable \in Undecided$ then
            $testeResult \leftarrow$ result of two-sided test of equality between $ZS[variable]$ and
              $MZSA$
            if $MZSA > ZS[variable]$ & $testResult[p\_value] < 0.025$ then
                $variable[importance] =$ "Unimportant"
                remove $variable$ from $Undecided$
            else if $ZS[variable] > MZSA$ & $testResult[p\_value] < 0.025$ then
                $variable[importance] =$ "Important"
                remove $variable$ from $Undecided$
                include $variable$ in $\mathscr{L}$
        end
    end
    $i \leftarrow i + 1$
end
```

Algorithm 1: Boruta Algorithm (source: Kursa and Rudnicki (2010))

Appendix C

Variables included in the model

Table C.1: Predictors of response (excluding dialer's history)

Source	Variable name	Description	Imp[1]
Customer	antig_account	number of months since account opening	Yes
	antig_net	number of months since internet subscription	Yes
	antig_voice	number of month since mobile phone subscription	Yes
	conc_profile	concorrential profile[2]	Yes
	conc_segmentation	concorrential segmentation[2]	Yes
	end_pf_day_n	days until contract expiratition date	Yes
	flg_4g	binary variable indicating 4g service subscription	Yes
	geo_district	geographical distribution	Yes
	buying_profile	customer's buying profile[3]	Yes
	premium	number of premium TV channels subscribed	Yes
	base_revenue	subscription's fee before campaign	Yes
	tecn_profile_subs	technical profile of subscription (cable, sattelite, . . .)	Yes
Contact Specific	campaign	type of campaign (prospecting, upselling, . . .)	Yes
	hour	timing of contact	Yes
	mobile_land_flag	flag indicative of mobile or land phone	Yes
	week_day	day of week when the contact is made	Yes
Interaction "metadata" records	nr_epg_requests	number of a specific TV box request, last month during hour k[4]	No
	nr_interactions_tvbox	number of interactions with the TV box, last month during hour k[4]	Yes
	nr_npvr_pvr	number of recording TV box requests, last month during hour k[4]	No
	nr_restarts_tvbox	number of restarts of TV box, last month during hour k[4]	Yes
	nr_videoondemand	number of video-on-demand requests, last month during hour k[4]	Yes
	sum_airtime_in	monthly average of minutes on calls received during hour k[4]	Yes

[1] variable importance confirmed through the Boruta algorithm
[2] segmentation made in regards to propensity to subdue to competitors' prospecting initiatives
[3] profiling available in the company's database
[4] hour k is the timing of contact

Table C.2: Predictors of response from dialer's history

Variable	Description[1]	Window[2]	CN[3]	Hour[4]	Imp[5]
num_answ	number of answered calls	ALL	No	No	Yes
num_answ_30d		30d	No	No	Yes
num_answ_30d_hour		30d	No	Yes	Yes
num_answ_30d_hour_near		30d	No	Nearby	Yes
num_answ_30d_CN		30d	Yes	No	Yes
num_answ_30d_CN_hour_near		30d	Yes	Nearby	Yes
num_answ_hour		ALL	No	Yes	Yes
num_answ_hour_near		ALL	No	Nearby	Yes
num_answ_CN		ALL	Yes	No	Yes
num_answ_CN_hour_near		ALL	Yes	Nearby	Yes
num_att	number of attempted calls	ALL	No	No	Yes
num_att_30d		30d	No	No	Yes
num_att_30d_hora		30d	No	Yes	No
num_att_30d_hora_near		30d	No	Nearby	Yes
num_att_30d_CN		30d	Yes	No	Yes
num_att_30d_CN_hora_near		30d	Yes	Nearby	Yes
num_att_hora		ALL	No	Yes	Yes
num_att_hora_near		ALL	No	Nearby	No
num_att_CN		ALL	Yes	No	Yes
num_att_CN_hora_near		ALL	Yes	Nearby	Yes
num_usef	number of useful contacts	ALL	No	No	Yes
num_usef_30d		30d	No	No	Yes
num_usef_30d_hour		30d	No	Yes	Yes
num_usef_30d_hour_near		30d	No	Nearby	Yes
num_usef_30d_CN		30d	Yes	No	Yes
num_usef_30d_CN_hour_near		30d	Yes	Nearby	Yes
num_usef_hour		ALL	No	Yes	Yes
num_usef_hour_near		ALL	No	Nearby	Yes
num_usef_CN		ALL	Yes	No	Yes
num_usef_CN_hour_near		ALL	Yes	Nearby	No
freq	frequency of answered calls	ALL	No	No	Yes
freq_30d		30d	No	No	Yes
freq_30d_hour		30d	No	Yes	Yes
freq_30d_hour_near		30d	No	Nearby	Yes
freq_30d_CN		30d	Yes	No	Yes
freq_30d_CN_hour_near		30d	Yes	Nearby	Yes
freq_hour		ALL	No	Yes	Yes
freq_hour_near		ALL	No	Nearby	Yes
freq_CN		ALL	Yes	No	Yes
freq_CN_hour_near		ALL	Yes	Nearby	Yes

[1] general description of the variable
[2] window with respect to which the variable is calculated (previous 30 days or the whole dialer's history)
[3] flag indicative of the fact that the variable acts on the specific customer number
[4] describes if variables are specific to hour k, act between hour k-1 and k+1 or all hours
[5] variable importance confirmed through the Boruta algorithm

Table C.3: Predictors of response from dialer's history (continued)

Variable	Description[1]	Window[2]	CN[3]	Hour[4]	Imp[5]
freq_usef	frequency of useful contacts	ALL	No	No	Yes
freq_usef_30d		30d	No	No	Yes
freq_usef_30d_hour		30d	No	Yes	Yes
freq_usef_30d_hour_near		30d	No	Nearby	Yes
freq_usef_30d_CN		30d	Yes	No	Yes
freq_usef_30d_CN_hour_near		30d	Yes	Nearby	Yes
freq_usef_hour		ALL	No	Yes	Yes
freq_usef_hour_near		ALL	No	Nearby	Yes
freq_usef_CN		ALL	Yes	No	Yes
freq_usef_CN_hour_near		ALL	Yes	Nearby	Yes
hour_weight	percentage of all answered calls that happened during that timing	ALL	No	Yes	Yes
hour_near_weight		ALL	No	Nearby	Yes
recency_answ	days since last answer	ALL	No	No	Yes
recency_answ_hour		ALL	No	Yes	Yes
recency_answ_hour_CN		ALL	Yes	Yes	Yes
recency_answ_CN		ALL	Yes	No	Yes
recency_att	days since last attempt	ALL	No	No	Yes
recency_att_hour		ALL	No	Yes	Yes
recency_att_hour_CN		ALL	Yes	Yes	Yes
recency_att_CN		ALL	Yes	No	Yes
recency_usef	days since last useful contact	ALL	No	No	Yes
recency_usef_hour		ALL	No	Yes	Yes
recency_usef_hour_CN		ALL	Yes	Yes	Yes
recency_usef_CN		ALL	Yes	No	Yes
outcome_last_att	outcome of last attempt	ALL	No	No	Yes
outcome_last_att_hour		ALL	No	Yes	Yes
outcome_last_att_hour_CN		ALL	Yes	Yes	Yes
outcome_last_att_CN		ALL	Yes	No	Yes
duration_last_att_CN	duration of last attempt to CN	ALL	Yes	No	Yes
duration_last_att_ok	duration of last answer to CN	ALL	Yes	No	Yes
duration_last_att_usef	duration of last useful contact to CN	ALL	Yes	No	Yes

[1] general description of the variable

[2] window with respect to which the variable is calculated (previous 30 days or the whole dialer's history)

[3] flag indicative of the fact that the variable acts on the specific customer number

[4] describes if variables are specific to hour k, act between hour k-1 and k+1 or all hours

[5] variable importance confirmed through the Boruta algorithm

Appendix D

Hyperparameter search

Model	Parameter	Levels	Short explanation
RF	number of trees	50,100,150	maximum number of trees grown in the forest
	number of bins	8,16,32,64	number of bins for the histogram to build
	max depth	3:18, by = 1	length of the longest path from a root to a leaf
	col sample rate	0.5:1.0, by = 0.1	fraction of predictors sampled for splitting
GBM	number of trees	50,100,150,200	maximum number of trees grown in the forest
	number of bins	8,16,32,64	number of bins for the histogram to build
	max depth	5:15, by = 1	length of the longest path from a root to a leaf
	sample rate	0.5:1.0, by = 0.1	fraction of rows sampled from training dataset
	col sample rate	0.4:1.0, by = 0.1	fraction of predictors sampled for splitting
	learn rate	0.01:0.1, by = 0.01	rate of error correction from each tree to the next
Log Reg	alpha	0.0:1.0, by=0.25	regularization distribution between L1 and L2
	lambda	0, 1e-5, 1e-4, 1e-3	regularization strength
ANN	activation	Rectifier, Maxout	neuron activation function
	hidden layer	(10,10);(25,25);(30,30,30)	configuration of hidden layer (number of neurons)
	epochs	50,100,150	number of full cycles (forward and backward)
	l1	0, 1e-4, 1e-3	controls l1 regularization (lets only strong weights survive)
	l2	0, 1e-4, 1e-3	controls l2 regularization (prevents any single weight from getting too big)

Appendix E

Hyperparameter search results

Table E.1: ANN hyperparameter search

Model #	Hour	Activation	Brier Score	Hidden Layer	Epochs	l1	l2
1	10	Rectifier	0,0806	25 25	41	1E-03	0E+00
2	10	Rectifier	0,0808	30 30 30	38	1E-04	1E-03
3	10	Rectifier	0,0808	10 10	57	1E-04	1E-03
4	10	Maxout	0,0812	25 25	11	0E+00	0E+00
5	10	Maxout	0,0812	10 10	48	1E-03	1E-04
6	10	Rectifier	0,0814	10 10	43	1E-03	1E-04
7	10	Maxout	0,0817	10 10	35	1E-04	0E+00
8	10	Maxout	0,0820	25 25	18	1E-04	1E-03
9	10	Maxout	0,0820	25 25	16	0E+00	1E-04
10	10	Maxout	0,0820	25 25	10	0E+00	1E-04
11	10	Maxout	0,0821	30 30 30	16	0E+00	1E-03
12	10	Maxout	0,0825	12 12 12	34	1E-04	1E-04
13	10	Rectifier	0,0831	30 30 30	50	1E-04	1E-03
14	10	Rectifier	0,0840	10 10	96	1E-03	1E-04
15	10	Rectifier	0,0840	12 12 12	89	0E+00	1E-03
16	10	Maxout	0,0860	12 12 12	15	1E-04	1E-04
1	15	Maxout	0,0885	25 25	15	1E-04	0E+00
2	15	Rectifier	0,0887	10 10	49	0E+00	1E-03
3	15	Maxout	0,0887	10 10	23	0E+00	1E-04
4	15	Maxout	0,0890	25 25	12	0E+00	1E-03
5	15	Rectifier	0,0891	25 25	43	1E-03	1E-03
6	15	Rectifier	0,0897	12 12 12	50	1E-03	1E-03
7	15	Rectifier	0,0898	30 30 30	49	0E+00	1E-04
8	15	Rectifier	0,0902	10 10	86	0E+00	1E-04
9	15	Rectifier	0,0902	10 10	50	0E+00	1E-04
10	15	Maxout	0,0903	12 12 12	31	1E-03	1E-04
11	15	Maxout	0,0904	10 10	25	0E+00	1E-04
12	15	Maxout	0,0905	10 10	39	1E-03	1E-03
13	15	Maxout	0,0911	25 25	13	1E-03	0E+00
14	15	Maxout	0,0912	30 30 30	11	1E-03	0E+00
15	15	Rectifier	0,0917	10 10	58	1E-03	1E-03
16	15	Maxout	0,0922	12 12 12	40	1E-04	1E-03
1	18	Rectifier	0,0964	12 12 12	49	1E-03	1E-03
2	18	Rectifier	0,0968	25 25	32	0E+00	0E+00
3	18	Rectifier	0,0969	25 25	38	1E-03	1E-04
4	18	Rectifier	0,0973	10 10	47	1E-03	0E+00
5	18	Maxout	0,0974	25 25	20	1E-04	0E+00
6	18	Maxout	0,0979	25 25	15	0E+00	1E-03
7	18	Maxout	0,0983	10 10	20	1E-03	1E-03
8	18	Maxout	0,0984	10 10	21	1E-04	1E-03
9	18	Maxout	0,0985	10 10	21	1E-04	1E-03
10	18	Maxout	0,0988	10 10	27	0E+00	1E-03
11	18	Rectifier	0,0991	30 30 30	27	1E-04	1E-04
12	18	Rectifier	0,0992	12 12 12	79	1E-03	1E-03
13	18	Maxout	0,0995	10 10	40	1E-04	1E-04
14	18	Rectifier	0,1000	25 25	80	0E+00	0E+00
15	18	Rectifier	0,1018	10 10	88	1E-03	1E-03
16	18	Rectifier	0,1022	10 10	64	1E-03	1E-04
1	20	Maxout	0,0890	10 10	23	1E-04	1E-03
2	20	Maxout	0,0894	25 25	24	0E+00	1E-03
3	20	Rectifier	0,0895	12 12 12	112	1E-03	1E-03
4	20	Rectifier	0,0896	12 12 12	55	1E-03	1E-04
5	20	Maxout	0,0896	25 25	14	1E-04	1E-03
6	20	Rectifier	0,0896	30 30 30	54	0E+00	0E+00
7	20	Maxout	0,0896	10 10	47	1E-04	1E-03
8	20	Maxout	0,0897	12 12 12	29	1E-04	1E-03
9	20	Rectifier	0,0898	10 10	42	1E-03	0E+00
10	20	Maxout	0,0899	25 25	20	1E-03	1E-03
11	20	Rectifier	0,0899	12 12 12	35	1E-04	1E-03
12	20	Maxout	0,0900	25 25	29	1E-03	0E+00
13	20	Rectifier	0,0902	10 10	92	1E-03	1E-03
14	20	Rectifier	0,0902	30 30 30	28	1E-04	1E-04
15	20	Rectifier	0,0903	10 10	48	1E-04	1E-03
16	20	Rectifier	0,0906	10 10	50	1E-04	1E-03

Table E.2: Logistic Regression hyperparameter search

Model #	**Hour**	**Brier Score**	**Lambda**	**Alpha**
1	10	0,08138	0E+00	0,75
2	10	0,08138	1E-05	0,75
3	10	0,08140	1E-04	0
4	10	0,08142	1E-04	0,25
5	10	0,08146	1E-04	1
6	10	0,08151	1E-03	0,25
7	10	0,08160	1E-03	0,75
8	10	0,08254	1E-02	1
9	10	0,08526	1E+00	0
10	10	0,08555	1E-01	0,25
11	10	0,09067	1E+00	0,5
12	10	0,09067	1E-01	1
13	10	0,09067	1E+00	1
14	10	0,09067	1E-01	0,75
15	10	0,09067	1E+00	0,75
16	10	0,09067	1E+00	0,25
1	15	0,08931	1E-04	0
2	15	0,08932	1E-05	0,75
3	15	0,08932	1E-05	0,5
4	15	0,08932	1E-05	0,25
5	15	0,08933	0E+00	0,25
6	15	0,08943	1E-03	0,75
7	15	0,08948	1E-03	1
8	15	0,08950	1E-02	0
9	15	0,09042	1E-02	0,75
10	15	0,09447	1E-01	0,25
11	15	0,09758	1E-01	0,5
12	15	0,10030	1E+00	0,5
13	15	0,10030	1E-01	1
14	15	0,10030	5E-01	0,75
15	15	0,10030	1E+00	0,75
16	15	0,10030	1E+00	1
1	18	0,09729	0E+00	0,5
2	18	0,09729	0E+00	1
3	18	0,09729	1E-05	1
4	18	0,09729	1E-04	0
5	18	0,09736	1E-04	1
6	18	0,09737	1E-03	0
7	18	0,09765	1E-03	0,75
8	18	0,09858	1E-02	0,5
9	18	0,09901	1E-02	0,75
10	18	0,10430	1E-01	0,25
11	18	0,10849	1E-01	0,5
12	18	0,11200	1E+00	0,5
13	18	0,11200	1E-01	0,75
14	18	0,11200	5E-01	0,25
15	18	0,11200	5E-01	0,5
16	18	0,11200	5E-01	1
1	20	0,08928	1E-02	0,25
2	20	0,08929	1E-03	1
3	20	0,08934	1E-03	0,25
4	20	0,08943	1E-03	0
5	20	0,08945	0E+00	0
6	20	0,08945	0E+00	0,5
7	20	0,08945	0E+00	1
8	20	0,08945	1E-05	1
9	20	0,08965	1E-02	0,75
10	20	0,08969	1E-01	0
11	20	0,09207	1E+00	0
12	20	0,09453	1E-01	0,5
13	20	0,09650	1E-01	1
14	20	0,09650	1E+00	0,75
15	20	0,09650	5E-01	0,75
16	20	0,09650	1E+00	0,5

Table E.3: GBM hyperparameter search

Model #	**Hour**	**Brier Score**	**Ntrees**	**Learn rate**	**Max depth**	**Sample rate**	**Col sample rate**
1	10	0,0796	200	0,05	7	0,8	0,8
2	10	0,0799	150	0,05	9	0,7	0,6
3	10	0,0801	100	0,10	8	0,6	0,5
4	10	0,0803	200	0,03	9	0,7	0,9
5	10	0,0805	100	0,08	10	0,5	0,7
6	10	0,0805	50	0,10	10	0,6	1,0
7	10	0,0804	200	0,02	12	0,9	0,4
8	10	0,0817	138	0,07	14	0,5	0,5
9	10	0,0815	50	0,10	13	0,6	0,4
10	10	0,0815	100	0,07	15	0,6	0,4
11	10	0,0820	150	0,02	12	0,6	0,7
12	10	0,0815	110	0,09	11	0,5	0,6
13	10	0,0827	100	0,09	15	0,8	0,8
14	10	0,0822	50	0,02	13	0,6	0,9
15	10	0,0827	200	0,01	7	0,5	1,0
16	10	0,0840	108	0,09	13	0,6	0,5
1	15	0,0880	200	0,03	11	0,6	0,6
2	15	0,0881	100	0,04	11	0,9	0,4
3	15	0,0883	196	0,04	11	0,8	0,6
4	15	0,0883	100	0,03	11	0,5	0,9
5	15	0,0886	50	0,05	5	0,8	0,7
6	15	0,0891	138	0,09	11	0,8	0,5
7	15	0,0899	50	0,10	13	0,9	0,4
8	15	0,0900	50	0,03	6	0,7	0,8
9	15	0,0905	134	0,08	13	0,8	1,0
10	15	0,0916	50	0,05	14	0,9	0,6
11	15	0,0916	146	0,02	14	0,7	0,7
12	15	0,0917	200	0,01	5	0,8	0,6
13	15	0,0919	100	0,04	14	0,7	0,9
14	15	0,0919	108	0,09	13	0,5	0,5
15	15	0,0949	50	0,10	15	1	1,0
16	15	0,0952	50	0,01	13	1	0,4
1	18	0,0951	100	0,08	7	1	0,4
2	18	0,0954	200	0,08	5	0,5	0,7
3	18	0,0958	100	0,06	10	0,9	0,8
4	18	0,0961	100	0,07	9	0,7	0,9
5	18	0,0967	50	0,08	12	1	0,4
6	18	0,0967	150	0,05	13	0,5	0,4
7	18	0,0967	100	0,03	15	0,8	0,4
8	18	0,0975	126	0,10	11	0,6	0,6
9	18	0,0977	50	0,03	6	0,8	0,7
10	18	0,0978	134	0,07	13	0,9	1,0
11	18	0,0988	50	0,09	12	0,9	0,6
12	18	0,0990	100	0,02	8	1	0,6
13	18	0,0992	50	0,02	7	0,5	1,0
14	18	0,1011	128	0,03	15	0,6	0,6
15	18	0,1026	150	0,01	12	0,8	1,0
16	18	0,1056	50	0,01	10	1	0,5
1	20	0,0884	144	0,10	7	0,5	0,9
2	20	0,0884	200	0,03	7	1	0,7
3	20	0,0884	200	0,08	5	0,9	0,9
4	20	0,0888	100	0,09	8	0,9	1,0
5	20	0,0889	150	0,09	6	0,5	1,0
6	20	0,0892	132	0,08	9	1	0,5
7	20	0,0894	50	0,10	9	0,8	0,5
8	20	0,0894	100	0,03	12	1	1,0
9	20	0,0894	100	0,05	12	0,7	0,4
10	20	0,0898	50	0,08	10	0,5	0,8
11	20	0,0908	50	0,10	12	0,6	0,6
12	20	0,0910	100	0,01	7	0,7	1,0
13	20	0,0914	50	0,07	12	0,9	0,7
14	20	0,0919	100	0,04	13	0,7	0,8
15	20	0,0924	122	0,09	14	0,5	0,5
16	20	0,0925	150	0,02	15	0,7	1,0

Table E.4: RF hyperparameter search

Model #	Hour	Brier score	Ntrees	Nbins	Max depth	Col sample rate
1	10	0,0804	150	64	15	0,6
2	10	0,0806	150	32	11	0,9
3	10	0,0806	50	64	13	0,5
4	10	0,0806	100	32	12	0,5
5	10	0,0807	100	8	16	0,6
6	10	0,0807	150	8	12	0,7
7	10	0,0809	50	16	16	0,6
8	10	0,0809	50	8	10	0,8
9	10	0,0810	50	8	9	0,9
10	10	0,0810	50	32	8	0,8
11	10	0,0811	150	8	10	0,9
12	10	0,0813	144	16	7	0,6
13	10	0,0813	150	32	13	0,6
14	10	0,0813	100	8	12	0,8
15	10	0,0816	150	8	13	0,8
16	10	0,0836	150	8	17	0,8
1	15	0,0888	50	16	15	1,0
2	15	0,0888	100	16	10	0,7
3	15	0,0889	150	64	8	0,6
4	15	0,0889	50	32	9	0,7
5	15	0,0890	144	16	9	0,6
6	15	0,0890	50	16	17	0,8
7	15	0,0891	50	8	8	0,5
8	15	0,0893	50	32	6	0,6
9	15	0,0897	50	16	5	0,7
10	15	0,0897	100	32	5	0,6
11	15	0,0898	150	64	14	0,7
12	15	0,0902	50	8	15	0,5
13	15	0,0908	150	32	3	0,9
14	15	0,0908	50	16	3	0,8
15	15	0,0910	150	16	3	0,8
16	15	0,0920	100	64	17	1,0
1	18	0,0962	100	32	17	0,8
2	18	0,0962	150	16	16	1,0
3	18	0,0963	50	32	14	1,0
4	18	0,0965	100	16	17	0,9
5	18	0,0965	100	16	10	0,8
6	18	0,0967	50	16	10	0,5
7	18	0,0968	138	8	9	0,8
8	18	0,0971	50	8	10	0,5
9	18	0,0973	50	32	11	1,0
10	18	0,0975	146	64	12	0,6
11	18	0,0977	50	16	12	0,7
12	18	0,0977	50	64	6	0,6
13	18	0,0981	100	16	13	1,0
14	18	0,0996	150	64	3	0,5
15	18	0,0996	100	8	16	0,5
16	18	0,1026	100	64	18	1,0
1	20	0,0885	50	64	8	0,6
2	20	0,0886	150	32	8	0,7
3	20	0,0886	50	32	9	0,6
4	20	0,0886	100	32	8	0,7
5	20	0,0887	100	32	14	0,6
6	20	0,0888	150	32	6	0,8
7	20	0,0888	142	32	6	1,0
8	20	0,0888	100	16	6	0,7
9	20	0,0890	50	64	12	0,6
10	20	0,0890	150	8	17	0,7
11	20	0,0892	150	32	4	0,5
12	20	0,0892	50	64	15	0,8
13	20	0,0893	50	16	17	1,0
14	20	0,0896	150	8	3	1,0
15	20	0,0899	50	32	17	0,9
16	20	0,0902	100	8	14	0,8

Appendix F

Control and treatment samples comparison

To add to the argument that a randomized sampling procedure ensured that the customer accounts sent to the control and treatment groups were similar, the profiles of three variables are examined. In figure F.1, the geographical distribution of customers is studied. Figure F.2 analyzes the distribution of the base fee of subscription per customer accounts at the moment they were assigned to the campaign. At last, the technical profile of the subscription (cable, satellite or other) is looked at in figure F.3.

Although, to serve as an illustration, only three variables are showcased in the present document, the study conducted was far more thorough. The conclusion drawn from the analysis is that the samples have a very similar behavior across the board.

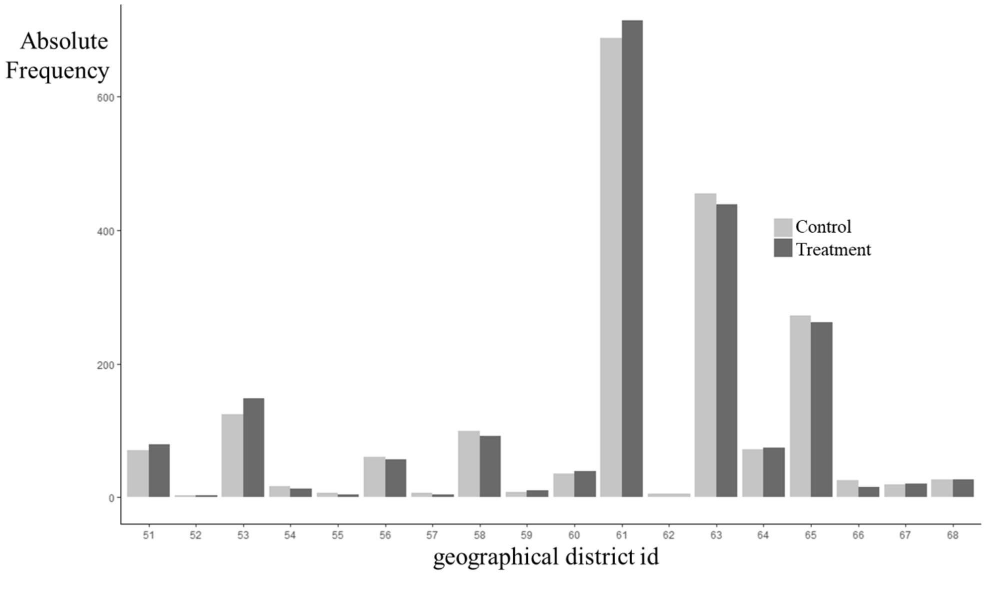


Figure F.1: Geographical distribution of customer accounts assigned to each sample

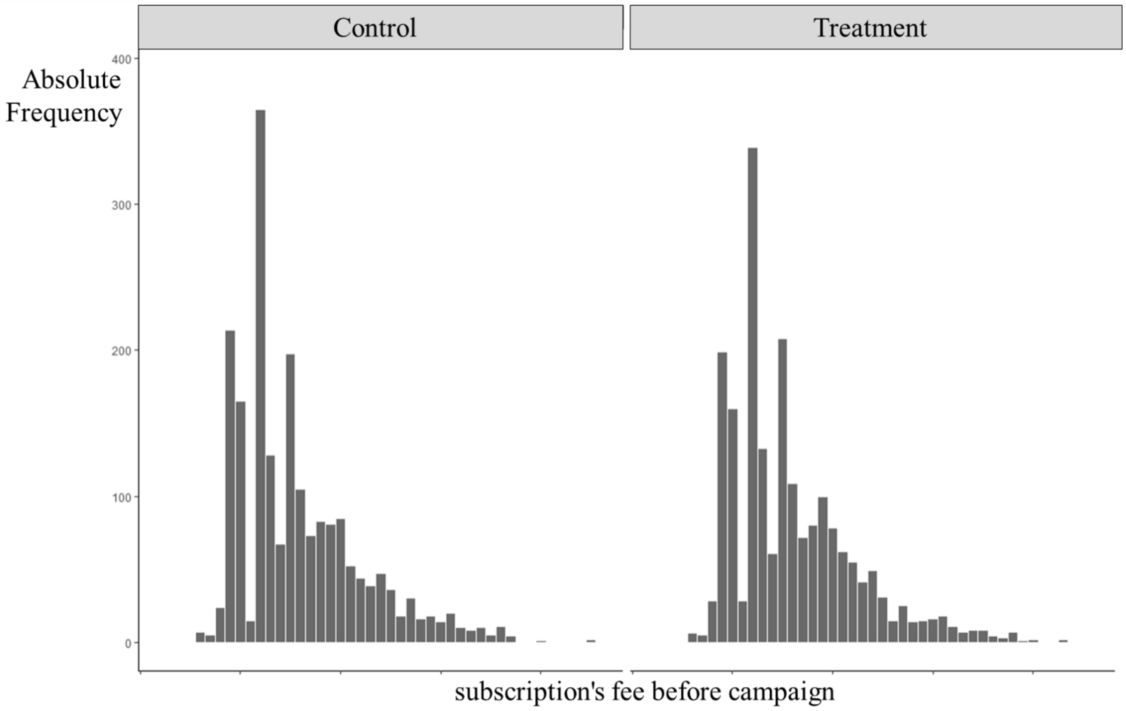


Figure F.2: Subscription fee’s distribution before the campaign started

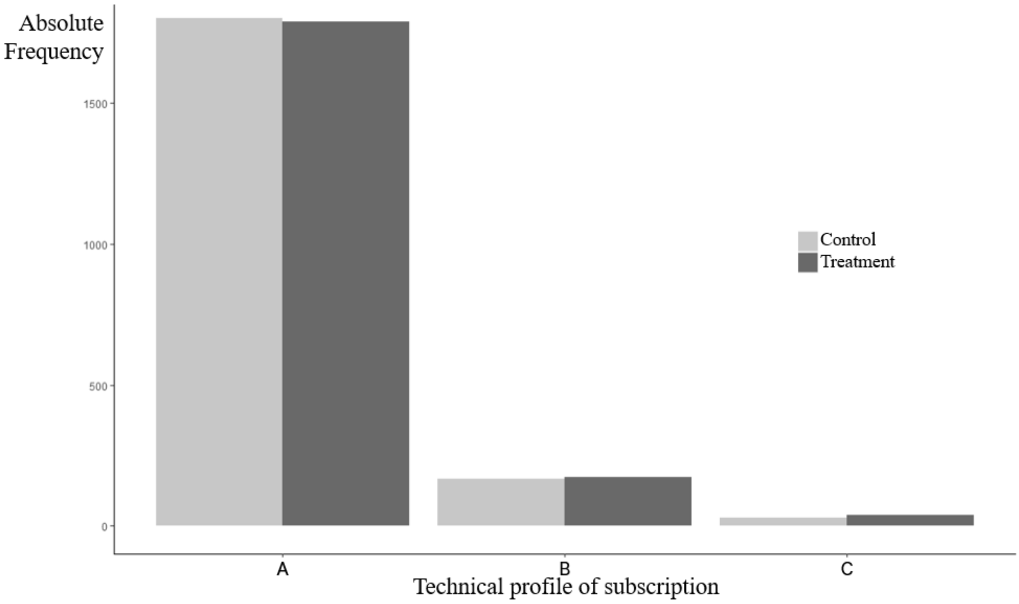


Figure F.3: Comparison of subscription’s technical profiles

Appendix G

Dashboard

As mentioned in the body of this thesis, a dashboard was crafted to accompany the progress of the pilot test. Said tool allowed for a comparison of an array of indicators between the control and treated groups at different granularities.

The main panel (figure G.1) greeted the users (mostly operational teams) with the indicators more meaningful to them, that is, sales evolution profiles and aggregate results. Occasionally, some values are not disclosed (ND) due to corporate confidentiality.

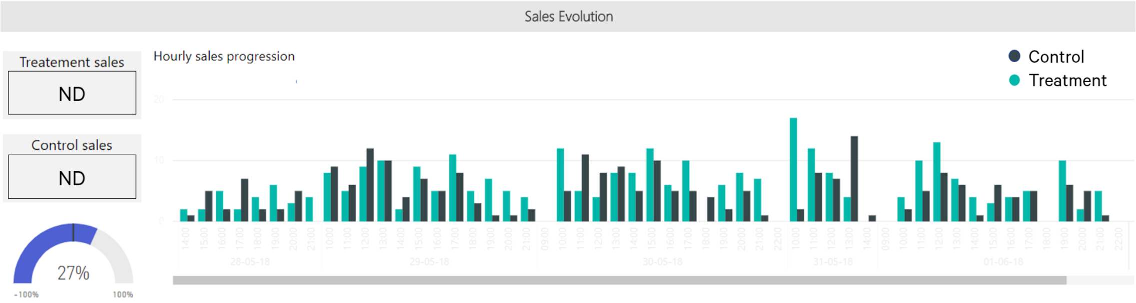


Figure G.1: Greeting panel

A section dedicated to an aggregate benchmark of all conversion rates (answering rate, useful contact rate and sales hit ratio) followed (figure G.2).

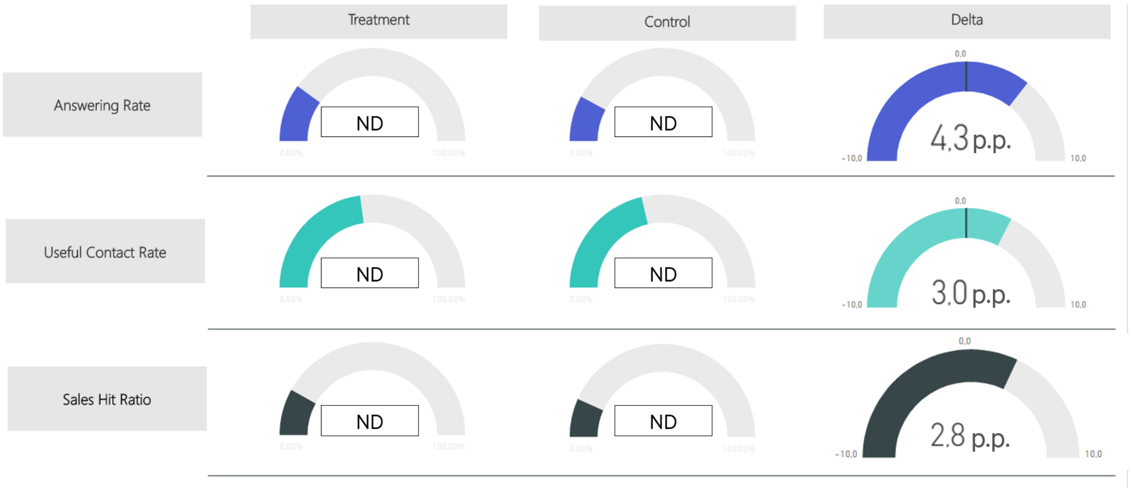


Figure G.2: Conversion rates' panel

Moreover, in order to provide a deeper understanding into each conversion rate's profile, the panel that followed tracked its hourly and daily evolutions (figure G.3).

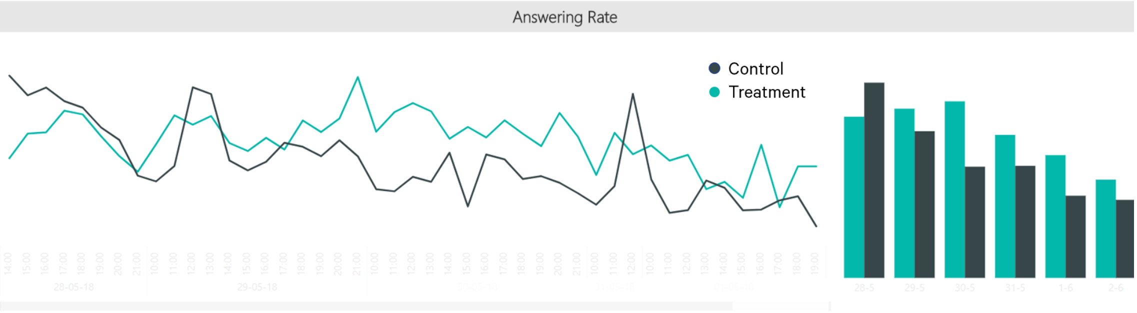


Figure G.3: Hourly and daily evolution of the answering rate

Besides, an overview of call duration for answered calls, useful contacts and sales was also included (figure G.4).

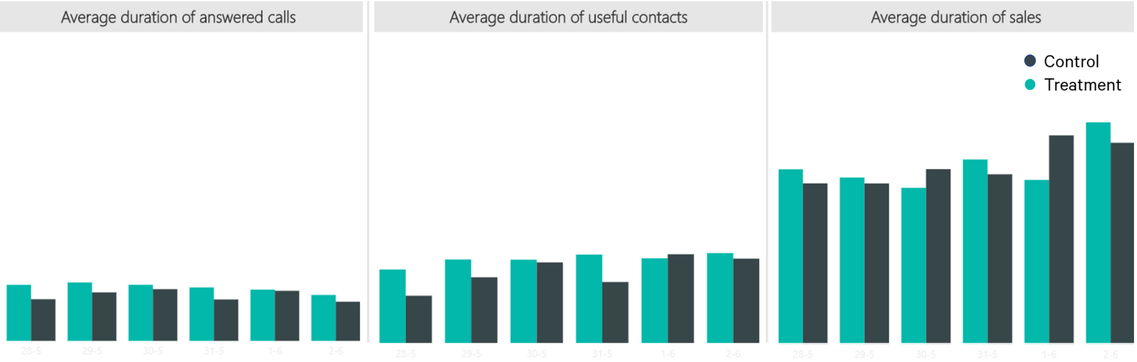


Figure G.4: Average daily call duration for answered calls, useful contacts and sales

Finally, to provide a global overview of the batch's behavior, several cumulative profiles were showcased. Those included the total number of attempted calls (presented as an example in figure G.5), the total number of useful contacts generated, the cumulative sales profile and the number of closed customer accounts.

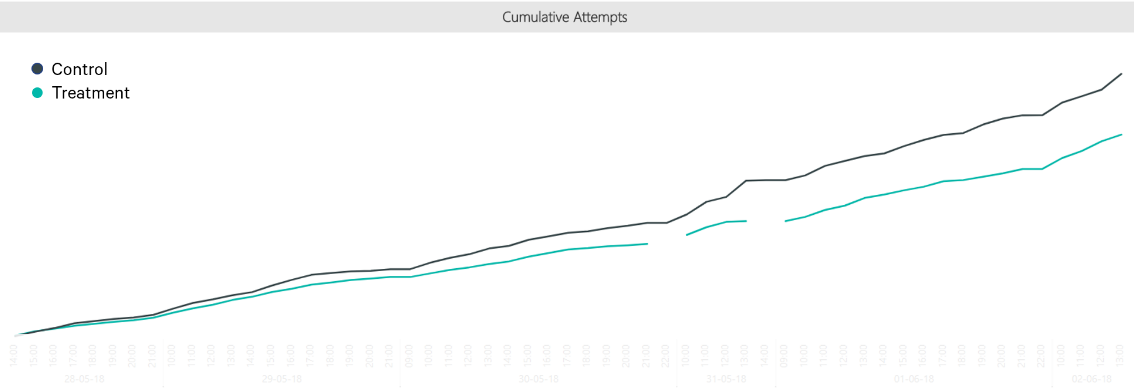


Figure G.5: Example of a cumulative evolution profile (cumulative calls attempted)