National College of Ireland

Sales Forecasting: Machine Learning Solution to B2B Sales Opportunity Win-Propensity Computation

MSc Research Project
Data Analytics

Marina Lambert
x16115350

School of Computing
National College of Ireland

Supervisor: Mr Jorge Basilio

National College of Ireland
Project Submission Sheet – 2017/2018
School of Computing

Student Name:	Marina Lambert
Student ID:	x16115350
Programme:	Data Analytics
Year:	2016-2018
Module:	MSc Research Project
Lecturer:	Mr Jorge Basilio
Submission Due Date:	13/08/2018
Project Title:	Sales Forecasting: Machine Learning Solution to B2B Sales Opportunity Win-Propensity Computation
Word Count:	9,001 (Excluding Submission Sheet, References, Appendix)

I hereby certify that the information contained in this (my submission) is information pertaining to research I conducted for this project. All information other than my own contribution will be fully referenced and listed in the relevant bibliography section at the rear of the project.

ALL internet material must be referenced in the bibliography section. Students are encouraged to use the Harvard Referencing Standard supplied by the Library. To use other author's written or electronic work is illegal (plagiarism) and may result in disciplinary action. Students may be required to undergo a viva (oral examination) if there is suspicion about the validity of their submitted work.

Signature:	
Date:	8th August 2018

PLEASE READ THE FOLLOWING INSTRUCTIONS:

1. Please attach a completed copy of this sheet to each project (including multiple copies).
2. **You must ensure that you retain a HARD COPY of ALL projects**, both for your own reference and in case a project is lost or mislaid. It is not sufficient to keep a copy on computer. Please do not bind projects or place in covers unless specifically requested.
3. Assignments that are submitted to the Programme Coordinator office must be placed into the assignment box located outside the office.

Office Use Only	
Signature:	
Date:	
Penalty Applied (if applicable):	

Sales Forecasting: Machine Learning Solution to B2B Sales Opportunity Win-Propensity Computation

Marina Lambert
x16115350
MSc Research Project in Data Analytics

8th August 2018

Abstract

This research is focused on finding an optimal machine learning solution for computation of a sales win-propensity score for B2B software sales. Currently sales propensity scoring at opportunity level is a manual, time-consuming and subjective task carried out by a salesperson. Customised model stack involving Random Forests, GLM, boosting, trees, and neural networks is a proposed solution in this research. Unlike reviewed related works, research focused on developing an approach using open-source tools and commonly met sales variables, hence proposed solution can be lifted and re-purposed cross-industry. Integration of provided solution into CRM will allow for accuracy improvement in sales forecasting, result in better planning, more effective resource management, saving time and costs.

Acknowledgements

This research would have been simply impossible without support of my supervisor, Mr. Jorge Basilio. His expertise, extensive support and advice, countless hours invested is what got this piece of work done. I am forever grateful.

I also would like to thank my husband, Alex, and my brother, Ruslan, for their support, cooked meals, and a great deal of patience provided during the course of the past two years.

1 Introduction

Sales forecasting is a vital process within any industry. Accuracy of sales forecasting affects not only sales, but many other areas within the business: strategy and planning, finance, marketing, operations, and company's performance assessment. (Lu and Kao; 2016; Yu et al.; 2011) Win-propensity of sales opportunities is an instrumental part of a sales forecast process and assessment of sales performance. (Monat; 2011)

Nevertheless of a major progress within forecasting methods as the outcome of machine learning research advancement, sales forecasting methods experienced very little improvement, especially within B2B sales domain (Bohanec et al.; 2017c). Most popular machine learning approach to sales forecast is time-series, mainly utilizing blackbox methods, high-level, and therefore, providing little insight into key drivers affecting specific

sale closure. (Bohanec et al.; 2017a) In contrast to time-series, research on propensity-scoring of sales opportunities using machine learning algorithms is almost non-existent, even though it is a fundamental process for sales business and its forecasting. (Yan et al.; 2015)

Single salesperson in a reasonably large organization manages multiple sales opportunities simultaneously. Within commercial sales this might mean hundreds of active opportunities at a time. (Tang, L. and Xu, X. and Rangan, V.; 2017) Majority of CRM systems provide an ability for a seller to score the opportunity subjectively via fields such as "Sales Stage" or "Forecast Stage" based on company sales operational framework. (Tang, L. and Xu, X. and Rangan, V.; 2017) Besides an obvious bias, it is found that salespersons are too optimistic in their judgments, which leads to forecast inaccuracy and larger implications on the organizational planning and high-level forecast. (Bohanec et al.; 2017c) As Yan et al. (2015) state, in some cases opportunities would be intentionally underrated by sellers to avoid unwanted attention from internal competition, whilst some opportunities would be intentionally overrated due to the pressure from sales management to meet performance or forecasting metric standards.

Automation of opportunity win-propensity computation for business-to-business (B2B) sales is vital for a number of reasons: achieving better sales productivity and sales go-to-market alignment (Lawrence et al.; 2010), better planning (Lu and Kao; 2016), higher efficiency and sale prioritization (Duncan and Elkan; 2015; Yan et al.; 2015), more effective alignment of resources (D'Haen and Van Den Poel; 2013), understanding of the driving attributes behind successful sale (Bohanec et al.; 2017b). Based on the outlined factors, it is clear, that improvement in win-propensity accuracy will increase effectiveness of sales opportunity management and provide cost- and time-saving solution, making process more streamlined operationally and mitigating risk of late delivery (slippage).

Based on extensive model research, model ensemble stack involving boosting, trees, random forests, and neural nets was concluded to be a most optimal solution. Standalone, Adaboost, Random Forest, and C5.0 Tree algorithms were identified as top-performing for the purpose propensity-scoring. This project will discuss related works to the topic, methodology, implementation, and design of a proposed model stack solution using CRISP-DM framework (Chapman et al.; 2000), as well as some discussion on future works.

2 Related Work

This section is going to discuss related works for computing sales opportunity win-propensity score for B2B sales environment. Research in this specific area is still quite scarce, however some of the approaches utilised in clinical studies, marketing, and even time-series forecasting have been relevant for this research project.

2.1 Time-Series Forecasting

Time-series approach is more researched in the area of forecasting of B2B and B2C sales, and has seen a number of interesting developments, going beyond traditional ARIMA and regression techniques. Neural networks proven effective for sales predictions and outperforming ARIMA models. (Tkac and Verner; 2016) In contrast, B2B computer software sales (Lu and Kao; 2016) and fast-paced B2C environments, such as fashion retailers (Yu et al.; 2011; Xia et al.; 2012) found ELM model faster, more efficient and less computationally complex than neural network models, more suitable for real-time time-series

forecasting. To further improve model accuracy, MARS - multivariate adaptive splines, has been identified as a better method to select applicable variables for prediction and provided more accurate performance than neural network's back-propagation (Lu et al.; 2012) Some of the recent time-series works promote use of ensemble of machine learning methods to improve predictability and model performance.(Lu and Kao; 2016; Lu; 2014; Gurnani et al.; 2017) Gurnani et al. (2017) research demonstrated that fusion with XGboost made ARIMA more robust to the non-linear data features, high dimensionality, trends, and seasonality, resulting in higher accuracy than neural networks and SVM. (Gurnani et al.; 2017)

2.2 Propensity Modelling: Sales-Related

Very little research has been carried out to date within predictive modelling for sales pipeline in general and especially within win-propensity field (Yan et al.; 2015; Bohanec et al.; 2017a). According to Bohanec et al. (2017a) majority of research that has done demonstrates little evidence of successful business adaptation. Lawrence et al. (2010) development of OnTARGET and MAP solutions to identify new sales opportunities and align the resources to them via a propensity modelling carried out for implementation in IBM. However Lawrence et al. (2010) do not provide evidence of which algorithms were used, no details on modelling solutions, and how model accuracy was evaluated. In fact, evaluation is based on the assessment of changes within indirectly-related metrics, such as pipeline and revenue growth and quota attainment. Changes in these metrics could be down to a number of external factors outside of propensity automation impact.

D'Haen and Van Den Poel (2013) developed a three-phase method for sales customer acquisition, inclusive of a propensity-based solution in the second phase. Second phase of this approach consisted of logistic regression, decision trees, and neural networks methods used to calculate weighted propensity outcome if a prospected lead should become a sales opportunity or not. However, even though authors are calling for a fully automated solution to the prospect list generation, model-generated list was assessed manually by the company representatives, separating it into "good" and "bad" leads, rather than building customised domain-specific criteria into the modelling process.

Similarly, Duncan and Elkan (2015) developed two propensity-based methods, DQM and FFM, to automate marketing lead conversion into a sales opportunity leveraging its win-propensity, thus taking into account not only lead conversion propensity, but a sales opportunity win-propensity as well. Authors are using data from *Salesforce* system, as would this research, calling out a need of a large historical dataset, thought not quantifying size exactly, and singling out the following *Salesforce* features as crucial to the modelling: industry, customer company size, company market value, geographical location.

Yan et al. (2015) published research directly related to generation of win-propensity for sales opportunities. The method is based on the two-dimensional Hawkes dynamic clustering process and is tackling main limitation of the active sales pipeline - its dynamic nature. Sales pipeline expands and contracts during course of the quarter - slipped opportunities come in from previous quarters, deals from future quarters are brought in early, newly created opportunities get to close within the given quarter. This approach is allowing live assessment of opportunity propensity within active pipeline rather than training models on historical data standalone. However, is mainly based on variables generated from salesperson's interaction with the opportunity and updates to the customer profile.

Thus, the approach is directly reliant on absence, presence, and frequency of updates to the opportunities. This requires regular time windows when exactly modelling should take place and additionally, does rely on regular, uniformed, and disciplined interaction of a salesperson with a CRM system.

As correctly pointed out by Tang, L. and Xu, X. and Rangan, V. (2017) due to the high volume of the opportunities being managed at one time by a single salesperson, the updates to the customer profile or opportunity itself might not be consistent or regular. In fact, quarter-end period would be especially problematic, as opportunities can be updated and closed few days after quarter ends, as main volume of opportunity closure usually falls within last couple of weeks of the quarter (Tang, L. and Xu, X. and Rangan, V.; 2017) This irregularity and untimely updates to the opportunity, can create bias and affect the propensity outcome using Yan et al. (2015) methodology.

Tang, L. and Xu, X. and Rangan, V. (2017) developed a modelling forecasting engine, which interlinks CRM and data storage solutions, allowing for almost life modelling on-demand. Forecast engine consists of a server-run model library with time-series, neural network, probability, and hybrid models to tackle time-series forecast and win-propensity, relying heavily on large amount of historical data analysis to produce win-propensity for current active pipeline. Win-propensity models consist of logistic regression, gradient boosted decision trees and long-short-term memory. Win-propensity models in Tang, L. and Xu, X. and Rangan, V. (2017) research require historical snapshots of data to operate, and are computationally more expensive than simpler probabilistic models, though provide higher accuracy and robustness to data quality.

Tang, L. and Xu, X. and Rangan, V. (2017) goal is to standardise the solution across multiple companies as customers, and therefore, variable selection and feature engineering are not extensive. Though some variables are singled out due to consistent significance for predictability, such as close dates and quarters, opportunity age, opportunity revenue size, seller's assigned forecast category, geography, etc. This research outlines direct relationship between win, age, and amount: larger opportunities take longer to close.

Furthermore, Tang, L. and Xu, X. and Rangan, V. (2017) solution is outlining one specific problem with the dual nature of forecasting within sales. That is use of win-propensity scores to aggregate the overall quarterly forecast. Win-propensity of opportunities can be utilised to forecast at the opportunity level, but should not be aggregated to provide a time-series or overall "forecast number". As authors themselves state, around 50 percent of the opportunities get created and closed within the given fiscal quarter, which means multiplication of win-propensity scoring by revenue size and aggregation to overall forecast will never provide a true forecast at a given time during the quarter. Therefore, there is a clear gap between win-propensity opportunity-level forecasting and higher level, time-series forecasting. Leverage of win-propensity for time-series in such dynamic environment is still a topic to be researched.

Deep learning method of Restricted Boltzmann Machines (RBM) in conjunction with sufficient statistics (SS) feature proved robust enough to time-variant sales opportunity nature and data quality challenges. (Zhang et al.; 2014) SS with RBM performed well for win-propensity calculation for sales pipeline; same approach is used for contour detection in images.

All of the mentioned works on sales win-propensity or works closely related to the topic leveraged complex modelling approaches and architectures, often customized solutions, with some core covariates, such as industry, time dimensions, i.e. close dates and opportunity age, geography, revenue size. Majority of discussed research outlined demand for

large and wide datasets. Tang, L. and Xu, X. and Rangan, V. (2017)'s research quantifies general variable span of 200-400 variables, with 50-100 actively updated. Challenges, such as large amount of variables, poor data quality, large amount of missing values, dominance of categorical variables over continuous, high dimensionality in conjunction with complex and dynamic nature of the pipeline requires customised machine learning solution.

2.3 Propensity Modelling: Clinical Studies

Clinical industry heavily leverages propensity score for treatment effect estimation and assessment of treatment groups. Unlike sales, it boasts an extensive research within the area of propensity calculation using machine learning. Same as within sales, datasets often have large amount of missing data (Zhao et al.; 2016), skewed data (Linden and Yarnold; 2016), corrupted-labels data (Wang et al.; 2018). Unlike sales, the environment is static rather than dynamic, and datasets are usually small, as its relevant to the size of the treatment groups. (Westreich, D. and Lessler, J. and Funk, M.J.; 2010)

Propensity modelling is naturally a classification task, usually binary, and therefore, logistic regression method was most popular solution to it.(Westreich, D. and Lessler, J. and Funk, M.J.; 2010) With the recent developments in machine learning research, application of logistic regression to this problem is diminishing for a number of reasons. Logistic regression suffers from sensitivity to the data quality (Wang et al.; 2018), does not handle missing data well (Zhao et al.; 2016), has risk of being affected by skewed data and therefore, developing bias in predictions (Linden and Yarnold; 2016). According to Westreich, D. and Lessler, J. and Funk, M.J. (2010) logistic regression models is not an ideal choice for propensity calculation due to sensitivity to data quality and presumption of linear relationship within data. In addition, Zhao et al. (2016) demonstrates that Mahalanobis distance utilised in regression, works well for continuous variables, but not for categorical and discrete variables.

Zhao et al. (2016) recommends use of Random Forest algortihms for propensity scoring as alternative to logistic regression. Random Forests have propensity principle built into model design and therefore, are part of model output. This algorithm is not dependant on the Mahalanobis distance, nonparametric in nature, and robust to missing data even with forty percent missing values per variable, handling mixture of categorical and continuous variables well. (Zhao et al.; 2016) According to Zhao et al. (2016) Random Forest treats all variables as discrete, and therefore would be a good model to use in dataset with discrete variables. Zhao et al. (2016) and Caruana and Niculescu-Mizil (2006) researches demonstrates that Random Forest often surpasses other high-performing nonparametric models in accuracy, including artificial neural networks, boosting, and SVM.

Earlier work of Westreich, D. and Lessler, J. and Funk, M.J. (2010) recommends use of CART trees and meta-classifiers (boosting) techniques for propensity scoring, which are capable of providing explicit probabilities and are less bias in its results than logistic regression approach. Wang et al. (2018) use of boosting method, XGboost combined with spectral clustering to compute propensity scores provided excellent performance scores even for a dataset with 40 percent corrupted labels.

Use of optimal discriminant analysis for win-propensity also demonstrated robustness to data quality and distribution of treatment groups for multivaled treatments. Monte Carlo permutation in the ODA allows for assessment of statistical siginificance of propensity results.(Linden and Yarnold; 2016) ODA nonparametric nature proved robust

to skewed data and outliers. (Linden and Yarnold; 2016)

2.4 Related Work Conclusions

Majority of sales-related work is focused around time-series research. Research on sales propensity scoring is seldom, however some of the discussed approaches in marketing and clinical studies areas can be adopted. This includes use of decision trees and ensemble methods such as Random Forests and boosting.

Discussed data challenges, inclusive of missing and corrupted data, large amount of variables, in its majority categorical and discrete, high-dimensionality in conjunction with dynamic nature of sales opportunities and change in factors, such as new product launch (Tang, L. and Xu, X. and Rangan, V.; 2017), require optimum machine learning solution to be robust to listed data challenges and simple enough in further optimisation and adaptation.

Related works demonstrated strong preference towards ensemble methods and super learners (Zhao et al.; 2016; Wang et al.; 2018), multi-stage modelling designs involving a number of algorithms for clustering to neural networks (D'Haen and Van Den Poel; 2013) due to discussed data and pipeline challenges. Some research resulted in the development of complex model libraries (Lawrence et al.; 2010; Tang, L. and Xu, X. and Rangan, V.; 2017). This work is going to search for a simpler solution, easier in adaptation. The above works do not indicate use of customised model-stacking and this is the approach this research project is going to employ.

3 Methodology, Design, and Solution Development

The dataset used for this research project has been provided by an IT Software company, and is an extract from *Salesforce* system. CRISP-DM framework was applied to the research (Chapman et al.; 2000; Azevedo et al.; 2008) using the following steps:

- Business Understanding
- Data Understanding
- Data Preparation
- Modelling
- Evaluation and deployment

3.1 Business Understanding

According to Chapman et al. (2000) business understanding is the first step of the CRISP-DM process, which involves clear understanding of the objective and the data mining problem from a business perspective. Objective of this research is to apply a machine learning solution to automate B2B sales win-propensity. Background to the research problem has already been provided.

To revisit: currently, assignment of the propensity score is a manual and subjective process, carried out by a salesperson, usually loosely based on some business operational guidance. This is not an accurate assessment of probability of a sale (opportunity) closure

and leads to forecast inaccuracies. Salesperson in a relatively large organization can manage hundreds of active opportunities at one time, at various stages of the sale. It is natural that the assessment of propensity for those opportunities is irregular, not consistent, and bias - as it is further affected by the pressures from internal competition and sales management. Accurate and objective automated win-propensity score will achieve a more streamlined sales management model, cost- and time- saving, help to prioritise sales resources based on the closure probability of the opportunities and decrease potential sale loss.

As per CRISP-DM framework (Chapman et al.; 2000), next step is the understanding of available data, variables, data quality issues, which would be covered in the next section.

3.2 Data Understanding and Feature Engineering

This subsection is going to provide a better insight into *B2B Sales Dataset*. A number of related works mentioned the importance of selecting a large dataset due to data quality issues, missing data, and modelling processes. (Duncan and Elkan; 2015; Tang, L. and Xu, X. and Rangan, V.; 2017) Selected main dataset is quite large and wide, detailed definition of each field can be viewed under in the table 1 in section 6: Appendix.

The following datasets were used for this research project:

- Main: *Thesis Main 1 Input B2B Sales Dataset [B2B Sales Dataset]*

  This is a .csv export from *Salesforce* system at the courtesy of an IT software company. Dataset was anonymised and each categorical variable is assigned with a numeric code and a grouped field with code and corresponding text. This dataset was enhanced with additional features from secondary datasets and feature engineering.

- Secondary: *Thesis Secondary 2 Input Accounts*

  .csv extract from *B2B Sales Dataset* of Account ID, Grouped ID, Account Name and Grouped Account Name for industry labelling purposes. Industry variable was joined with *B2B Sales Dataset*

- Secondary: *Thesis Secondary 3 Input Product Matrix*

  .csv feature-engineered binary product matrix for each product family of products per opportunity and total product count, was joined with *B2B Sales Dataset*.

- Secondary: *2015 Fortune 1000 List with Industry Website* (Rudis; 2016)

  This is an external secondary dataset containing a list of Fortune 1000 companies and its industry description, used to enhance industry variable with *Accounts* dataset via application of *fuzzyjoin* techniques (Robinson and Elias; 2018) . Industry variable then was carried over to main *B2B Sales Dataset*.

B2B Sales Dataset consists of 69 columns with 148,199 individual sales opportunities and 670,680 missing values between all the variables. Out of 69 columns, 7 columns are brought in via joining tables with the *Product Matrix* for binary count of each product family out of 6 categories and total product count for each opportunity. Additionally, industry code is added from the *Accounts* table. For process flow on joining datasets, please view figure 12 in section 6: Appendix.

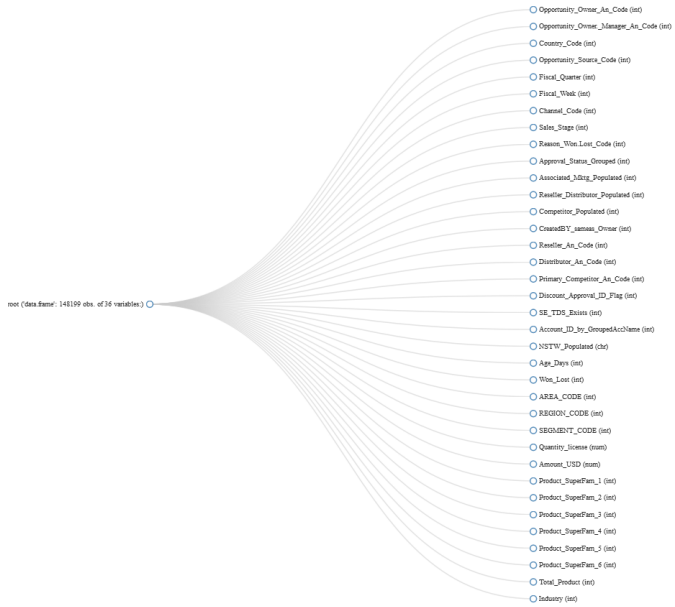


Figure 1: Variable Correlation Plot

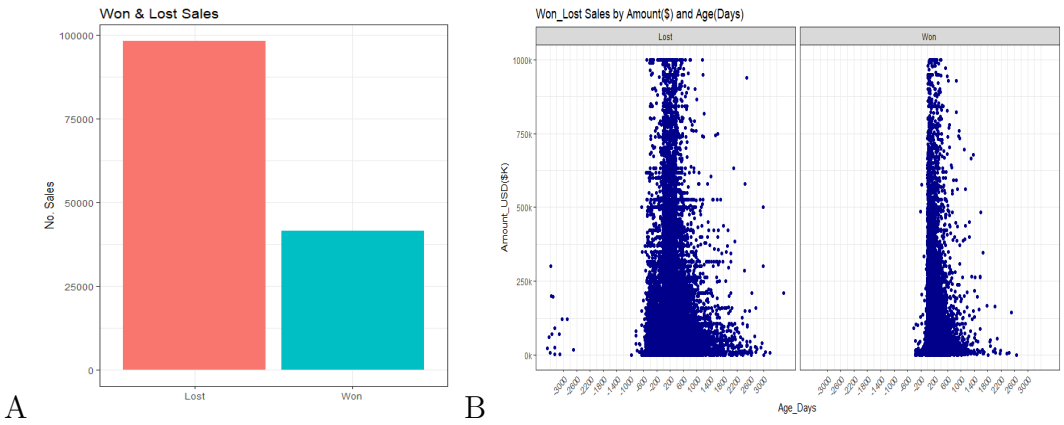
Industry code has been feature-engineered and grouped into 17 different customised categories, such as "Government" or "Banking/Finance/Insurance" or "Energy/Fuels/Utilities" using a combination of strategies:

- manual labelling of customer ("Account Name") to corresponding industry based on business knowledge
- use of a mixture of fuzzyjoin techniques, such as soundex and Jaccard distance measurement (Robinson and Elias; 2018) to match customer names with a Fortune 1000 list and extract its industry information, additionally labelling this industry information into customised dataset industry categories
- use of keywords such as "ministry" or "hospital" within a text string of a customer name to assign the industry leveraging *grepl - Rbase* function. (Bates et al.; 2018)

The combination of listed techniques allowed to populate a newly created industry field for 49 percent of the dataset.

Additionally, during the process of compiling the *B2B Sales Dataset*, variables containing sensitive information were anonymised using various data-generator tools.(Bailey and Bailey.; 2013; Brocato; 2018; Keen; 2005) For example, names of opportunity owners and managers were replaced, product names were removed, and re-coded into mentioned above *Product Matrix*. Sales notes from "Next Step to Win" field were removed and replaced with a binary variable "NSTW Populated" to indicate if the field "Next Step to Win" was populated by the salesperson or not. In addition, every categorical variable such as country, opportunity owner, approval status, etc. was assigned a numeric code corresponding to a unique value from each variable to keep its distribution in tact. These updates were carried out field-by-field basis.

The only field which could not have been anonymised straight away was "Account Name", as it was leveraged for labelling industry. Hence why, once the industry variable



A B

Figure 2: A.Won/Lost Sales/ B. Sales by Age

was compiled, field was removed out of the dataset, leaving in tact its corresponding numeric IDs only.

Once the dataset was pushed together, categorical variables were removed, leaving behind its corresponding numeric code as a variable. Some duplicate variables and observation IDs, were removed. As per breakdown in figure 1.A , 36 variables and 148,199 observations remained to undergo initial analysis and further data cleaning process.

3.3 Data Preparation for Modelling

According to Chapman et al. (2000) next stage is data preparation, which entails all activities necessary to prepare dataset for modelling, for example: data cleaning, feature selection, and pre-processing.

3.3.1 Initial Analysis: Exploring Variables

This subsection is going to briefly review major independent variables in relation to the dependent variable: `"Won_Lost"`, where 0 indicates sales opportunity with status "Lost" and 1 indicates sales opportunity with status "Won". Sales opportunities further could be related to as "sales".

Out of 148,199 opportunities 41,543 are won (28 percent of total) and 98,277 sales are lost (66 percent), for 8,379 (6 percent) status is unknown. Missing values in this case means that opportunity had an active open status, which is not possible in a historical dataset: all opportunities with close date in the past should either be "Won" or "Lost". Therefore, observations with missing values based on variable `"Won_Lost"` should be excluded from further analysis. Sales by `"Won_Lost"` statuses can be observed in the figure 2.A.

Naturally, there always would be more lost opportunities than won. It is important to note, that as the opportunity progresses through sales stages more fields are becoming compulsory for a salesperson to populate. This naturally makes opportunities closer to status "Won" "cleaner", containing less missing values. As a lot of opportunities are lost at early sales stages it has a knock-on effect when preparing data for modelling. Data preparation for modelling process results in eliminating observations with large portion of missing values, and hence the 30:70 proportion shifts towards 40:60 won to lost.

Moving on to figure 2.B it can be observed that some sales have negative age, which is

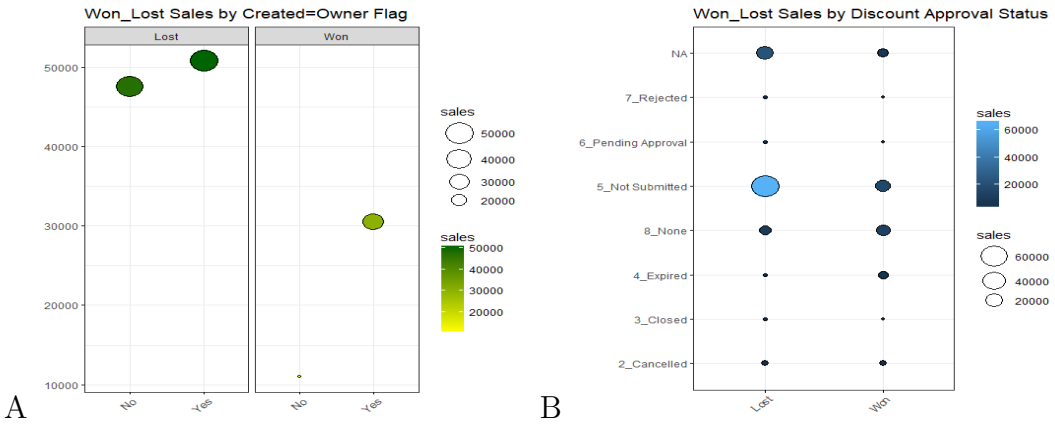


Figure 3: A.Won/Lost by Created=Owner/ B.Sales by Amount

not possible, as it means that close date of the opportunity is earlier than date of creation. This can only be the case if a sale was logged into CRM system with backdated closure. It is not part of the normal process, and therefore, opportunities with negative age should be removed from further analysis. Tang, L. and Xu, X. and Rangan, V. (2017) have observed a clear relationship between age and amount: larger opportunities take longer to close. However, that finding is not supported here. As can be seen from figure 2.B as value of opportunities grows, age remains almost the same, and in fact smaller value opportunities take longer to close than the majority of the larger ones.

Additionally, as per figure 3.B there does not seem to be a large amount of sales won because they were discounted. Majority of won sales have no discount status, discount was not submitted for approval, or no discount is present. Lastly, as can be seen from figure 3.A a significant amount of won sales has been conducted and created by the same salesperson, rather than created by marketing, handed over from channel (reseller or distributor) or handed over from other salespersons. In this case based on the flag indicating that opportunity creator and owner is the same person.

3.3.2 Data Cleaning

Firstly, any records with missing values in a dependent variable: "Won Lost" were removed as per previous section discussion. Secondly, duplicate of a dependent variable - "Sales Stage" and "AREA" as variable containing aggregation of "Country" variable were also removed. Furthermore, any observations without product count, amount, age and customer were excluded. Opportunity owner, manager, and account id corresponding variables were removed - in order not to train the model on predicting personal performance or win of sale based on a specific customer.

After assessment of missing values as per figure 4.B the decision was made to remove distributor and reseller variables as well, as 72 percent of reseller and 87 percent of distributor names are missing from the dataset, and in majority of cases were simply not populated by the salesperson rather than meaningfully missing.

Finally any observations with over 19 percent missing values were also removed. For the purposes of the study, rather than using missing value imputation, the decision has been made to use clean data set with no missing values. As in majority of cases, variables with a missing values do not have a meaning of absence, but rather means that fields were not populated by the salesperson.

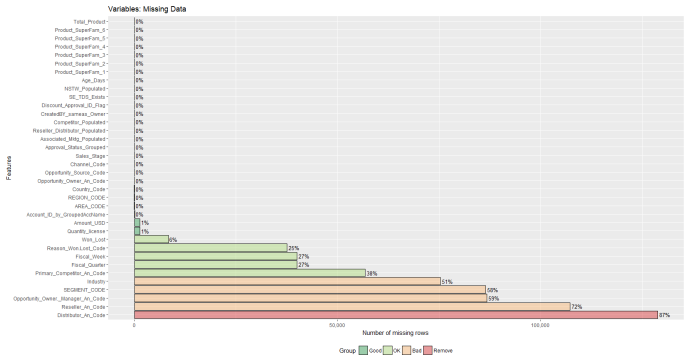
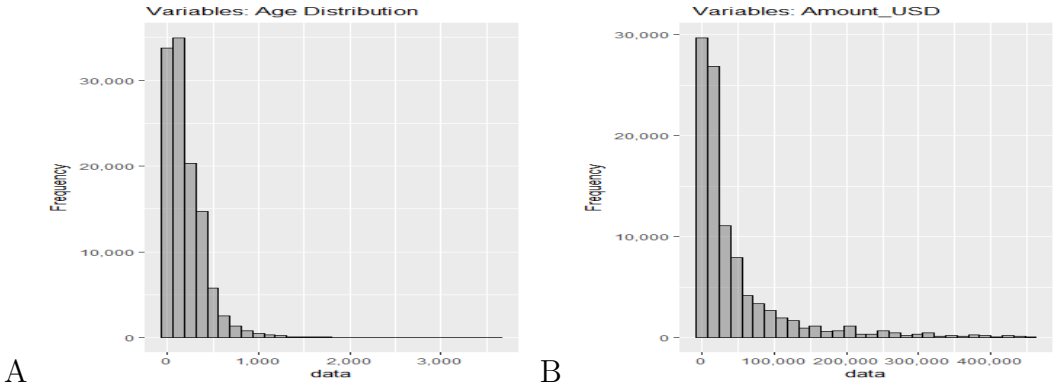


Figure 4: Variable Correlation Plot

Negative values and outliers were removed from the continuous variables "Age" and outliers removed from "Amount USD", the distribution was skewed to the left, as can be seen in figure 5 A and B.

As discussed, no strong relationship between age and amount was observed in contract to Tang, L. and Xu, X. and Rangan, V. (2017)'s findings. In this case, figure 6 only shows a weak positive significant correlation of less than 0.2. In fact, there is a stronger negative correlation of 0.45 between "Age Days" and dependent variable "Won Lost", shows that older the opportunity, less chance it is going to be a win.



A B

Figure 5: A. Age Distribution / B. Amount USD Distribution

Figure 6 shows correlation plot with only statistically significant correlations based on p-value. No strong correlations are observed. Moderate positive correlation of 0.4 between "SE TDS" and "Amount USD" fields shows that involvement of pre-sales engineering team to the sale allows increase in sales revenue. Correlation of .46 observed between "Country Code" and "Region Code", as Region is a hierarchical roll-up of "Area Code", which is in turn a roll-up of "Country Code". When tested on modelling, exclusion of variables "Created BY same as Owner", "Region Code" AND "Discount Approval ID Flag" had a negative effect on the model accuracy, and therefore were left in tact.

3.3.3 Further Data Pre-Processing and Feature Selection

As an outcome of initial data assessment and cleansing process, it became clear that there are a number of complexities requiring further data processing for modelling: dominance of binary and categorical variables, distribution of the numeric continuous variables of "Age Days" and "Amount USD" are skewed to the left, some variables retained in the

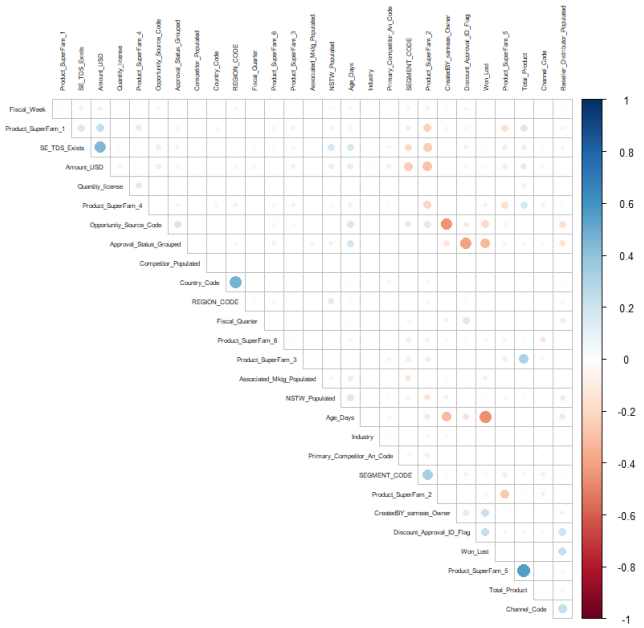


Figure 6: Variable Correlation Plot

dataset are moderately correlated with each other, however their removal is resulting in negative effect on the accuracy of models: "Discount Approval ID Flag", "Region Code", and "Created BY same as Owner".

A number of approaches to further optimise dataset performance were tested and put through selected models: normalising the variables, one-hot encoding the variables, and finally, normalising and one-hot encoding the variables. (Prabhakaran; 2018; Kuhn; 2018) Based on the modelling outcome of ROC method, the most optimum approach was selected - normalising variables.

Furthermore, due to large amount of preserved covariates (27), feature selection techniques were explored. As argued by Li et al. (2017) feature selection methods that are specific to the algorithm, usually with use of filter and wrapper elements, are more efficient and accurate. Therefore, variables were assessed via use of variable importance functionality of *caret* package (Kuhn; 2018). Varimp function allows for feature assessment based on the specific model used, hence why is selected as a method. Kuhn (2018) Whist MARS variable importance is based on GCV method, Random Forest classification method is more wrapper-style, and tree methods are more filter-style, based on the assessment of variable weights from how the feature is positioned in the tree splits. For example, if the feature is positioned in the first split of a tree - its more important than feature in the second split etc. Kuhn (2018) Variable importance can then be plotted ranked by feature importance, as shown in figure 7 for every model used when ran standalone, variable importance is scaled between 0 and 100 to make it cross-comparable between models.

Variable importance functionality is currently not applicable to the model stack approach (Kuhn; 2018; Deane-Mayer and Knowles; 2016) Therefore each model used in the model stack was run standalone to acquire variable importance plotted and analysed as per figure 7.

Based on the variable importance assessment for each model, as per figure 7, 10 top variables used by every model were put through the model stack: "Age Days", "Ap-

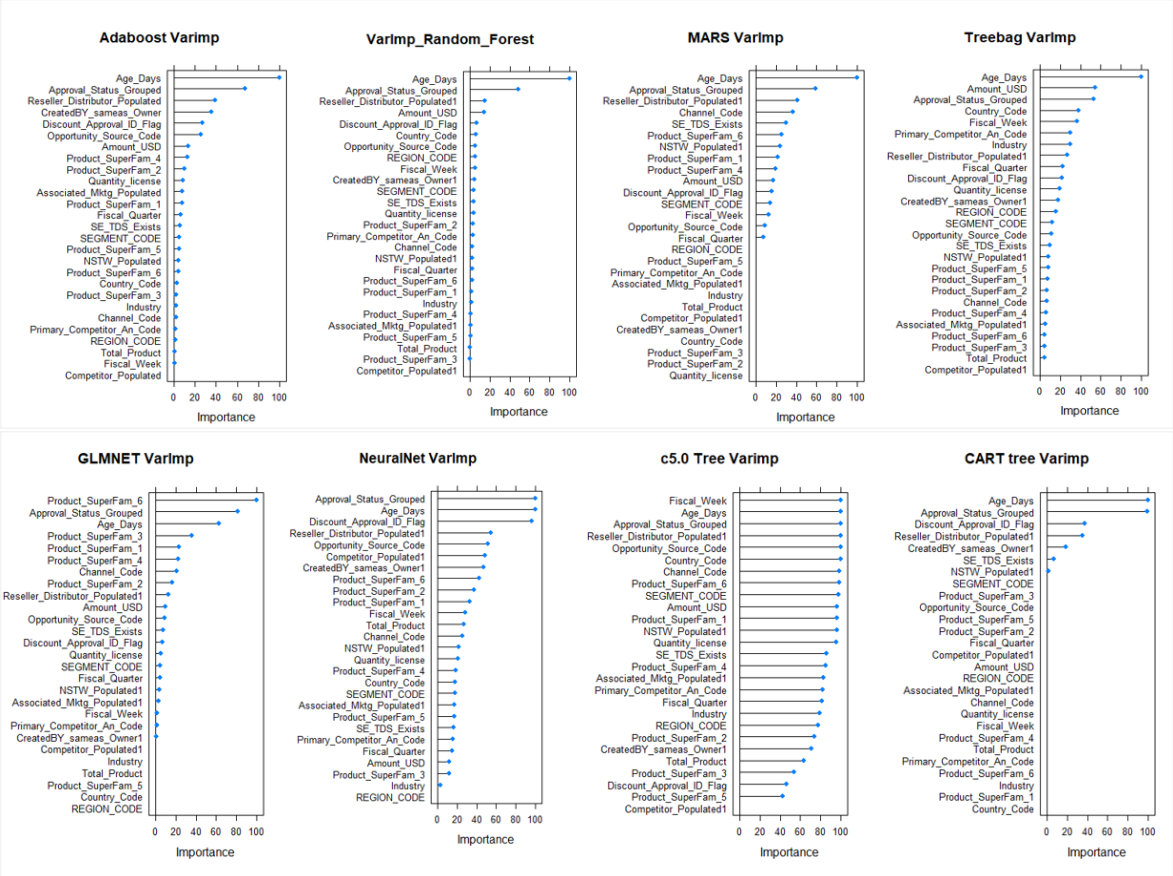


Figure 7: Variable Importance

proval Status Grouped", "Reseller Distributor Populated", "Amount USD", "Channel Code", "Discount Approval ID Flag", "SE TDS Exists", "NSTW Populated", "CreatedBY sameas Owner", "Opportunity Source Code". The performance of the model stack have not been improved in comparison to when all 27 variables were used. Then feature selection was expanded to 17 variables, inclusive of the above ten, and was trialled via same model stack as used previously: "Age Days", "Approval Status Grouped", "Reseller Distributor Populated", "Amount USD", "Channel Code", "Discount Approval ID Flag", "SE TDS Exists", "NSTW Populated", "CreatedBY sameas Owner", "Opportunity Source Code", "Product SuperFam 1", "Product SuperFam 3", "Product SuperFam 6", "Product SuperFam 4", "Fiscal Week", "Associated Mktg Populated", "SEGMENT CODE". The performance of the model stack still has not been improved.

Nevertheless, while the variable importance is the optimum solution when using the model standalone, in case of the model stack it did not improve the accuracy of the model, and therefore, all 27 variables were left in tact.

Chapman et al. (2000) CRISP-DM guide states that data preparation is often a step to which the analyst repeatably goes back whilst modelling to optimise model performance. As per already discussed examples of testing variable selection, different pre-processing methods, such as normalisation and one-hot encoding, this was exactly the case in this research. Changes to data preparation stages were made in accordance to the improvement within models used. Next section is going to specifically focus on modelling process and what type of models were utilised.

3.4 Modelling: Choice of Algorithms

Based on the related works, a number of algorithms was selected to predict the win-propensity score for *B2B Sales Dataset*. In this case, models would solve for a classification binary problem, where the opportunity is either "Won" (1: 100 percent probability of winning a sale) or "Lost" (0: 0 percent probability of winning a sale). The propensity is the by-product of the model output, conditional probability of a sale having status 1 ("Won"). This section is going to discuss models applied in the context of application to the desired propensity score with a taste of model accuracy during model training, what related works driven the selection of models, and methodology of a final solution used. Summary of model training results can be viewed in the table 8A and figure 8B.

3.4.1 GLM

This research is dealing with a binomial classification problem, and logistic regression is a classical approach to its resolution. Therefore, first chosen "base" solution is a *Generalised Linear Model (GLM)* (Kuhn; 2018). Historically, GLM is a the most popular solution used for calculation of the propensity score (Linden and Yarnold; 2016; Westreich, D. and Lessler, J. and Funk, M.J.; 2010) Logistic regression's conditional probability of fitting the point is the propensity output. Logistic regression algorithms, such as GLM, are linear in nature, which often makes propensity scores obtained naive and bias (Westreich, D. and Lessler, J. and Funk, M.J.; 2010), sensitive to data quality, especially missing values, and does not deal well with high-dimensionality and categorical variables, as is based on the Mahalonobis distance, which deals better with continuous variables (Zhao et al.; 2016). Hence, encoding the categorical variables into the numeric made use of GLM possible in this research as well as in the research of D'Haen and Van Den Poel (2013). GLM is used as a base method, and in this case, ROC scores produced are 83.44 percent during the training process and 85.48 percent during training as part of *CaretList* (Deane-Mayer and Knowles; 2016).

3.4.2 C5.0 Tree, CART, and Treebag

Decision trees are easily interpreted and are a popular solution to the propensity calculation within medical studies (Westreich, D. and Lessler, J. and Funk, M.J.; 2010). Trees provide clear visibility on how exactly propensity was calculated via branching, what variables are involved and their importance (by order of splits), have great performance. Especially popular are Classification and Regression Trees or CART model, which is a bagging tree solution, capable of the propensity output as part of the model design (Westreich, D. and Lessler, J. and Funk, M.J.; 2010). Treebag is an alternative bagging method offered by *caret* package that might work for this purpose (Kuhn; 2018). Model transparency is one of the most important factors for successful business adaptation discussed by previous research (Bohanec et al.; 2017b,c; Tang, L. and Xu, X. and Rangan, V.; 2017).

As a result, C5.0, CART, and Treebag models (Kuhn; 2018) were chosen in this research to calculate propensity score. C5.0 Tree achieved highest results standalone scoring 87 percent during model training, but dropped to 83.51 percent when trained as part of *caretList* /citepweb:caretensemble2018. Treebag's accuracy standalone is 85.84 percent, whilst as part of /textitCaretList function (Deane-Mayer and Knowles; 2016) - 86.22 percent. Finally, performance of CART is least accurate of all trialled models,

standalone: 79.80 percent, as part of *CaretList* (Deane-Mayer and Knowles; 2016): 80.43 percent.

3.4.3 Random Forest

Random Forest is an ensemble method based on regression and classification decision trees, requires little tuning. (Zhao et al.; 2016). The conditional probability of a propensity-score, by the design of the Random Forest algorithm is calculated and stored within its terminal nodes (Zhao et al.; 2016), during model run. This allows to export this conditional probability as an output using *caret* package, by specifying desired output as probability (Kuhn; 2018), thus avoiding additional calculations of the propensity score. In addition, as discussed in section 2, the method has been selected due to it robust nature towards discrete and categorical variables, non-parametric nature, robustness towards data quality issues; and performance that often surpasses other non-parametric algorithms with a more computationally expensive nature (Zhao et al.; 2016). Indeed, algorithm achieved better accuracy than all other algorithms during training, scoring 87.54 percent during training standalone, and 88.30 percent when run as part of the *CaretList* package (Deane-Mayer and Knowles; 2016).

3.4.4 MARS

Multivariate adaptive regression splines (MARS) has not been used within reviewed works as a main algorithm for propensity calculation. As per discussed related works MARS has only been applied within a time-series method for variable selection purposes (Lu; 2014). MARS is a non-parametric model, that has excellent variable selection capabilities and is able to resolve nonlinear regression of high complexity (Lu; 2014) MARS in this case performed slightly better than Neural Networks during training - 86.05 percent, while Neural Networks were 85.97 percent, and training as part of *CaretList* allowed for further improvement in accuracy to 86.99 percent, bringing MARS up into top three performing models when trained as part of *caretList* (Deane-Mayer and Knowles; 2016).

3.4.5 Neural Networks

Neural Networks has been employed in a number of of propensity works due to its robustness to data quality, sufficient complexity, and ability to deal with high dimensionality (Westreich, D. and Lessler, J. and Funk, M.J.; 2010). However, optimizing neural network for the purpose might take extensive amount of time, to avoid overfitting (Westreich, D. and Lessler, J. and Funk, M.J.; 2010). D'Haen and Van Den Poel (2013) uses neural network within a multi-phase approach as a final algorithm to generate a weighted list of sales prospects from existing and new sales customers. Gurnani et al. (2017) uses neural networks in the time-series forecasting approach. Due to its backpropagation nature, it might be computationally expensive, and slower than model such as ELM (Yu et al.; 2011). Neural network in this research has achieved a ROC score of 85.97 percent standalone and 86.22 percent as part of the *CaretList* (Kuhn; 2018), which puts it behind Random Forest, Adaboost, and MARS solutions.

A

TRAIN

Model Type	ROC Model Standalone	ROC_caretList	ROC_caretStack_ GLM	ROC_caretStack_ RF	ROC_caretStack_ Adaboost
RandomForest (ranger)	87.54%	88.30%			
Adaboost	87.26%	88.12%			
C5.0Tree	87.00%	83.51%			
MARS (earth)	86.05%	86.99%			
NeuralNetworks (nnet)	85.97%	86.52%			
Treebag	85.84%	86.22%			
GLM (glmnet)	83.44%	83.76%			
CART (rpart)	79.80%	80.43%			
avg (standalone) vs stack (actual)	85.36%	85.48%	88.21%	88.18%	86.27%

Best Performing	RandomForest: 87.54%	RandomForest: 88.30%	caretStack_GLM: 88.21%

B

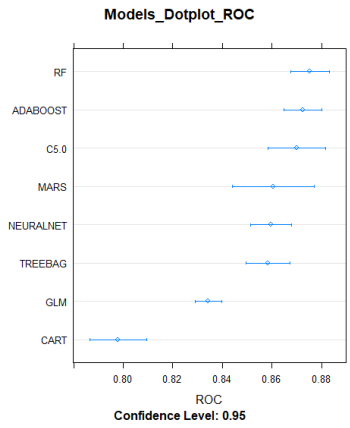


Figure 8: E. Model Training Results/ F. Training Standalone: Dotplot

3.4.6 Adaboost

Boosting trees and other boosting solutions were used in related works for calculating sales win-propensity by Tang, L. and Xu, X. and Rangan, V. (2017) - Gradient Boosted Decision Trees and in a time-series approach by Gurnani et al. (2017), where a hybrid of XGBoost and ARIMA gave better performance than other solutions. Furthermore, boosting proven effective for win-propensity calculation in a number of medical research papers (Westreich, D. and Lessler, J. and Funk, M.J.; 2010; Wang et al.; 2018), with the ability to deal with up to 40 percent label-corrupted data without significant deprivative effect on the model accuracy Wang et al. (2018). Boosting combines weaker learners into a single super-learner solution (Wang et al.; 2018) and produces probabilities of class membership (Westreich, D. and Lessler, J. and Funk, M.J.; 2010). Previous research demonstrated that trees and boosting algotithms are performing better than logistic regression (Wang et al.; 2018), therefore, Adaboost Classification Trees algorithm was selected for this research (Kuhn; 2018). Adaboost performed with accuracy second only to Random Forest, when trained standalone: 88.26 percent and as part of *CaretList* function (Kuhn; 2018) scoring 88.12 percent versus 88.30 percent for Random Forest. However, during model testing Adaboost accuracy surpasses Random Forest.

3.4.7 Model Stack

CaretEnsemble package provides an ability to create a customised meta-model, which is the approach applied in this research using *caretStack* function (Deane-Mayer and Knowles; 2016). A number of discussed related works use multi-model approach to compute propensity scores. Wang et al. (2018) argues that super learners and customised model ensembles can outperform other models for propensity calculation. D'Haen and Van Den Poel (2013) use a multi-phase process of KNN, logistic regression, decision trees, and neural networks to prepare sales prospect list with propensity weighting. Duncan and Elkan (2015) use DQL and MQL models for conversion of marketing leads into sales leads using propensity scoring. Tang, L. and Xu, X. and Rangan, V. (2017) use a whole library of models, including OSAMs - models responsible for win-propensity forecasting, inclusive of logistic regression, trees and LSTM. None of the reviewed works use model stack approach to calculate propensity score for sales. This research is showing that model stack approach does provide better results than standalone model for win-propensity scoring. During model training, model stack compiled from the above discussed models, produced

a ROC score of 88.21 percent second only to Random Forest - 88.30 percent. Model stack was trained with application of three methods: GLM, Adaboost, and Random Forest. Model stack with GLM method provided best results and this is the meta-model taken to the model training phase. It has proven its accuracy when tested, and surpasses all other discussed standalone models. Further details on the model performance would be discussed in section 4.

3.4.8 Model Training: Conclusions

As could be seen from testing results in table 8A - three top performing models during model training for B2B sales win-propensity are: Random Forest, Adaboost, and C5.0 when trained standalone versus MARS when trained as part of *caretList* function (Deane-Mayer and Knowles; 2016). While boosting (Wang et al.; 2018) and Random Forest (Zhao et al.; 2016) performance was given extensive credit for propensity calculation, reviewed related works show no evidence of using MARS standalone for sales win-propensity calculation.

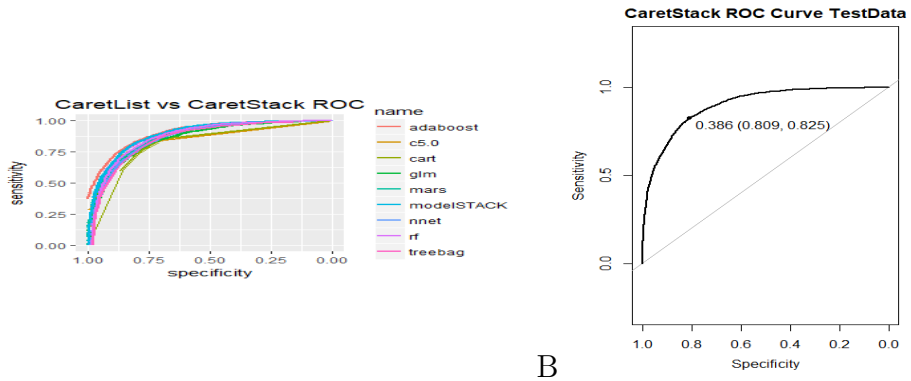
As per table 8A developed model stack using *caretStack* functionality with applied GLM method (*caretStack GLM)* achieved better performance during model training than most models (Deane-Mayer and Knowles; 2016), second only to Random Forest. However, model stack outperforms Random Forest during model testing process. All discussed models were put through training and, consequently, testing process. As per applied CRISP-DM framework (Chapman et al.; 2000; Azevedo et al.; 2008): evaluation of the modelling results should be thorough and are important for correct model choice to respond to business objective at hand. Therefore, evaluation process, results of the model testing phase, and deployment would be discussed further in the next chapter.

4 Evaluation and Results Discussion

4.1 Model Results and Evaluation Methods

As per Chapman et al. (2000) CRISP-DM framework, next step after modelling is evaluation of the modelling results. This step allows to assess model performance and select correct model solution for research objective at hand. This research project uses ROC method to evaluate model performance and accuracy. Consequently, its area-under-the-curve (AUC) metric, sensitivity, specificity. ROC is used in the number of the related works and is one of the most popular methods to measure model performance, hence why is employed in this research (Yan et al.; 2015; Bohanec et al.; 2017a,b; Tang, L. and Xu, X. and Rangan, V.; 2017; D'Haen and Van Den Poel; 2013).

As can be seen from the subfigures 9A and 9B, all models were measured using ROC method. However, AUC standalone is not an exhaustive measure, especially when, as per figure 9G accuracy of plotted models is quite close to each other. To review additional ROC metrics in detail and calculate misclassification rate, confusion matrices were run. Confusion matrices for best performing models can be viewed in table 13 in section 6: Appendix. Based on the information collected from ROC curves and confusion matrices table in the figure 10 was constructed, containing details of model training performance. Train data results were briefly discussed in the previous section, for further details on AUC for training phase please see table 8E. This section is going focus on the evaluation of results when trained models were tested.



A B

Figure 9: G. Model Testing ROC Plot/ F. caretStack ROC Plot

The table 10 shows the output of model testing. Table includes "Accuracy" (AUC), sensitivity (positive class is "Won") and "Misclassification Rate" metrics; the latter one calculated by dividing a number of misclassified observations on the total number of observations within the testing dataset. The table 9 is showing results for standalone model run ("Models Standalone"), model run as a bulk using *caretList* function ("CaretList") and finally, model stack results ("CaretStack GLM") using *caretStack* function from *caretEnsemble* package (Kuhn; 2018; Deane-Mayer and Knowles; 2016).

From the standalone model predictions, Adaboost algorithm was the most accurate: 81.27 percent accuracy, followed by C5.0 with 80.38 percent, and Random Forest with 80.35 percent accuracy, Treebag - 79.70 percent accuracy. In line with the argument from the related works discussed in section 2 and 3, ensemble algorithms such as Random Forest (Zhao et al.; 2016), boosting super-learners (Wang et al.; 2018), and various tree ensemble models (Westreich, D. and Lessler, J. and Funk, M.J.; 2010; Tang, L. and Xu, X. and Rangan, V.; 2017) proven effective for propensity score computation not only in clinical studies, but also in B2B Sales environment for IT software.

TEST

Model Type	Models Standalone				CaretList				CaretStack_GLM			
	Accuracy	Sensitivity	Total Misclassified (no.)	Misclassifcation Rate	Accuracy	Sensitivity	Misclassified (no.)	Misclassifcation Rate	Accuracy	Sensitivity	Misclassified (no.)	Misclassifcation Rate
Adaboost	81.27%	77.68%	1093	18.73%	80.71%	80.75%	1126	19.29%				
C5.0Tree	80.38%	78.03%	1145	19.62%	78.53%	79.58%	1253	21.47%				
RandomForest (ranger)	80.35%	78.87%	1147	19.65%	80.93%	79.54%	1113	19.07%				
Treebag	79.70%	76.00%	1185	20.31%	79.32%	80.71%	1207	20.68%				
MARS (earth)	79.22%	76.98%	1213	20.78%	79.11%	78.74%	1219	20.89%				
NeuralNetworks (nnet)	79.08%	77.68%	1221	20.92%	78.98%	80.00%	1227	21.02%				
GLM (glmnet)	77.91%	73.52%	1289	22.09%	77.33%	79.79%	1323	22.67%				
CART (rpart)	76.64%	70.46%	1363	23.36%	76.64%	73.79%	1363	23.36%				
avg (standalone) vs stack (actual)	**79.32%**	**76.15%**	**1207**	**20.68%**	**78.94%**	**79.11%**	**1229**	**21.06%**	**81.80%**	**78.95%**	**1062**	**18.20%**
Best Performing	*Adaboost*		1093	18.73%	Best Performing	*Adaboost*	1113	19.07%	Best Performing	*Stack of Models*	1062	18.20%

Figure 10: Model Testing Results

Application of *caretList* function (Deane-Mayer and Knowles; 2016) allowed to run all 8 models at once, and decreased performance of C5.0 Tree by 1.85 percent in accuracy, see table 10 The rest of the models were also affected either slightly negatively or positively in terms of accuracy, except CART. CART achieved accuracy of 76.64 percent as a standalone and as part of *caretList* (Deane-Mayer and Knowles; 2016), remaining the lowest performing model.

Developed model stack using *caretStack* functionality with logistic regression model method (GLM) (Deane-Mayer and Knowles; 2016), using all eight discussed models,

further improved the accuracy, achieving 81.80 percent. Developed meta-model from model stack achieved better results during testing phase than any of the standalone or *caretList*-trained models (Deane-Mayer and Knowles; 2016) . It misclassified least amount of observations - 1062 vs 1093 from Adaboost out of total of 5836, and achieved better sensitivity of 78.95 percent vs 77.87 percent for Random Forest, which is specifically prediction accuracy of wins as a positive class. When trained as part of the *caretList*, better sensitivity is apparent (see table 10). However, in terms of a number of actual true positives for "Won" class, model stack identified 1968 observations correctly versus 1926 by Adaboost within *caretList* (Deane-Mayer and Knowles; 2016). Overall, developed model stack performed better than other models for the task.

4.2 Model Deployment and Further Discussion

To avoid sampling bias during modelling process and optimise model performance for deployment, training based on balanced dataset is important. For this purpose k-folds with a cross-validation technique were employed (Kuhn; 2018; Prabhakaran; 2018) and the functionality of *createDataPartition* (Kuhn; 2018) ensured proportionate partitions of the data for testing and training, while k-folds ensured avoidance of one-off sampling bias and contributed to models' optimisation by creating multiple fold model runs. This allowed to produce optimised results with little extra computational effort and compensating for slightly unbalanced dependant variable "Won Lost". Since there is always going to be less wins than losses in the sales dataset, taking precautions to avoid sampling bias, and balanced split of the dataset according to the dependent variable can make a big difference to a model performance and prediction of wins.

Deployment is the final stage of CRISP-DM framework (Chapman et al.; 2000). As part of future works, developed model stack can be used in mainly two development scenarios: automating opportunity propensity score calculation in the CRM system outright, preventing salesperson from forecasting at opportunity-level, or semi-automating it.

4.2.1 Use Case 1: Automated Win-Propensity Forecast in CRM

Propensity output of the model stack should be placed into production to auto-populate sales stage within CRM system (or related sales propensity score fields) and therefore prevent salespersons from manually entering propensity score. Opportunities can be then prioritised by salespersons based on the value of the propensity score.

4.2.2 Use Case 2: Semi-Automated Win-Propensity in CRM

Propensity output of the model stack can autopopulate sales stage in the CRM system, whilst salespersons require to populate "Forecast Stage" field or equivalent, consisting of groups such as "Best Case" or "Commit" indicating commitment or no commitment to forecast. This will allow comparison of the model propensity output (likelihood of sale to be a win) with salespersons forecast judgement on the sale, thus providing an opportunity to review if sale is under/over-forecasted and introducing a business cross-validation on the model performance.

As per above use cases the opportunities then can be reviewed based on the propensity score as per figure 11 and prioritised by sales management and operationally accordingly.

For example, opportunity with `Deal_ID`: "609276" as per figure 11 has win-propensity of only 22 percent, of a quite large size for commercial business: 294,089 dollars. This

Deal_ID	Opportunity Source	Owner	Created by	Segment	Area	Fiscal Wk	Discount Approval	Industry	SE_TDS_Exists	Channel	Reseller_ Distributor_ Populated	Win-Propensity	Amount_USD
609276	Customer	Godfry Steade	Godfry Steade	1_ENT	13_ITALY	11	Approved	Data/Technology/Electronics	1	Direct	0	22%	$294,089
462312	Customer	Marvin Linden	Marvin Linden	3_COMMERCIAL	24_UKI	14	Expired	Energy/Fuels/Utilities	1	Indirect	1	94%	$235,468

Figure 11: Opportunity Win-Propensity Output Examples

opportunity can have a significant impact on the forecast, if forecasted and lost. Nevertheless opportunity has a small chance of win, discount has been approved and scheduled to close in week 11 of the quarter. Deal can be reviewed by sales in conjunction with Sales Engineers (SE) and other teams working on the sale to understand if the right solution was offered to customer, right discounting and licensing, what else needs to be done, or should it be de-prioritised and moved out of the quarter.

In contrast, opportunity with *Deal_ID*: "462312" has 94 percent win-propensity, quite large - 235,468 dollars, can also have a significant impact on the quarterly forecast. As per figure 11 it is scheduled to close in week 14 - last week of the quarter. Deals closing in week 14 always have high risk of slipping to future quarter. In this case, propensity score can allow for sales to review the deal and see if the deal can be closed earlier, for example by providing customer with a discount (as can be seen from figure 11 discount approval is "Expired") or by triggering reseller and /or distributor to get order processed earlier with the customer (as per figure 11 *Channel* is "Indirect" and *Reseller/Distributor Populated* is "1").

5 Conclusion and Future Work

This research project has achieved its main objective: development of a machine learning solution to automate B2B sales opportunity win-propensity score computation. Benefits of automation of sales propensity, as discussed, are extensive. It can achieve more accurate and objective opportunity assessment, forecasting and future planning, facilitate better resource allocation, increase operational efficiency and, consequently save time and cost to the business overall. In addition, the process of modelling itself is capable of providing some business insights to what drives the win-propensity of a sale. Though, in this case, applied solution is a black-box method, and therefore, insights on a sale that can be derived from the model itself are limited.

The amount of related works in use of machine learning for sales opportunity propensity is still surprisingly scarce, considering intensive development within machine learning field in the past few years. Clinical studies provide extensive research within the area of propensity, where the popularity of traditional logistic regression solution is diminishing at the expense of more accurate non-parametric, non-linear ensemble algorithms usually involving trees or boosting in its design (Wang et al.; 2018; Westreich, D. and Lessler, J. and Funk, M.J.; 2010; Zhao et al.; 2016). In the meantime, research within sales and marketing that uses propensity for some purpose, such as converting marketing leads to sales opportunities (Duncan and Elkan; 2015), developing list of sales prospects (D'Haen and Van Den Poel; 2013), development of all-round sales forecasting solution (Tang, L. and Xu, X. and Rangan, V.; 2017), and research specifically on sales propensity score (Yan et al.; 2015) - all use multi-model approach to solve for sales propensity, usually involving multiple non-parametric algorithms and/or complex modelling architecture. This is not surprising, due to poor data quality: corrupted labels, missing data, inconsistently populated fields, as well as high-dimensionality, and dominance of discrete and categorical

variables.

This research provided a simpler solution to sales propensity automation, not applied in any of the reviewed related works, leveraging the functionality of *caretEnsemble* package (Deane-Mayer and Knowles; 2016) - model stack. Use of opensource software: R and RStudio (Gentleman, R., and Ihaka, R.; 2018; Allaire; 2018), made this approach easily reproducible at a low cost. The meta-model developed from stack of eight models achieved better prediction accuracy than any of them standalone, inclusive of non-parametric models such as Random Forest and Neural Networks, and boosting ensemble algorithms such as Treebag and Adaboost.

Furthermore, developed model stack is relying on quite standard range of variables, which could be found in some form or shape in any CRM system, such as opportunity age, presence or absence of discount, presence or absence of pre-sales team involvement, products and geographical location. It does not rely on customer intelligence information, such as company or revenue size, market valet share, etc., which is often an expensive information to acquire from a third party source. However, such intelligence information has an ability to improve model stack performance and provide further insight to a successful sale.

As part of the future works, this model stack solution can be put into production and developed into an add-on compatible with an applicable CRM system to automate or semi-automate opportunity propensity score. For this purpose, further improvements to the model stack might be required - such as choice of boosting algorithms that are less computationally heavy than Adaboost; improvement to feature selection to lessen computational load, further optimisation of model choices for model stack blend, possible introduction of bayesian models. Application of machine learning to automate day-to-day sales operational business tasks and related model production methods is certainly an area requiring a lot of further research.

6 Appendix

Table 1: *B2B Sales Dataset* Variables Description

_	VariableName	UsedinModelling	VariableDescription
1	DealID		Unique sales opportunity ID
2	OpportunityOwnerAn		Salesperson name who owns the opportunity (anonymised)
3	OpportunityOwner.ManagerAn		Manager of the salesperson (Anonymised)
4	OpportunityOwnerAnCode		Numeric Code assigned to a sales person name
5	OpportunityOwner.ManagerAnCode		Numeric code assigned to a sales manager name
6	Country		Customer country
7	CountryCode	Yes	Customer country code
8	OpportunitySource		Source of the opportunity. Examples are channel-generated / marketing-generated
9	OpportunitySourceCode	Yes	Numeric code for opportunity source
10	OpportunitySourceGrouped		Numeric code and text value grouped into one field for opportunity source
11	OpportunityCurrency		Opportunity Currency
12	FiscalYear		Fiscal Year (not calendar). system-calculated based on close date
13	FiscalQuarter	Yes	Fiscal Quarter (not calendar). Between 1 and 4. System-calculated based on close date
14	FiscalWeek	Yes	Fiscal Week number of Quarter (not calendar). System-calculated based on close date
15	Channel		Type of channelling the sale. 7 possible values. Example: Direct and indirect (via reseller/distribution)
16	ChannelCode	Yes	Numeric code value of the Channel field
17	ChannelGrouped		Grouping of numeric code and its text value in one field for Channel
18	OpportunityId.18digit.		Another unique sales opportunity ID
19	SalesStage		Probability of sale between 0 (lost) and 100(Won). Assigned by salesperson
20	ReasonWon.Lost		If the opportunity is won or lost this field provides a dropdown of reasons
21	ReasonWon.LostCode		Numeric Code for Reason Won-Lost
22	ReasonWon.LostGrouped		Grouping of Text and its numeric code for ReasonWonLost
23	ApprovalStatus		Discount approval status. Contains 8 possible entries; example: Approved (discount approved); Expired (discount approval expired)
24	ApprovalStatusCode		Numeric code for each stage of the approval status
25	ApprovalStatusGrouped	Yes	Grouping of numeric code and its coressponding text value for Approval Status
26	AssociatedMktgActivity		Text field with reference to specific marketing campaigns opportunity was generated from
27	AssociatedMktgPopulated	Yes	Binary flag. 1 if AssociatedMktgActivty is populated; 0 - if it not
28	ResellerDistributorPopulated	Yes	Binary flag. 1 if ResellerAn and/or DistributorAn is populated; 0 if both are blank
29	CompetitorPopulated	Yes	Binary flag. 1 if PrimaryCompetitorAn is populated; 0
30	CreatedBYsameasOwner	Yes	Binary flag. 1 if CreatedbyAn is the same as OpportunityOwnerAn; otherwise 0
31	CreatedbyAn		Creator of an opportunity name
32	ResellerAn		Reseller Company Name
33	ResellerAnCode		Numeric code for reseller names in ResellerAn
34	DistributorAn		Numeric code for distributor names in DistributorAn
35	DistributorAnCode		Numeric code for distributor names in DistributorAn
36	PrimaryCompetitorAn		Text field; name of competitor for an opportunity
37	PrimaryCompetitorAnCode	Yes	Numeric code for PrimaryCompetitorAn
38	DiscountApprovalID		Unique ID for generated discount
39	DiscountApprovalIDFlag	Yes	Binary flag. 1 if DiscountApprovalID is populated; 0 if DiscountApprovalID is blank
40	SETDSExists	Yes	Binary flag. 1 if Sales Engineer is engaged into sale for opportunity; 0 if not
41	AccountID		Sales customer unique ID
42	AccountIDbyGroupedAccName		Sales customer ID based on de-duplication of customer names
43	AccountIDCode		Code assigned to sales customer unique ID
44	Band		Band based on AmountUSD; grouping of opportunity amount into categories
45	BandCode		Code assigned to band values for Band
46	NSTWPopulated	Yes	Binary flag. 1 if sales follow-up notes (Next Steps To Win) are populated; 0 if blank
47	ForecastStageOpenClosedLost		Opportunity Stage. Closed; Open; or Lost
48	AgeDays	Yes	Number of days between closed and created opportunity date
49	AgeBand		Banding of AgeDays
50	WonLost [ForecastStageBinary]	Yes	Binary flag based on ForecastStageOpenClosedLost field . 1 if "Closed"; 0 if "Lost" and "blank" if "Open"
51	AREA		Company-specific geographical roll-up of countries; for example Iberia is Spain and Portugal; Russia is all of Russia and CIS
52	AREACODE		Corresponding numeric code for AREA values
53	REGION		Company-specific geographical roll-up of AREA values
54	REGIONCODE	Yes	Corresponding numeric code for REGION values
55	SEGMENT		Company-specific sales segment; such as enterprise or commercial
56	SEGMENTCODE	Yes	Corresponding numeric code for SEGMENT values
57	FileNameSource		Source of corresponding system extracts; as collated into one
58	OpportunityCount		Row count in this case; as smallest granular level is a single opportunity
59	Quantitylicense	Yes	Software license quantity to be sold in the opportunity
60	AmountUSD [TotalPrice.converted.]	Yes	Total opportunity sales amount in dollars
61	ProductSuperFam1	Yes	Binary flag. 1 if this product superfamily is present in the opportunity; 0 if absent
62	ProductSuperFam2	Yes	Binary flag. 1 if this product superfamily is present in the opportunity; 0 if absent
63	ProductSuperFam3	Yes	Binary flag. 1 if this product superfamily is present in the opportunity; 0 if absent
64	ProductSuperFam4	Yes	Binary flag. 1 if this product superfamily is present in the opportunity; 0 if absent
65	ProductSuperFam5	Yes	Binary flag. 1 if this product superfamily is present in the opportunity; 0 if absent
66	ProductSuperFam6	Yes	Binary flag. 1 if this product superfamily is present in the opportunity; 0 if absent
67	TotalProduct	Yes	Total product count assigned to opportunity
68	Industry		Asigned industry segmentation based on labelling described in Section 3.2
69	Industrycode	Yes	Corresponding numeric code to the industry segments in Industry

Figure 12: Process Flow

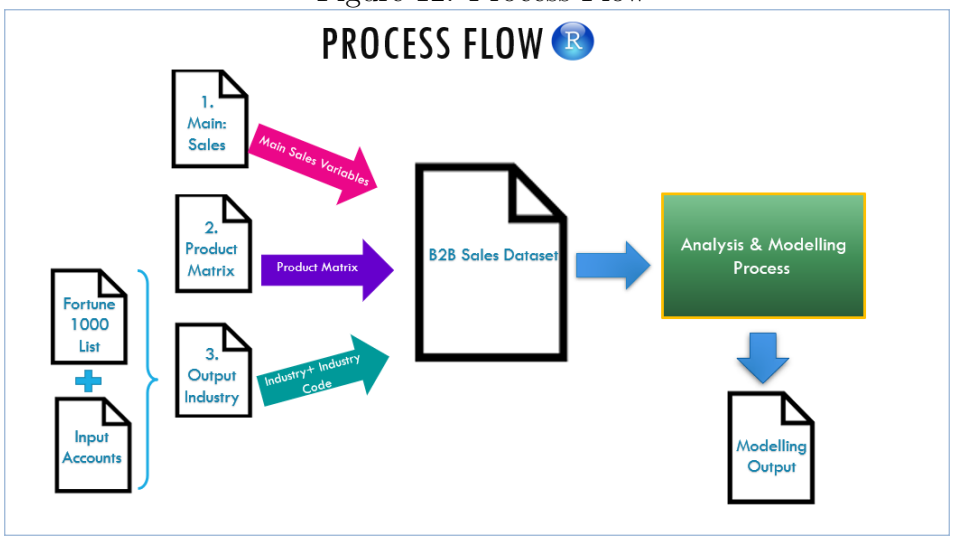


Figure 13: Confusion Matrices for Top Performing Models

random forest

Confusion Matrix and Statistics

	Reference	
Prediction	Lost	Won
Lost	2998	453
Won	694	1691

Accuracy	0.8035
95% CI	(0.793, 0.8136)
No Information Rate	0.6326
P-Value [Acc > NIR]	<2.2e-16
Kappa	0.5869
Mcnemar's Test P-Value	1.38E-12
Sensitivity	0.7887
Specificity	0.812
Pos Pred Value	0.709
Neg Pred Value	0.8687
Prevalence	0.3674
Detection Rate	0.2898
Detection Prevalence	0.4087
Balanced Accuracy	0.8004
'Positive' Class : Won	

C5.0

Confusion Matrix and Statistics

	Reference	
Prediction	Lost	Won
Lost	2965	486
Won	659	1726

Accuracy	0.8038
95% CI	(0.7934, 0.8139)
No Information Rate	0.621
P-Value [Acc > NIR]	<2.2e-16
Kappa	0.5895
Mcnemar's Test P-Value	3.71E-07
Sensitivity	0.7803
Specificity	0.8182
Pos Pred Value	0.7237
Neg Pred Value	0.8592
Prevalence	0.379
Detection Rate	0.2958
Detection Prevalence	0.4087
Balanced Accuracy	0.7992
'Positive' Class	Won

Adaboost

Confusion Matrix and Statistics

	Reference	
Prediction	Lost	Won
Lost	2930	521
Won	572	1813

Accuracy	0.8127
95% CI	(0.8025, 0.8227)
No Information Rate	0.6001
P-Value [Acc > NIR]	<2e-16
Kappa	0.6112
Mcnemar's Test P-Value	0.1304
Sensitivity	0.7768
Specificity	0.8367
Pos Pred Value	0.7602
Neg Pred Value	0.849
Prevalence	0.3999
Detection Rate	0.3107
Detection Prevalence	0.4087
Balanced Accuracy	0.8067
'Positive' Class	Won

caretStack

Confusion Matrix and Statistics

	Reference	
Prediction	Lost	Won
Lost	2970	481
Won	581	1804

Accuracy	0.818
95% CI	(0.8079, 0.8278)
No Information Rate	0.6085
P-Value [Acc > NIR]	<2.2e-16
Kappa	0.621
Mcnemar's Test P-Value	0.002382
Sensitivity	0.7895
Specificity	0.8364
Pos Pred Value	0.7564
Neg Pred Value	0.8606
Prevalence	0.3915
Detection Rate	0.3091
Detection Prevalence	0.4087
Balanced Accuracy	0.8129
'Positive' Class	Won

References

Allaire, J. (2018). Rstudio version 1.1.447 [computer software].
URL: *https://www.rstudio.com/*

Azevedo, A. I., Lourenço, R. and Santos, M. F. (2008). Kdd, semma and crisp-dm: a parallel overview, *IADIS European Conference Data Mining*, Amsterdam, Netherlands, pp. 182–185.
URL: *https://pdfs.semanticscholar.org/7dfe/3bc6035da527deaa72007a27cef94047a7f9.pdf*

Bailey, D. and Bailey., S. (2013). Online generate csv test data [computer programme].
URL: *http://www.convertcsv.com/generate-test-data.htm*

Bates, D., Chambers, J., Dalgaard, P., Gentleman, R., Hornik, K., Ihaka, R., Kalibera, T., Lawrence, M., Leisch, F., Ligges, U., Lumley, T., Maechler, M., Morgan, M., Murrell, P., Plummer, M., Ripley, B., Sarkar, D., Temple Lang, D., Tierney, L., Urbanek, S., Schwarte, H., Masarotto, G., Iacus, S., Murdoch, D. and Falcon, S. (2018). Documentation for package "base" version 3.6.0.
URL: *https://www.rdocumentation.org/packages/base/versions/3.5.1/topics/grep*

Bohanec, M., Robnik-Šikonja, M. and Kljajić Borštnar, M. (2017a). Decision-making framework with double-loop learning through interpretable black-box machine learning models, *Industrial Management & Data Systems* **117**(7): 1389–1406.
URL: *https://doi.org/10.1108/IMDS-09-2016-0409*

Bohanec, M., Robnik-Šikonja, M. and Kljajić Borštnar, M. (2017b). Explaining machine learning models in sales predictions, *Expert Systems with Applications* **71**(Supplement C): 416–428.
URL: *http://www.sciencedirect.com/science/article/pii/S0957417416306327*

Bohanec, M., Robnik-Šikonja, M. and Kljajić Borštnar, M. (2017c). Organizational Learning Supported by Machine Learning Models Coupled with General Explanation Methods: A Case of B2B Sales Forecasting, *Organizacija* **50**(3): 217–233.
URL: *http://organizacija.fov.uni-mb.si/index.php/organizacija/article/viewFile/780/1171*

Brocato, M. (2018). Mockaroo data generator [computer programme].
URL: *https://mockaroo.com/*

Caruana, R. and Niculescu-Mizil, A. (2006). An empirical comparison of supervised learning algorithms, *23d International Conference on Machine Learning*, Pittsburgh, USA, pp. 161–168.
URL: *https://www.cs.cornell.edu/ caruana/ctp/ct.papers/caruana.icml06.pdf*

Chapman, P., Clinton, J., R., K., Khabaza, T., Reinartz, T., Shearer, C. and Wirth, R. (2000). Crisp-dm 1.0: Step by step data-mining guide.
URL: *https://www.the-modeling-agency.com/crisp-dm.pdf*

Deane-Mayer, Z. A. and Knowles, J. E. (2016). Package "caretEnsemble".
URL: *https://cran.r-project.org/web/packages/caretEnsemble/caretEnsemble.pdf, https://github.com/zachmayer/caretEnsemble*

D'Haen, J. and Van Den Poel, D. (2013). Model-supported business-to-business prospect prediction based on an iterative customer acquisition framework, *Industrial Marketing Management* **42**(4): 544–551.
URL: *http://dx.doi.org/10.1016/j.indmarman.2013.03.006*

Duncan, B. A. and Elkan, C. P. (2015). Probabilistic Modeling of a Sales Funnel to Prioritize Leads, *21th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, KDD '15, ACM, New York, NY, USA, pp. 1751–1758.
URL: *http://doi.acm.org/10.1145/2783258.2788578*

Gentleman, R., and Ihaka, R. (2018). R Language 3.5.0.
URL: *https://www.r-project.org/, download R: https://ftp.heanet.ie/mirrors/cran.r-project.org/*

Gurnani, M., Korke, Y., Shah, P., Udmale, S., Sambhe, V. and Bhirud, S. (2017). Forecasting of sales by using fusion of machine learning techniques, *2017 International Conference on Data Management, Analytics and Innovation, ICDMAI 2017*, Pune, India, pp. 93–101.

Keen, B. (2005). Generatedata [computer programme].
URL: *https://www.generatedata.com/*

Kuhn, M. (2018). "Caret: Classification and Regression Training".
URL: *https://topepo.github.io/caret/,https://cran.r-project.org/web/packages/caret/caret.pdf*

Lawrence, R., Perlich, C., Rosset, S., Khabibrakhmanov, I., Mahatma, S., Weiss, S., Callahan, M., Collins, M., Ershov, A. and Kumar, S. (2010). Operations Research Improves Sales Force Productivity at IBM, *Interfaces* **40**(1): 33–46.
URL: *http://dx.doi.org/10.1287/inte.1090.0468*

Li, J., Cheng, K., Wang, S., Morstatter, F., Trevino, R. P., Tang, J. and Liu, H. (2017). Feature Selection: A Data Perspective, *ACM Computing Surveys* **50**(6): 94:1–94:45.
URL: *http://doi.acm.org/10.1145/3136625*

Linden, A. and Yarnold, P. R. (2016). Combining machine learning and propensity score weighting to estimate causal effects in multivalued treatments, *Journal of Evaluation in Clinical Practice* **22**(6): 871–881.
URL: *https://onlinelibrary.wiley.com/doi/abs/10.1111/jep.12610*

Lu, C.-J. (2014). Sales forecasting of computer products based on variable selection scheme and support vector regression, *Neurocomputing* **128**: 491 – 499.
URL: *http://www.sciencedirect.com/science/article/pii/S0925231213008850*

Lu, C.-J. and Kao, L.-J. (2016). A clustering-based sales forecasting scheme by using extreme learning machine and ensembling linkage methods with applications to computer server, *Engineering Applications of Artificial Intelligence* **55**: 231 – 238.
URL: *http://www.sciencedirect.com/science/article/pii/S0952197616301257*

Lu, C. J., Lee, T. S. and Lian, C. M. (2012). Sales forecasting for computer wholesalers: A comparison of multivariate adaptive regression splines and artificial neural networks, *Decision Support Systems* **54**(1): 584–596.
URL: *http://dx.doi.org/10.1016/j.dss.2012.08.006*

Monat, J. (2011). Industrial sales lead conversion modeling, *Marketing Intelligence Planning* **29(2)**: 178–194.
URL: *https://www.emeraldinsight.com/doi/abs/10.1108/02634501111117610*

Prabhakaran, S. (2018). Caret package a practical guide to machine learning in r.
URL: *https://www.machinelearningplus.com/machine-learning/caret-package/*

Robinson, D. and Elias, J. (2018). Package "Fuzzyjoin".
URL: *https://cran.r-project.org/web/packages/fuzzyjoin/fuzzyjoin.pdf*

Rudis, R. (2016). 2015 fortune 1000 list with industry and website.
URL: *https://gist.github.com/hrbrmstr/ae574201af3de035c684*

Tang, L. and Xu, X. and Rangan, V. (2017). Hitting your number or not? A robust intelligent sales forecast system, *2017 IEEE International Conference on Big Data*, Boston, MA, USA, pp. 3613–3622.
URL: *https://ieeexplore.ieee.org/document/8258355/*

Tkac, M. and Verner, R. (2016). Artificial neural networks in business: Two decades of research, *Applied Soft Computing* **38**: 788 – 804.
URL: *http://www.sciencedirect.com/science/article/pii/S1568494615006122*

Wang, C., Wang, S., Shi, F. and Wang, Z. (2018). Robust Propensity Score Computation Method based on Machine Learning with Label-corrupted Data, *arXiv* pp. 1–26.
URL: *http://arxiv.org/abs/1801.03132*

Westreich, D. and Lessler, J. and Funk, M.J. (2010). Propensity score estimation: neural networks, support vector machines, decision trees (CART), and meta-classifiers as alternatives to logistic regression, *Journal of Clinical Epidemiology* **63**(8): 826–833.
URL: *https://www.sciencedirect.com/science/article/pii/S0895435610001022*

Xia, M., Zhang, Y., Weng, L. and Ye, X. (2012). Fashion retailing forecasting based on extreme learning machine with adaptive metrics of inputs, *Knowledge-Based Systems* **36**: 253–259.
URL: *http://dx.doi.org/10.1016/j.knosys.2012.07.002*

Yan, J., Zhang, C., Zha, H., Gong, M., Sun, C., Huang, J., Chu, S. and Yang, X. (2015). On machine learning towards predictive sales pipeline analytics, *Twenty-ninth AAAI conference on artificial intelligence*.

Yu, Y., Choi, T.-M. and Hui, C.-L. (2011). An intelligent fast sales forecasting model for fashion products, *Expert Systems with Applications* **38**(6): 7373 – 7379.
URL: *http://www.sciencedirect.com/science/article/pii/S0957417410014521*

Zhang, C., Li, X., Yan, J., Qui, S., Wang, Y., Tian, C. and Zhao, Y. (2014). Sufficient statistics feature mapping over deep boltzmann machine for detection, *22nd International Conference on Pattern Recognition*, Stockholm, Sweden, pp. 827–832.
URL: *https://ieeexplore.ieee.org/document/6976862/*

Zhao, P., Su, X., Ge, T. and Fan, J. (2016). Propensity score and proximity matching using random forest, *Contemporary clinical trials* **47**: 85–92.
URL: *https://www.contemporaryclinicaltrials.com/article/S1551-7144(15)30143-9/fulltext*