Information Management Master

Predictive Methods of Data Mining

LaGoste Company

Report Delivery

June, 2019

Group 12

Carlos Lopes (M20170532)

Inês Marques (M20170211)

Ricardo Gonçalves (M20170300)

Sofia Gouveia (M20180655)

Index

Figure Index

Table Index

1. Introduction

The aim of this project is to deliver a robust predictive model that can anticipate the outcome of a marketing campaign.
The data available come from a previous pilot campaign conducted to 3.000 customers and the model will be built in order to predict the maximum expected profit from the same campaign directed to 5.000 customers.
This model will allow us to know which customers are more likely to accept the new campaign, avoiding to contact customers that are less likely to accept it, therefore optimizing the budget available for these initiatives.
To achieve the purpose of this project the SAS Enterprise Miner Workstation will be used and the SEMMA approach will be followed.

2. LaGoste Company

LaGoste store is a well-established organization that operates in the fashion, sports and luxury sector with around 300.000 registered customers and serves more than 1.000.000 consumers per year.
The products available can be included in 5 major categories, namely: Sneakers, Rackets, T-shirts, Watches and Hats. Which are also classified as Premium Brand material and more mainstream articles.
Three different channel groups are available to order and acquire these products: physical stores, quarterly catalogs and the companies' website.
Although in the last 5 years the company had solid revenues, the profit growth perspectives for the next two years are unstable. To change this situation, strategic initiatives are being held, like a marketing efficiency program.
While the marketing department is under pressure to spend more wisely the annual budget our team of data scientist was established in order to build a predictive model to support direct marketing initiatives.
The total cost of the sample campaign was 9.000€ resulting from 3.000 contacts * 3€. 375 customers out of 3.000 accepted the offer (12.5% of the contacts), each contributing with 13€ of revenue. However, the campaign profit was -4.125€.
The model should recognize the customers that are most likely to purchase the offer and leave out the non-purchasing customers, increasing the profitability of the campaign.

3. Descriptive Data Analysis

To build the predictive model we will follow the SEMMA approach. Before building the model is necessary to understand and explore the data available in order to know the principle variables' features and the result of the pilot campaign.

3.1. File import

This node is used to import the data from an excel file and to define the variables in terms of role (Input, Target, Reject and ID) and level (Binary, Nominal and Interval).
The classification of the variables depends on the way its information will be used by the model. Regarding the level of the variables, the Binaries only have two results: 0 if the customer declines or 1 if accepts. The Nominal variables are classificatory and the Interval ones represent quantitative values.
Concerning the role, the variable Custid was defined as ID because it represents each customer's ID which means it needs to have a unique value for each one of them. The dependent variable of the model is the DepVar which means it must be defined as Target.
The rejected variables were Z_CostContact and Z_Revenue because they are constant values and Group, Element1, Element2, Element3, Element4, and Element5 because they are related to the members and number of the group, not having meaning to the model.

Name	Role /
Custid	**ID**
MntRackets	**Input**
MntSneakers	**Input**
MntPremium_Br	**Input**
MntTShirts	**Input**
MntWatches	**Input**
Kidhome	**Input**
AcceptedCmp4	**Input**
MntHats	**Input**
Marital_Status	**Input**
Recency	**Input**
NumWebVisitsM	**Input**
Year_Birth	**Input**
Teenhome	**Input**
NumDealsPurcha	**Input**
NumCatalogPurc	**Input**
NumWebPurchas	**Input**
NumStorePurcha	**Input**
AcceptedCmp2	**Input**
Complain	**Input**
Dt_Customer	**Input**
Education	**Input**
Income	**Input**
AcceptedCmp1	**Input**
AcceptedCmp5	**Input**
AcceptedCmp3	**Input**
Group	**Rejected**
Element5	**Rejected**
Z_CostContact	**Rejected**
Z_Revenue	**Rejected**
Element2	**Rejected**
Element1	**Rejected**
Element4	**Rejected**
Element3	**Rejected**
DepVar	**Target**

Figure 1: Variables' roles and levels from FileImport node.

3.2. Multiplot

The multiplot node enables the visualization of the graphics created with the variables of the model, which makes possible to understand the trends and variables' behavior.
Figure 2 presents the graphic comparing the variable Recency and the Dependent Variable. There we can observe that the customers which accepts more campaigns are the ones who have a recency between 2.5 and 22.5. Within these customers the ones with higher rate of acceptance are the 2.5 and 7.5.

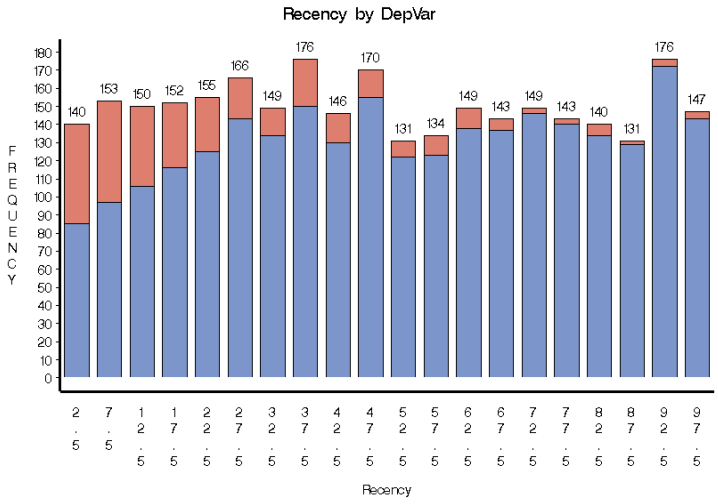

Figure 2: Graph of Recency by DepVar

When analyzing Figure 3 we can conclude that most customers made between 3 to 5 catalog purchases and the ones that accepted more the DepVar were the ones that made 4 catalog purchases.

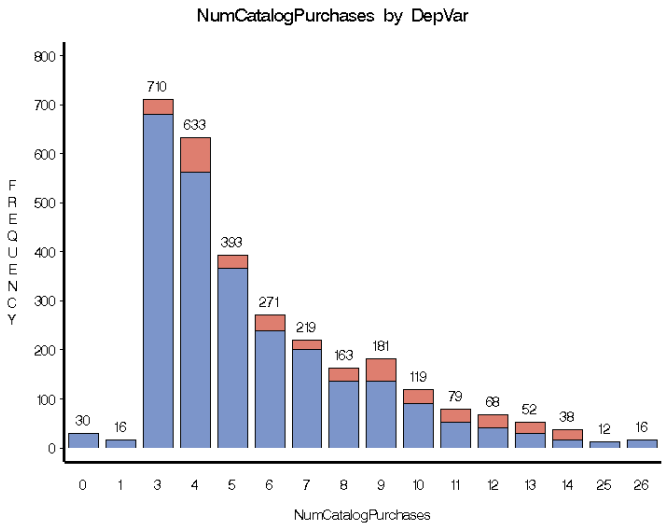

Figure 3: Graph of NumCatalogPurchases by DepVar

The percentage of customers' acceptance of campaign 5 and their acceptance of the current campaign (DepVar) is represented in Figure 4, which showed us that around 7,13% of customers that reject campaign 5 accepted the current campaign.

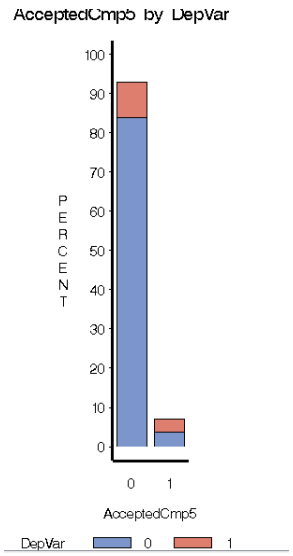

Figure 4: Customers who accepted campaign 5

Figure 5 shows around 70% of customers spent less than 160 monetary units during the last 18 months.

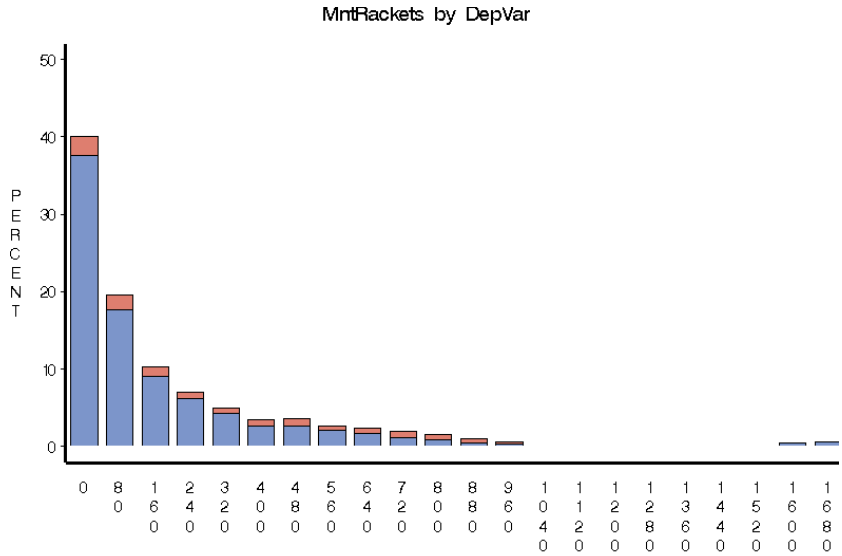

Figure 5: Graph of the monetary spent on Rackets by DepVar

By analyzing Figure 6, we can understand that most customers made between 4 to 9 web purchases in the last 18 months.

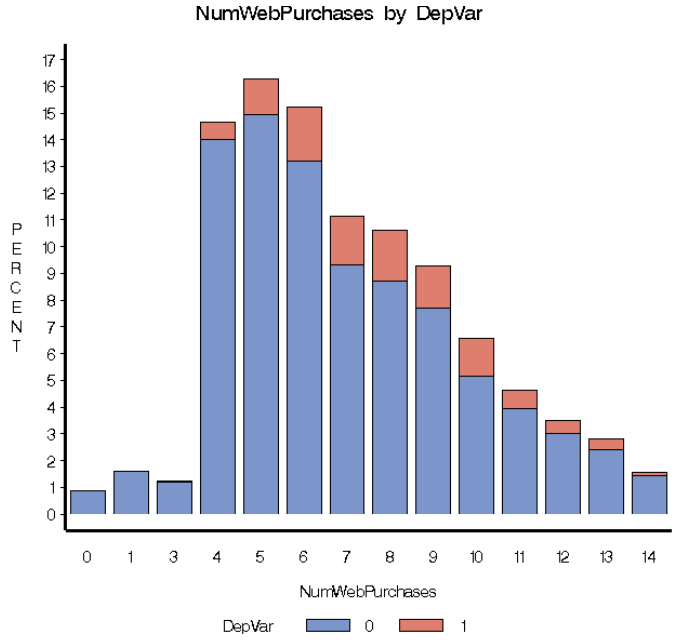

Figure 6: Graph of NumWebPurchases by DepVar

Figure 7 show us the number of kids in household. With the analysis of this graphic, we can understand that most are without kids (58%) or just one kid (39%).

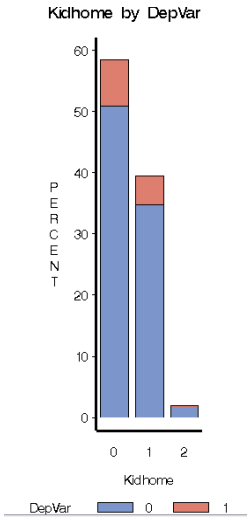

Figure 7: Graph of Kidhome by DepVar

Figure 8 show us the marital status of the customers. With the analysis of this graphic, we can understand that most are married (41,80%) or together (24,73%). From those customers, the ones with a higher percentage of acceptance of the new campaign are the ones who are married.

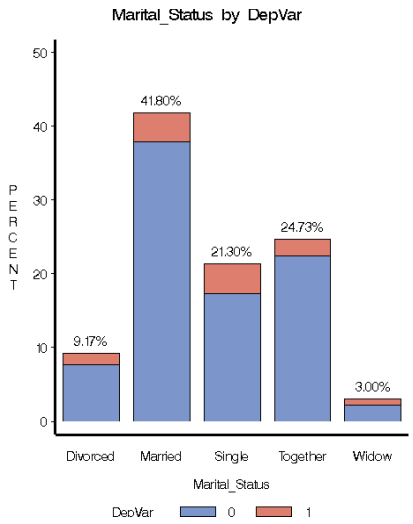

Figure 8: Graph of Marital Status by DepVar

Figure 9 shows that there is no particular distribution of customers who have accepted the current campaign by Income. While analyzing this graphic we could consider as outliers the values above 192000. The income Average is 91.995,63 u.m.

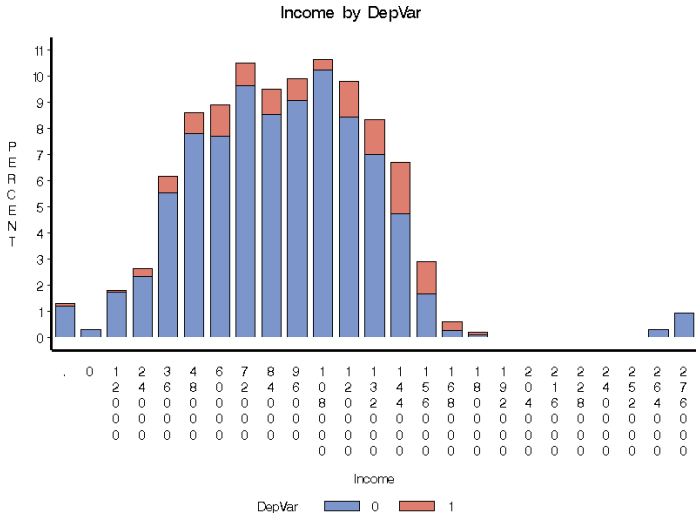

Figure 9: Income by DepVar

From the information gathered and analyzed above, we verify that typically the costumers have a high education, are mainly married and together, with a good income level, maximum 2 kids and the average 0,44 kids in home.

3.3. Stat Explore

The Stat Explore node presents a graph that relates the variables with the dependent variable, showing which ones are the most relevant variables.
As we can see in Figure 10 the variables that best explain our DepVar are Recency, NumCatalogPurchases, MntRackets and AcceptedCmp5 whereas the variables that are less relevant are Education, Kidhome and Complain.

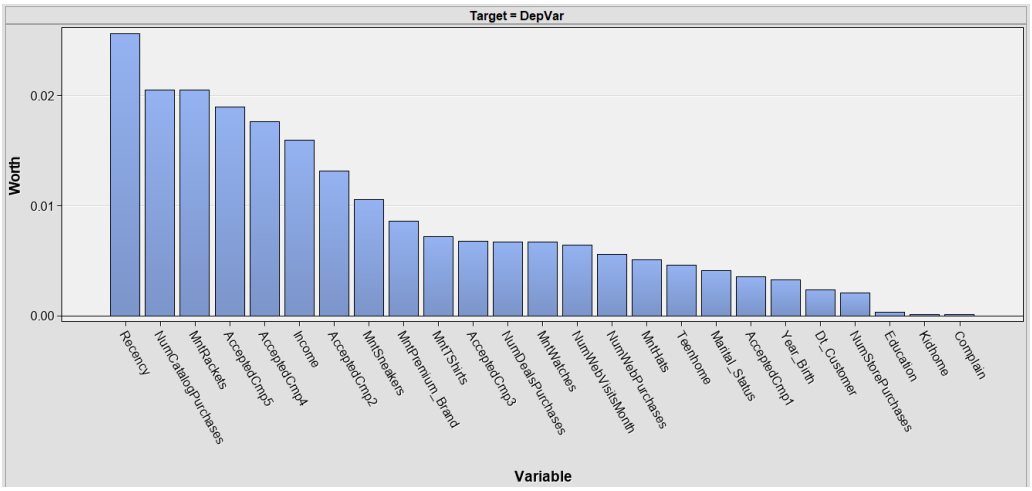

Figure 10: Variable worth Graph.

This node also gives descriptive information concerning data set variables like role, number of levels, missing values and variables' statistical data while dividing the variables into class and interval variables.
In the results for the class variables (Figure 11) we can observe that there are no missing values, that the percentage of graduated customers is 51.43% whereas 21.27% has PhD and that 41.80% of the customers are married and 24,73% are together.

Data Role	Variable Name	Role	Number of Levels	Missing	Mode	Mode Percentage	Mode2	Mode2 Percentage
TRAIN	AcceptedCmp1	INPUT	2	0	0	98.93	1	1.07
TRAIN	AcceptedCmp2	INPUT	2	0	0	91.60	1	8.40
TRAIN	AcceptedCmp3	INPUT	2	0	0	92.67	1	7.33
TRAIN	AcceptedCmp4	INPUT	2	0	0	92.30	1	7.70
TRAIN	AcceptedCmp5	INPUT	2	0	0	92.87	1	7.13
TRAIN	Complain	INPUT	2	0	0	98.93	1	1.07
TRAIN	Education	INPUT	5	0	Graduation	51.43	PhD	21.27
TRAIN	Marital_Status	INPUT	5	0	Married	41.80	Together	24.73
TRAIN	DepVar	TARGET	2	0	0	87.50	1	12.50

Figure 11: Output of StatExplore for class variables

Concerning the interval variables, Figure 12 shows us missing values for Income, MntHats and MntPremium_Brand (39, 60 and 49 missing values, respectively). In average, the customers have 46 years old (2019 – 1973) and an income of around 91995 monetary units (m.u.) per customer. The customers preferred the product sneakers (MntSneakers) which is the product with higher monetary units spent (305 m.u.) and the web has around 5 visits per month (NumWebVisitsMonth).

Variable	Role	Mean	Standard Deviation	Non Missing	Missing	Minimum	Median	Maximum	Skewness	Kurtosis
Dt_Customer	INPUT	20652.47	202.8481	3000	0	20299	20658	20999	-0.02613	-1.20731
Income	INPUT	91995.63	41992.34	2961	39	2379	91114	278861	0.827861	2.597744
Kidhome	INPUT	0.435	0.534668	3000	0	0	0	2	0.653277	-0.78088
MntHats	INPUT	54.59864	80.77088	2940	60	0	18	394	2.026623	3.637086
MntPremium_Brand	INPUT	58.14605	66.83243	2951	49	0	33	323	1.770147	2.732002
MntRackets	INPUT	187.2303	262.8342	3000	0	0	76	1699	2.493917	8.449003
MntSneakers	INPUT	305.7383	329.7255	3000	0	0	186	1494	1.144153	0.606732
MntTShirts	INPUT	27.54067	40.18685	3000	0	0	9	199	2.042068	3.839913
MntWatches	INPUT	40.67067	59.31153	3000	0	0	15	299	2.078337	4.009474
NumCatalogPurchases	INPUT	5.844667	3.428964	3000	0	0	5	26	2.230611	8.764587
NumDealsPurchases	INPUT	2.436	2.326451	3000	0	0	2	16	2.873995	10.88049
NumStorePurchases	INPUT	7.722667	3.359591	3000	0	0	7	15	0.493529	-0.51828
NumWebPurchases	INPUT	7.005	2.792055	3000	0	0	7	14	0.400393	-0.13077
NumWebVisitsMonth	INPUT	5.257333	2.760251	3000	0	0	5	20	0.962736	4.801524
Recency	INPUT	49.15	28.81225	3000	0	0	48	99	0.048417	-1.19752
Teenhome	INPUT	0.484	0.536507	3000	0	0	0	2	0.433099	-1.05119
Year_Birth	INPUT	1971.601	11.8526	3000	0	1944	1973	1999	-0.07462	-0.83665

Figure 12: Output of StatExplore for Interval variables

3.4. Variable Clustering

The Variable Clustering node improves the selection of clusters and variables to analyze since it makes easier the understanding and determination of the relationships that might exist. It also provides the access to the correlation matrix as shown in Figure 13.

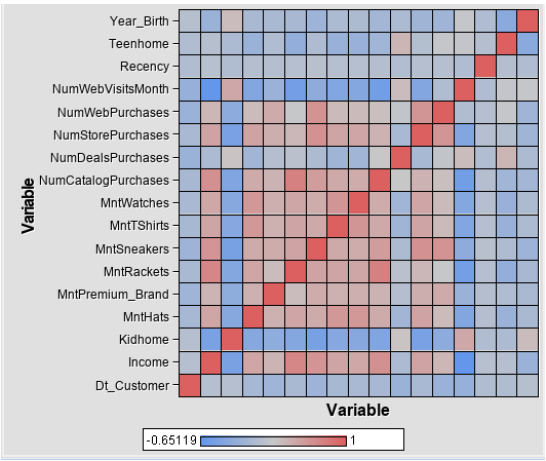

Figure 13: Correlation Matrix

3.5.Graph Explore

The Graph Explore node enables graphically the exploration and understanding of trends and patterns of large data volumes.
Figure 14 shows that 12,6% of costumers accepted the offer

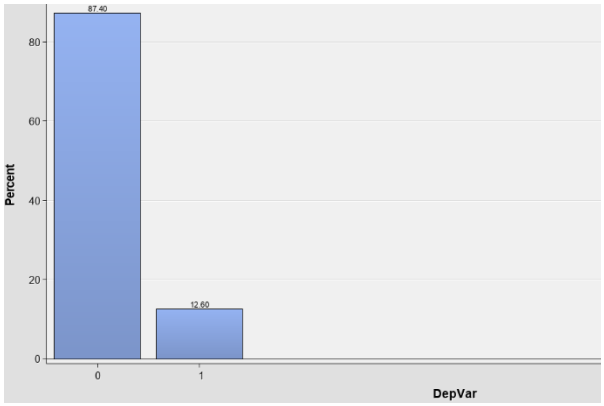

Figure 14: DepVar Graph

4. Data Pre-Processing

4.1.Outliers

Observations that broadly differ from the normal range are called outliers. When working with data mining it's important to understand the impact of this kind of data in the analysis, since they can represent errors, fraud or very profitable niche markets. However, outliers can cause bias and compromise the model.
In this way, outliers should be excluded but keeping in mind that the maximum value established to be acceptable for excluding outliers is 3%.
In the LaGoste project a visual manual method of extreme value cut-off was used to set the upper and lower limits, being 72 observations excluded, which represents 2,4% of the total data, lower than the maximum accepted (Figure 15).

Number Of Observations

Data Role	Filtered	Excluded	DATA
TRAIN	2928	72	3000

Figure 15: Excluded outliers (Filter node)

The variables that had outliers were only interval variables, namely: Income; MntRackets; NumCatalogPurchases and NumWebVisitsMonth. The maximum and minimum values for each of these variables are presented in Figure 16.

Filter Limits for Interval Variables
(maximum 500 observations printed)

Variable	Role	Minimum	Maximum	Filter Method	Keep Missing Values	Label
Income	INPUT	0	203927.75	MANUAL	Y	Income
MntRackets	INPUT	0	1133.55	MANUAL	Y	MntRackets
NumCatalogPurchases	INPUT	0	14.88	MANUAL	Y	NumCatalogPurchases
NumWebVisitsMonth	INPUT	0	10.67	MANUAL	Y	NumWebVisitsMonth

Figure 16: Filter limits for interval variables with outliers

4.2. Missing Values

As mentioned in chapter 3.3, missing values were found in three interval variables (Income, MntHats and MntPremium_Brand). The reason for the occurrence of these blank observations can be, for example, not wanting to reveal their income or not remembering about previous purchases.
In order to obtain more robustness of the data these missing values were filled by using the impute node, described in the Appendix 2.

4.3. Variable Transformation

After data treatment is important to work with the existing variables, creating new ones, to obtain relevant information, not available from the beginning, but also to improve variables' quality. To create these new variables a SAS Code node and a variable transform node were used.

Table 1: Variable transformation by SAS Code node.

Variable	Formula
MntProd	MntHats + MntRackets + MntSneakers + MntWatches + MntTShirts
Frq	(NumWebPurchases + NumStorePurchases + NumCatalogPurchases)/3
HigherEducation	1*(upcase(Education) in ("GRADUATION","MASTER","PHD"))

AcceptedCmp	AcceptedCmp=0;
	if (AcceptedCmp1=1 or AcceptedCmp2=1 or AcceptedCmp3=1 or AcceptedCmp4=1 or AcceptedCmp5=1) then do;
	AcceptedCmp=1;
	end;

Table 2: Variable transformation by Transform Variable node.

Variable	Formula
Age	year(today())-Year_Birth
RMntFrq	MntProd/Frq

The variable MntProd gives the total amount spent on LaGoste products; the variable Frq represents the frequency of purchases and the RMntFrq describes the average amount spent per purchase. These three variables are related to previous purchases record which can be relevant for the predictive model.
While the HigherEducation and Age are important variables for customers' social characterization.
The AcceptedCmp was created to know if the customer accepted any of the previous campaigns or not.

4.4. Coherence Checking

In the coherence checking node a SAS code was introduced to define some rules to flag when incoherent values result from interactions between input variables for some intervals.

4.5.Data Partition

After completing the data preparation, the dataset must be partitioned between training, validation and test sets.
The training set is used for preliminary model fitting, while the validation set is used to assess the adequacy of the model when performing the model comparison step and to prevent overfitting models to the training data. The final unbiased estimate of the model quality is performed with the test data set.
Given that partitioning provides mutually exclusive datasets, it is paramount to keep enough observations on each set to ensure the fit of the model and its ability to generalize.
For smaller datasets, it is not uncommon to omit the test set. As such, following the recommendations given in the practical classes, we decided to allocate 70% of the data to the training set and 30% to the validation set.
As for the partitioning method, the Data Partition node supports three types of partitioning: Simple Random, Stratified and Cluster.
In the present case, since the target variable is a class variable, is it recommended to use a stratified partitioning method, in order to keep the same proportions of the target variable (stratification variable) within each subgroup.
Figure 17 presents the results for the Data Partition step.

Summary Statistics for Class Targets

Data=DATA

Variable	Numeric Value	Formatted Value	Frequency Count	Percent	Label
DepVar	0	0	2553	87.1926	DepVar
DepVar	1	1	375	12.8074	DepVar

Data=TRAIN

Variable	Numeric Value	Formatted Value	Frequency Count	Percent	Label
DepVar	0	0	1786	87.2070	DepVar
DepVar	1	1	262	12.7930	DepVar

Data=VALIDATE

Variable	Numeric Value	Formatted Value	Frequency Count	Percent	Label
DepVar	0	0	767	87.1591	DepVar
DepVar	1	1	113	12.8409	DepVar

Figure 17: Data Partition Results

4.6.Variable Selection

The main purpose of variable selection is to determine which are the most relevant to help predict the behavior of the target variable while also avoiding redundancy of information between them.

Unnecessary predictors will not only add noise to the estimation of other quantities that we are interested in but can also cause collinearity. Furthermore, if the ultimate goal is to use the selected model for prediction, we can save time and/or money by not measuring redundant predictors.

There are several techniques to achieve dimensionality reduction. In this project, we applied two distinct methods:

- Stepwise Regression;
- Manual method.

4.6.1.Stepwise Regression

Regression models can be used to describe the relationship between a set of independent variables (predictors / regressors) and the dependent variable (target), as well as to assess in what extent a predictor variable is relevant to the target behavior.

Given that the target variable is binary, a logistic regression model should be preferred (over linear regression).

The Regression node supports forward, backward and stepwise selection methods.

The forward method begins with no candidate effects in the model and adds effects until the entry significance level or the stop criterion is met.
On the contrary, the backward method begins with all candidate effects and removes effects until the stay significance level or the stop criterion is met.
The stepwise method begins with no candidate effects, as in the forward model, and then systematically adds effects that are significantly associated with the target. However, after an effect is added to the model, stepwise may remove any effect already in the model that is not significantly associated with the target. This process continues until the stay significance level or the max steps criterion is met (the default value is set to the number of effects in the model). Additionally, the process is also be terminated if an effect added in one step is the only effect deleted in the next step.

Figure 18 presents the last iteration of the Stepwise Regression algorithm performed and the selected variables.

Type 3 Analysis of Effects

Effect	DF	Wald Chi-Square	Pr > ChiSq
AcceptedCmp	1	4.3798	0.0364
AcceptedCmp2	1	45.7733	<.0001
AcceptedCmp3	1	47.8371	<.0001
AcceptedCmp4	1	43.8369	<.0001
AcceptedCmp5	1	47.1396	<.0001
Education	4	11.1045	0.0254
Frq	1	74.9060	<.0001
IMP_Income	1	12.5017	0.0004
Marital_Status	4	64.7487	<.0001
MntProd	1	42.7953	<.0001
MntRackets	1	91.6045	<.0001
NumCatalogPurchases	1	114.7155	<.0001
NumDealsPurchases	1	83.4391	<.0001
NumWebVisitsMonth	1	90.6065	<.0001
Recency	1	144.0508	<.0001
Teenhome	1	27.7757	<.0001

Figure 18: Variables selected by the Stepwise Regression method

4.6.2.Manual Method

As recommended during the practical classes, this assessment was performed by combining the “worth” of the different variables (found in the Stat Explore (2) node results) with the Spearman correlation between variables (presented in the Correlation Matrix node).
The variable worth plot (Figure 19) ranks input variables according to their calculated worth in predicting the target variable.

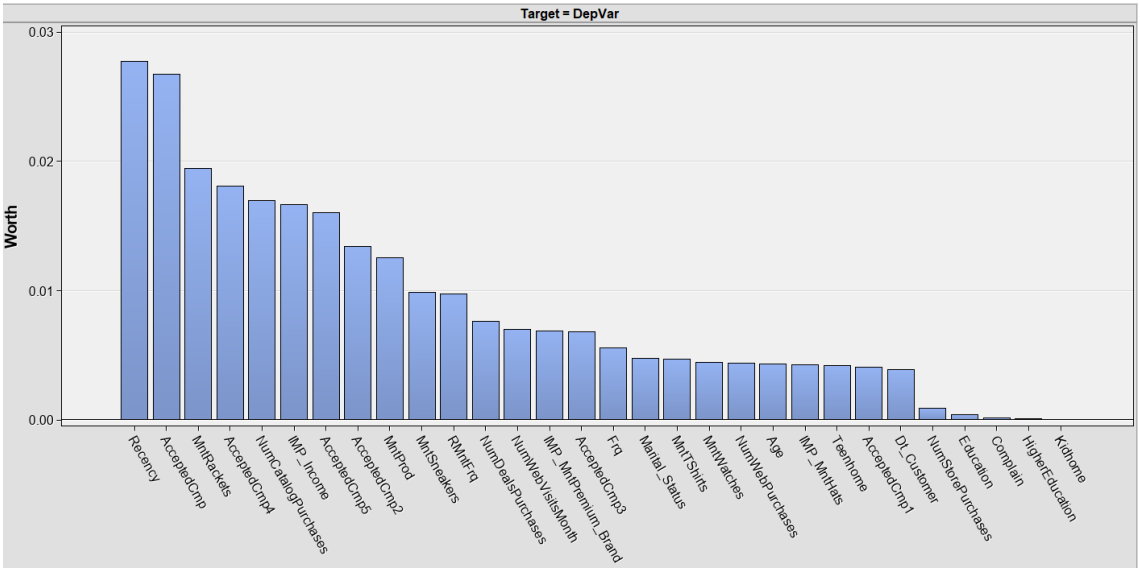

Figure 19: Variable worth plot (Stat Explore (2) node)

Complementing this observation, the correlation matrix node allows us to analyze the Spearman rank correlation in a cross-tabular format (see Appendix 3).

The Spearman correlation is a measure that quantifies the extent of statistical dependence between pairs of observations. It resembles the Pearson correlation, but has the advantage that it is a nonparametric measure, appropriate for evaluating both linear and non-linear relations.

Although traditionally we consider that two variables are highly correlated if the Spearman correlation value is above is 0.7, for the current project a cut-off value of 0.8 was advised.

As the use of highly correlated variables in the predictive model would add redundancy, this allowed us to select some variables that could be excluded. The decision on which variable to select was based on the variable worth.

Table 3 depicts the steps and decisions taken regarding variable selection based on the Spearman Correlation combined with the variable's worth.

Table 3: Manual Method using Variable Worth value and Spearman correlation

Variable A	Worth	Variable B	Worth	Spearman correlation	Action
MntRackets	0.0195	MntProd RMntFreq	0.0125 0.0098	0.88 0.862	Keep MntRackets
NumCatalog Purchases	0.0170	Frq	0.0056	0.809	Keep NumCatalog Purchases
Imp_Income	0.0166	MntProd RMntFreq	0.0125 0.0098	0.828 0.81	Keep Imp_Income
MntProd	0.0125	MntRackets Imp_Income MntSneakers RMntFrq Frq	0.0195 0.0166 0.0099 0.0098 0.0056	0.88 0.828 0.895 0.972 0.805	Keep MntProd

MntSneakers	0.0099	MntProd RMntFrq	0.0125 0.0098	0.895 0.863	Exclude MntSneakers
RMntFrq	0.0098	MntRackets Imp_Income MntProd MntSneakers	0.0195 0.0166 0.0125 0.0099	0.862 0.81 0.972 0.863	Exclude RMntFrq
Frq	0.056	NumCatalog Purchases MntProd NumStore Purchases	0.0170 0.0125 0.0009	0.809 0.805 0.864	Exclude Frq

Given that class variables do not present measurable correlations, this particular step could only be applied to interval variables.

As such, the selection of class variables was made through the observation of their relative worth and position in the overall variable worth ranking, as shown in Table 4.

Additionally, we took into account that the variables AcceptedCmp1-5, which were used to create the new variable AcceptedCmp, would certainly be correlated to the latter. Also, the new combined variable is highly ranked in terms of worth and above the original variables. For these reasons, we decided to exclude all five variables (AcceptedCmp1-5) from the predictive model.

Table 4: Variable Worth Ranking (Top 20)

Ranking	Variable	Worth	Action
1	Recency	0.0277	Keep
2	AcceptedCmp	0.0267	Keep
3	MntRackets	0.0195	Keep
4	AcceptedCmp4	0.0181	Exclude
5	NumCatalogPurchases	0.0170	Keep
6	IMP_Income	0.0166	Keep
7	AcceptedCmp5	0.0160	Exclude
8	AcceptedCmp2	0.0134	Exclude
9	MntProd	0.0125	Keep
10	MntSneakers	0.0099	Exclude
11	RMntFrq	0.0098	Exclude
12	NumDealsPurchases	0.0076	Keep
13	NumWebVisitsMonth	0.0070	Keep
14	IMP_MntPremium_Brand	0.0069	Keep
15	AcceptedCmp3	0.0068	Exclude
16	Frq	0.0056	Exclude
17	Marital_Status	0.0048	Keep
18	MntTShirts	0.0047	Exclude
19	MntWatches	0.0044	Exclude
20	NumWebPurchases	0.0044	Exclude

4.6.3.Method comparison and final selection

After obtaining the results from the different variable selection methods applied, we decided to test each pack of variables selected through all the models that will be explained on the next chapter (Artificial Neural Networks, Logistic Regression and Decision Trees).
Considering that the purpose of this project is to select the most profitable model, the final decision on which set of variables to select was based on the results from the Estimated Profit Curve node.
Also, since there would be a 50€ penalization for each variable selected above the maximum limit of 10, we applied a trial-and-error method, to find if there were any variables that could be dropped without affecting the final result, particularly in the case of the stepwise regression method, which returned a total of 12 variables.
Several combinations were tested, but none of them surpassed the maximum estimated profit of 4.913.
The final decision was based on a combination of the results provided by both the selection models applied, as shown in Table 5.

Table 5: Test results from variable selection methods in terms of highest profit

Variable Selection Method	Variables	Highest Profit
Stepwise Regression	1. AcceptedCmp 2. Education 3. Frq 4. IMP_Income 5. Marital_Status 6. MntRackets 7. MntProd 8. NumCatalogPurchases 9. NumDealsPurchases 10. NumWebVisitsMonth 11. Recency 12. Teenhome	(4.913€ - 100€) = 4.813€
Manual Method	1. Recency 2. AcceptedCmp 3. MntRackets 4. NumCatalogPurchases 5. IMP_Income 6. MntProd 7. NumDealsPurchases 8. NumWebVisitsMonth 9. IMP_MntPremium_Brand 10. Marital_Status	4.770€
Combination of both methods	1. Recency 2. AcceptedCmp 3. MntRackets 4. NumCatalogPurchases 5. IMP_Income 6. MntProd	4.913€

	7. NumDealsPurchases 8. NumWebVisitsMonth 9. Marital_Status 10. Teenhome	

Thus, the variables that will be used to develop the predictive models and selected in the subsequent manual Metadata node as input variables are the following:

1. Recency
2. AcceptedCmp
3. MntRackets
4. NumCatalogPurchases
5. IMP_Income
6. MntProd
7. NumDealsPurchases
8. NumVisitsWebMonth
9. Marital_Status
10. Teenhome

5. Predictive Model

In this chapter, we will be developing the predictive model with training and validation data, based on the following methods: neural networks, decision trees and logistic regression.
But before moving to the predictive models it was necessary to do the dimensionality reduction through Metadata Manual method. This method is used in the middle of an analytical process, with the objective of maintaining the relevant variables to the analysis.

Variables - Meta

(none) not Equal to

Columns: Label Mining Basic Statis

Name	Hidden	Hide	Role	New Role	Level	New Level	New Order	New Report
DepVar	N	**Default**	Target	**Target**	Binary	**Default**	**Default**	**Default**
NumWebVisitsM	N	**Default**	Input	**Default**	Interval	**Default**	**Default**	**Default**
IMP_Income	N	**Default**	Input	**Default**	Interval	**Default**	**Default**	**Default**
Custid	N	**Default**	ID	**ID**	Interval	**Default**	**Default**	**Default**
NumDealsPurcha	N	**Default**	Input	**Default**	Interval	**Default**	**Default**	**Default**
NumCatalogPurc	N	**Default**	Input	**Default**	Interval	**Default**	**Default**	**Default**
MntProd	N	**Default**	Input	**Default**	Interval	**Default**	**Default**	**Default**
MntRackets	N	**Default**	Input	**Default**	Interval	**Default**	**Default**	**Default**
Marital_Status	N	**Default**	Input	**Default**	Nominal	**Default**	**Default**	**Default**
AcceptedCmp	N	**Default**	Input	**Default**	Interval	**Default**	**Default**	**Default**
Teenhome	N	**Default**	Input	**Default**	Interval	**Default**	**Default**	**Default**
Recency	N	**Default**	Input	**Default**	Interval	**Default**	**Default**	**Default**
MntTShirts	N	**Yes**	Input	**Rejected**	Interval	**Default**	**Default**	**Default**
MntSneakers	N	**Yes**	Input	**Rejected**	Interval	**Default**	**Default**	**Default**
MntHats	Y	**Yes**	Rejected	**Rejected**	Interval	**Default**	**Default**	**Default**
MntPremium_Br	Y	**Yes**	Rejected	**Rejected**	Interval	**Default**	**Default**	**Default**
Kidhome	N	**Yes**	Input	**Rejected**	Interval	**Default**	**Default**	**Default**
Z_CostContact	Y	**Yes**	Rejected	**Rejected**	Interval	**Default**	**Default**	**Default**
Year_Birth	N	**Yes**	Rejected	**Rejected**	Interval	**Default**	**Default**	**Default**
dataobs	N	**Yes**	ID	**Rejected**	Interval	**Default**	**Default**	**Default**
Z_Revenue	Y	**Yes**	Rejected	**Rejected**	Interval	**Default**	**Default**	**Default**
NumStorePurcha	N	**Yes**	Input	**Rejected**	Interval	**Default**	**Default**	**Default**
MntWatches	N	**Yes**	Input	**Rejected**	Interval	**Default**	**Default**	**Default**
RMntFrq	N	**Yes**	Input	**Rejected**	Interval	**Default**	**Default**	**Default**
NumWebPurchas	N	**Yes**	Input	**Rejected**	Interval	**Default**	**Default**	**Default**
Complain	N	**Yes**	Input	**Rejected**	Binary	**Default**	**Default**	**Default**
Dt_Customer	N	**Yes**	Input	**Rejected**	Interval	**Default**	**Default**	**Default**
Age	N	**Yes**	Input	**Rejected**	Interval	**Default**	**Default**	**Default**
Element1	Y	**Yes**	Rejected	**Rejected**	Nominal	**Default**	**Default**	**Default**
Element2	Y	**Yes**	Rejected	**Rejected**	Nominal	**Default**	**Default**	**Default**
Education	N	**Yes**	Input	**Rejected**	Nominal	**Default**	**Default**	**Default**
AcceptedCmp2	N	**Yes**	Input	**Rejected**	Binary	**Default**	**Default**	**Default**
AcceptedCmp3	N	**Yes**	Input	**Rejected**	Binary	**Default**	**Default**	**Default**
AcceptedCmp1	N	**Yes**	Input	**Rejected**	Binary	**Default**	**Default**	**Default**

Figure 20: Metadata Node with variable selection for the predictive models

As stated at the beginning of this topic and immediately after the execution of the manual metadata node, the predictive models will be used.

The Logistic regression algorithm aims to predict the output values based on input features from the data fed in the system. In this type of algorithm, the output is always with a value less than one in the form of probability distribution. In this project it was applied the Stepwise Regression, which aims to adjust the regression models with predictive variables chosen by an automatic process.

Neural Networks are computing systems inspired by the biological neural networks that constitute animal brains. This kind of algorithms learns to perform tasks by considering examples. In comparison with logistic regression they can accommodate a wider variety of nonlinear relationships between input and target variables. Here it is essential to decide the complexity of the model, that is the highest of hidden number of units, the highest the model's complexity. For this work we decided to set up five neural networks with 1 to 5 hidden units.

Decision Trees are a decision support that uses a tree-like model of decisions and their possible consequences, that helps to identify a most likely strategy to reach a goal. Each branch of the decision tree represents the result of the test and each leaf node represents a class label, that means, the decision taken after calculating all attributes. The decision trees

can also be referred to as Regression and Classification Tree and even as a non-parametric classifier. Decision trees are simple models easy to interpret and they can return values even with little information, but at the same time, they can be unstable, because a small change in the data can lead to a large change in the structure.

Model Name	Branches	Depth
Decision Tree	2	6

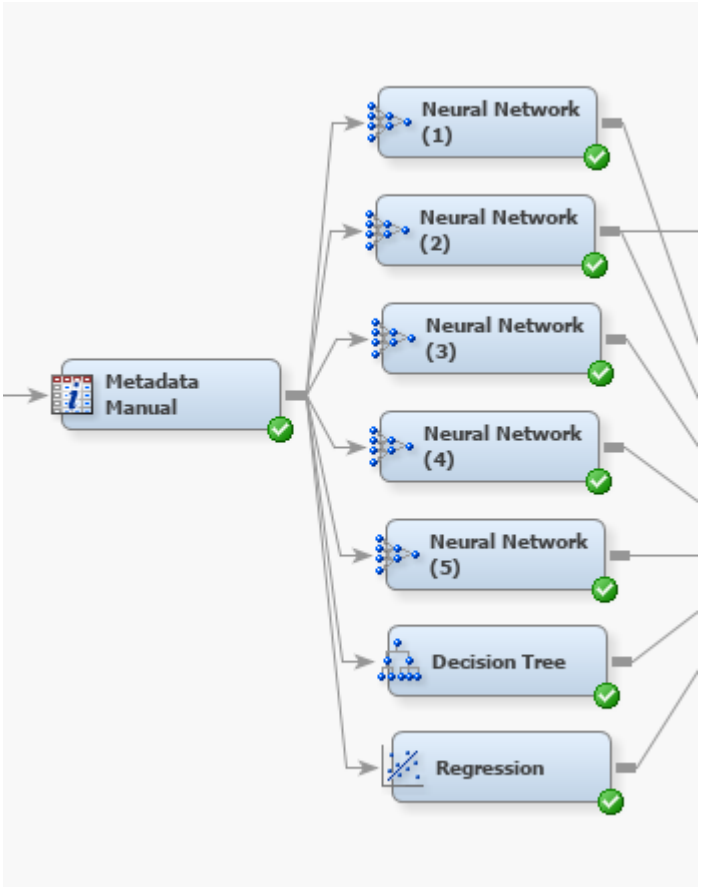

Figure 21: Example of Metadata Manual and predictive models

5.1.1.Model Assessment

The Model Comparison node enables us to compare the performance of competing models using various benchmarking criteria.

This first assessment took into consideration the Receiver Operating Characteristic (ROC) curves, which graphically display sensitivity versus 1-specificity, or the true positive rate versus the false positive rate.

In this case, both the training and the validation dataset results are presented (Figure 22). It should be noted, though, as previously mentioned, that it is the validation set that should be used to assess the adequacy of the models.

This is the reason why, despite Neural Network with 5 hidden units being the model which presents a higher ROC index, of 0.96 (Table 6), the Neural Network with 2 hidden units seems to be the one that performs the best against the validation dataset.

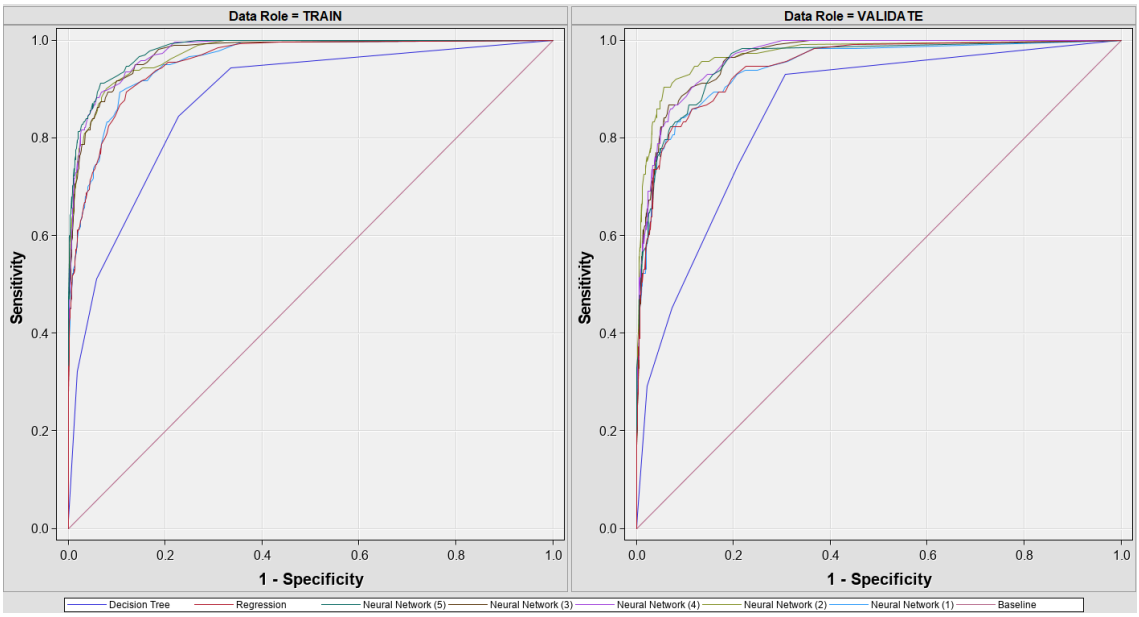

Figure 22: ROC Curve for the test and validation datasets

Table 6: ROC Index for predictive models

Model Description	ROC Index
Neural Network (5)	0.98
Neural Network (4)	0.976
Neural Network (3)	0.973
Neural Network (2)	0.972
Neural Network (1)	0.956
Regression	0.955
Decision Tree	0.876

These findings can be corroborated on the second assessment node (Estimated Profit Curve node), which consists on the determination of the financial projections of using each model for decision making. These results point to Neural Network with 2 hidden units being the most profitable model, with an estimated profit of 4.913€.

The graphical representation of these results is displayed in Figure 23, below, and the complete results are provided in the separated file "Group_12_PM.xls".

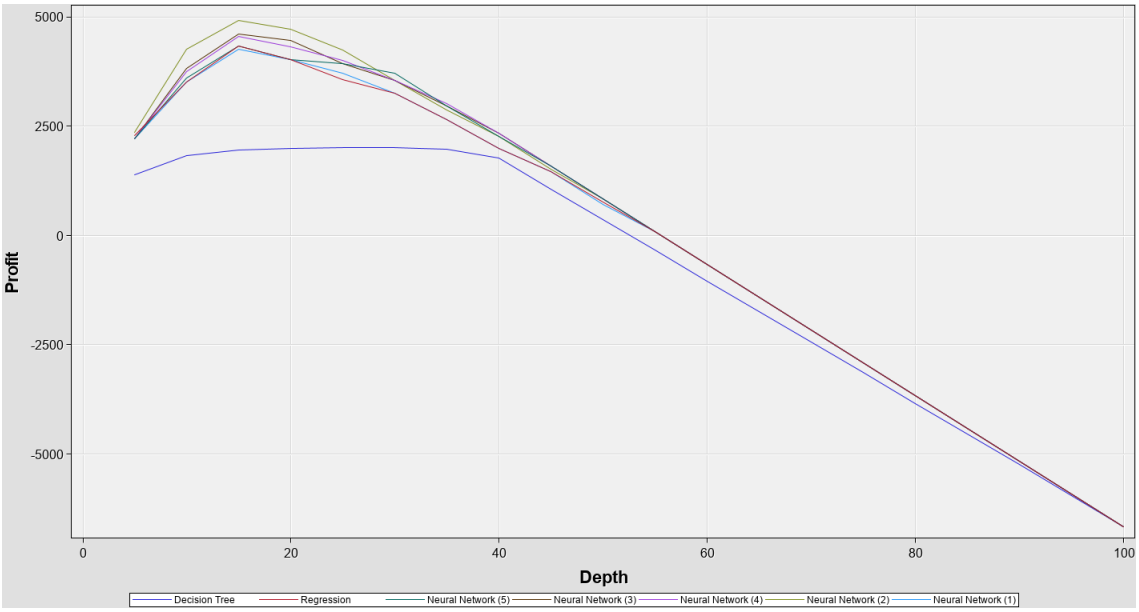

Figure 23: Estimated profit curves

5.1.2.Model Deployment

After the selection of the most profitable model (Neural Network with 2 hidden units), it is now possible to deploy the predictive model to the remaining customers.

For that, the corresponded dataset should be imported, using the LaGoste Score file import node. It is on the following node, the Score node, that the selected model will be applied to the new dataset.

Finally, a contact list must be extracted using the Contact list (Extract) node. This should represent the customers that are more probable to positively respond to the present campaign, thus maximizing the estimated profit.

According to the previous results, the maximum profit should be obtained by contacting the top 15% of the customer base, which translates to 750 customers to be contacted, with an average positive response of 73.5%, i.e., 551 sales. As aforementioned, this campaign should result on a final profit of 4.913€.

The full contact list is presented on the "Group_12_PM.xls" file.

6. Conclusions

This project developed a predictive model which can support marketing decisions, on a direct marketing area, based on data mining techniques. When this approach is correctly performed it is a major competitive advantage.
The process of modelling involves applying data mining technology to customers data to create a specialized model that gives every customer a probability score. This score predicts the action that a customer will take, for example predict which customers are most likely to accept an offer, which is the case presented in this project.
The main criterion used to choose the variables and the predictive model was based on profit maximization which is the main business goal in this project.

Given the efficiency improvement goal, this work proved essential, passing from 12,6% to more than 73,5% positive responses, which represents a remarkable improvement that could change the future of the company.

The Neural Network with 2 hidden units will be an easy “sale” to convince the chief marketing officer, because even considering that this model is not easy to explain to management, it is a strong predictive model. Additionally, the financial gain will be an extreme important factor for the decision.

7. Appendixes

Appendix 1: LaGoste Project Diagram

Appendix 2: Impute Node

This node is used to replace missing values with methods available like: the mean/median; random imputation, mode imputation and others.
In this project the method chosen was the use of the median value since the only missing values were from interval variables. The missing values of these interval variables are replaced by the median for that particular variable based on the remaining observations (Figure 24)

Imputation Summary
Number Of Observations

Variable Name	Impute Method	Imputed Variable	Impute Value	Role	Measurement Level	Label	Number of Missing for TRAIN
Income	MEDIAN	IMP_Income	91114	INPUT	INTERVAL	Income	39
MntHats	MEDIAN	IMP_MntHats	20	INPUT	INTERVAL	MntHats	58
MntPremium_Brand	MEDIAN	IMP_MntPremium_Brand	34	INPUT	INTERVAL	MntPremium_Brand	47

Figure 24: Filter limits for interval variables with outliers

Appendix 3: Spearman Rank Correlation (Correlation Matrix Node)

PEARSON

NAME	AcceptedCmp	Age	Custid	Dt_Customer	Frq	HigherEducation	Imputed: Income	Imputed: MntHats	Imputed: MntPremium_Brand	Incoherent	Kidhome	MntProd	MntRackets	MntSneakers	MntTShirts	MntWatches	NumCatalogPurchases	NumDealsPurchases	NumStorePurchases	NumWebPurchases	NumWebVisitsMonth	RMntFrq	Recency	Teenhome	Observation Number
AcceptedC...	1	0.070753	-0.01095	-.000846	0.317882	0.055873	0.344737	0.156315	0.196285	.	-0.17124	0.403816	0.327411	0.435415	0.148628	0.184959	0.371186	-0.021	0.177418	0.24387	-0.13525	0.377913	0.03895	-0.11723	-0.01079
Age	0.070753	1	-0.02015	-0.01832	0.196059	0.177351	0.246836	0.070918	0.057986	.	-0.26207	0.170654	0.124743	0.196196	0.063151	0.046089	0.156518	0.054913	0.147127	0.18116	-0.1363	0.168156	0.028854	0.342004	-0.02028
Custid	-0.01095	-0.02015	1	0.01286	-0.02045	-0.00172	-0.01093	-0.01532	0.002206	.	0.007152	-0.02645	-0.03261	-0.00944	-0.02388	-0.04456	-0.00829	-0.01811	-0.01327	-0.02996	-0.0133	-0.02409	0.014479	-0.01263	0.999921
Dt_Customer	-.000846	-0.01832	0.01286	1	-0.17307	0.02205	-0.01713	-0.11538	-0.19039	.	0.029399	-0.17903	-0.12005	-0.19257	-0.1205	-0.09214	-0.1227	-0.24315	-0.08204	-0.23475	-0.2728	-0.17817	-0.02926	-0.02167	0.012818
Frq	0.317882	0.196059	-0.02045	-0.17307	1	0.138501	0.776621	0.521264	0.493123	.	-0.55594	0.804517	0.642704	0.767816	0.532784	0.508565	0.808912	0.082089	0.86429	0.773681	-0.42489	0.706745	0.05193	-0.01348	-0.0199
HigherEduc...	0.055873	0.177351	-0.00172	0.02205	0.138501	1	0.210455	0.013461	-0.01575	.	-0.08452	0.136999	0.101787	0.176992	0.013475	-0.01098	0.090233	0.004851	0.118246	0.132058	-0.08415	0.140318	0.027236	0.121197	-0.00145
IMP_Income	0.344737	0.246836	-0.01093	-0.01713	0.776621	0.210455	1	0.554836	0.423579	.	-.5558	0.827629	0.733268	0.727839	0.562354	0.547657	0.732818	-0.14556	0.668434	0.493338	-.6662	0.810487	0.060969	0.007777	-0.0108
IMP_MntHats	0.156315	0.070918	-0.01532	-0.11538	0.521264	0.013461	0.554836	1	0.341413	.	-0.37144	0.651439	0.563106	0.404093	0.558519	0.541724	0.535434	-0.16393	0.46424	0.265872	-0.46152	0.636192	0.035419	-0.2127	-0.01537
IMP_MntPr...	0.196285	0.057986	0.002206	-0.19039	0.493123	-0.01575	0.423579	0.341413	1	.	-0.29292	0.450743	0.389701	0.379439	0.369377	0.359583	0.455094	0.0791	0.367089	0.394123	-0.22593	0.443844	0.03244	-0.02193	0.002333
Incoherent	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.
Kidhome	-0.17124	-0.26207	0.007152	0.029399	-0.55594	-0.08452	-0.5558	-0.37144	-0.29292	.	1	-0.55213	-0.45676	-0.49769	-0.38452	-0.37485	-0.51105	0.278547	-0.51202	-0.32656	0.502509	-0.54256	-0.04771	-0.04209	0.007008
MntProd	0.403816	0.170654	-0.02645	-0.17903	0.804517	0.136999	0.827629	0.651439	0.450743	.	-0.55213	1	0.880622	0.895257	0.644299	0.635194	0.7874	-0.13585	0.65899	0.518884	-0.53718	0.972118	0.062065	-0.20299	-0.02614
MntRackets	0.327411	0.124743	-0.03261	-0.12005	0.642704	0.101787	0.733268	0.563106	0.389701	.	-0.45676	0.880622	1	0.634732	0.601981	0.586342	0.701765	-0.22144	0.532553	0.332375	-0.57112	0.86249	0.053159	-0.3104	-0.03239
MntSneakers	0.435415	0.196196	-0.00944	-0.19257	0.767816	0.176992	0.727839	0.404093	0.379439	.	-0.49769	0.895257	0.634732	1	0.423477	0.395694	0.691926	-0.00711	0.60783	0.587071	-0.3578	0.863031	0.05916	-0.04373	-0.00911
MntTShirts	0.148628	0.063151	-0.02388	-0.1205	0.532784	0.013475	0.562354	0.558519	0.369377	.	-0.38452	0.644299	0.601981	0.423477	1	0.567613	0.53863	-0.16584	0.474354	0.281087	-0.4639	0.627118	0.053596	-0.21126	-0.02395
MntWatches	0.184959	0.046089	-0.04456	-0.09214	0.508565	-0.01098	0.547657	0.541724	0.359583	.	-0.37485	0.635194	0.586342	0.395694	0.567613	1	0.504268	-0.16591	0.456533	0.274213	-0.44843	0.629572	0.033172	-0.21107	-0.04435
NumCatalo...	0.371186	0.156518	-0.00829	-0.1227	0.808912	0.090233	0.732818	0.535434	0.455094	.	-0.51105	0.7874	0.701765	0.691926	0.53863	0.504268	1	-0.10746	0.558315	0.431269	-0.56228	0.710825	0.046284	-0.16847	-0.00786
NumDeals...	-0.021	0.054913	-0.01811	-0.24315	0.082089	0.004851	-0.14556	-0.16393	0.0791	.	0.278547	-0.13585	-0.22144	-0.00711	-0.16584	-0.16591	-0.10746	1	0.013429	0.31816	0.40486	-0.12465	-0.00434	0.425275	-0.01799
NumStoreP...	0.177418	0.147127	-0.01327	-0.08204	0.86429	0.118246	0.668434	0.46424	0.367089	.	-0.51202	0.65899	0.532553	0.60783	0.474354	0.456533	0.558315	0.013429	1	0.508986	-0.45925	0.568195	0.038514	-0.01267	-0.01299
NumWebP...	0.24387	0.18116	-0.02996	-0.23475	0.773681	0.132058	0.493338	0.265872	0.394123	.	-0.32656	0.518884	0.332375	0.587071	0.281087	0.274213	0.431269	0.31816	0.508986	1	0.015489	0.448574	0.043419	0.157912	-0.02927
NumWebVi...	-0.13525	-0.1363	-0.0133	-0.2728	-0.42489	-0.08415	-0.6662	-0.46152	-0.22593	.	0.502509	-0.53718	-0.57112	-0.3578	-0.4639	-0.44843	-0.56228	0.40486	-0.45925	0.015489	1	-0.53762	-0.05359	0.18888	-0.01304
RMntFrq	0.377913	0.168156	-0.02409	-0.17817	0.706745	0.140318	0.810487	0.636192	0.443844	.	-0.54256	0.972118	0.86249	0.863031	0.627118	0.629572	0.710825	-0.12465	0.568195	0.448574	-0.53762	1	0.05877	-0.20086	-0.02397
Recency	0.03895	0.028854	0.014479	-0.02926	0.05193	0.027236	0.060969	0.035419	0.03244	.	-0.04771	0.062065	0.053159	0.05916	0.053596	0.033172	0.046284	-0.00434	0.038514	0.043419	-0.05359	0.05877	1	-0.00611	0.014617
Teenhome	-0.11723	0.342004	-0.01263	-0.02167	-0.01348	0.121197	0.007777	-0.2127	-0.02193	.	-0.04209	-0.20299	-0.3104	-0.04373	-0.21126	-0.21107	-0.16847	0.425275	-0.01267	0.157912	0.18888	-0.20086	-0.00611	1	-0.01262
dataobs	-0.01079	-0.02028	0.999921	0.012818	-0.0199	-0.00145	-0.0108	-0.01537	0.002333	.	0.007008	-0.02614	-0.03239	-0.00911	-0.02395	-0.04435	-0.00786	-0.01799	-0.01299	-0.02927	-0.01304	-0.02397	0.014617	-0.01262	1