Xsell Regression Model

Propensity to Cross sell –Multi product Insurance Industry Case

© **Pristine** – www.edupristine.com

Business Problem:

Mr. A is a strategic marketing head of a Multi product Insurance Company. He has a large database of general insurance (GI) customers. He wants to cross sell health insurance products to his general insurance customers. The limited budget and low responses of the healthcare campaigns force him to think of an approach for targeted marketing.

He needs to identify and capture the characteristics of customers who buy health insurance.

The implementation of the same will involve design and development of a predictive model, selection of the population as model input and deployment and refresh process for regular use.

Background:

Predictive modeling is a form of data mining. Data mining is the “analysis of observational datasets to find unforeseen relationships and to summarize the data in ways that are useful to the data owner.” Predictive modeling takes these relationships and uses them to make inferences about the future.

There are different types of predictive models (neural network, GLM, Logistic etc). The appropriate model depends on the business scenario and the data. A logistic model is more accurate than any other predictive modeling methods when the dependent variable has binary response. Based on data characteristics, logistic models determine the propensity of response.

The dependent variable should be categorical with binary response.

Logistic regression can be used only with two types of target variables:

1. A categorical target variable that has exactly two categories (i.e., a binary or dichotomous)

2. A continuous target variable that has values in the range 0.0 to 1.0 representing probability values or proportions.

Logistic Function:

The logistic function: f (Z) =exp (Z)/1+exp (Z)

Where Z= β0+ β1X1+ β2X2+ β3X3+………+ βkXk

Where β0 is called the "intercept" and β1, β2, β3, and so on, are called the "regression coefficients" of independent variables x1, x2, x3 respectively.

For any data model the input data extracted is most important to build a robust model. We should collect the demographic and historical information in the data and do some initial analysis like data audit, exploratory data analysis and bi-variate profiling for better understanding of the data. Next step involves analysis of the relationship in the data. The key objective of analysis is to establish a relation between the dependent and independent variables using a logistic regression. Once the model is built on training data we need to validate that model on validation data and generate scores as the output.

Workflow:

Pull the relevant variables from the database. This step is called as Data Extraction.

The data is extracted by writing the SQL codes. The same is imported to the “R” for further analysis.

- See the excel/csv file for data.

Initial Data Analysis:

Before Data modeling, we should go through the data for better understanding of the available data and the business problem. Meantime it will helps in identifying missing values and outliers if there in your data.

Data Preparation Steps:

The data preparation steps involves following steps:

1. Data Audit and EDA

2. Data Profiling

Data Audit and EDA: - The data audit report is the initial report that we prepared to understand the data well. This report will consists of descriptive statistics for all the variables in the dataset and also will helps in identifying the missing value and levels of categorical predictive variables. This data Audit report serves as base for assessing the quality of the data we extracted and obtained from the client. Based on this report we can request for additional data which we seems to be important for our analysis and it will also helps in dropping some insignificant variable. Refer the data audit report fields below for better understanding.

Variable
Data Type
Label
Total no. of records in dataset
No. of valid (non missing) cases
No. of missing values
Fill Rate
Count of Unique Values
No. of Levels
Mean
Std deviation
Median
Mode
Minimum
Q1(25th percentile)
Q3(75th Percentile)
99th Percentile
Maximum

Data Preparation Steps:

Data Profiling:-Bivariate profiling assist in finding the frequency of each categorical variable with respect to the response variable. This would facilitate in binding/ grouping the categories which have same response rate so that the effect of that particular category can be captured in the model.

We prepare a report with following fields to further work on it.

Variable
Field categories
Frequency
Responses
Responses %

Data Treatment:

The first stage of any statistical modeling consists of data treatment activities. Approximately 80% of the entire modeling time is consumed by the data treatment techniques. Here we check the hygiene factor of our independent variables and try to make the data as exploitable as possible.

Before going to data treatment one has to find out the correlation between variables means finding the relationship of predictors with the response variable and also to find out the inter correlation among predictors. From this analysis we can exclude some of the predictors which are not important for the model building based on the significant correlation values. The first step of variable reduction happens in this stage and next is on basis of multicollinearity check. The variables selected from this step will undergo for further data treatment like missing value, extreme value treatment and multicollinearity check.

Missing value treatment:

We should check that the independent variables have sufficient information to establish a significant relation with the dependent variables.

Some of the basic rules for Missing value treatment are as below:

1. If the independent variables have a large amount of missing value (More than 40%-50%), we drop that independent variable from our analysis, since no relation can be established between that independent variable and the dependent variable in question.

2. If the percentage of missing value lies between 10% -40%, we try to establish a separate relation between the dependent and independent variables to understand any hidden pattern.

3. If our predictors are categorical variable, then we can make that missing values as one category but we will miss the information since that category will not comes significant in the model so better treat the missing value with those category which has highest frequency among all the categories of a variable.

4. For quantitative independent variable, treat the missing values with central tendency like mean, median and mode value of that variable.

5. Various other methods like exploration, regression method etc. has also been used to treat the missing values.

Note: These missing values are represented by dots ('.') for numerical variable and blank for categorical variables in the SAS Language.

Extreme value treatment:

This step is done to understand the distribution of our independent variables. Presence of an abnormal value in one of the independent variables can affect the entire analysis (Leverage variable). The extreme values are treated by capping. Capping is required as sometimes, the variable may contain extreme values corresponding to some observations; whereas in reality, such values are unlikely to exist. Such values may be a result of wrong keying the data or may represent the default values. There may be cases of negative values which is logically incorrect for a given variable. In such cases, these values need to be capped because they may affect the mean of the variable drastically.

- Some basic rules for capping are as below:
  - Don’t cap the value unless it is unrealistic
  - Cap it to the next highest/lowest (realistic) value.
  - Cap at a value so that the continuity is maintained.

Multicollinearity:

When two or more independent variables are related between them, we tell that they have multicollinearity among each other. In technical terms, we say the one variable can be explained as a linear combination of other variables. Multicollinearity among independent variables does not allow the independent variables to explain their impact on the dependent variable optimally due to high internal impact. Keeping collinear variables in the model makes it unstable. In such a scenario we drop one of the variables from the model. Multicollinearity among the variables indicates that these explanatory variables are carrying a common set of information in explaining the dependent variable.

Detection of Multicollinearity:

Variance Inflation Factor: We generally test the presence of multicollinearity using Variance Inflation Factor (VIF). Variance Inflation factor (VIF) is obtained by regressing each independent variable, say X on the remaining independent variables (say Y and Z) and checking how much of it (of X) is explained by these variables. Hence VIF = 1 / (1 – R2).

If the VIF value be greater than 2 we drop the variable from the model.

Steps in Multicollinearity Check:

- Add all the independent variables in the model to explain the response
- Check for the variable which has highest VIF
- Keeping in that variable in mind, go to correlation table .
- Identify the variable (has highest VIF) has the highest correlation with some other variable .
- Drop the variable with higher VIF and repeat the procedure till you get the VIF < 2

Data Modeling:

Once the data treatment is over we go ahead with the model building:

Development (Training) and Validation (Testing) Sample:-Before building the model on the data, divide entire data in the ratio of 70:30 as development sample and validation sample. Development sample is used to develop the model whereas validation sample will be used to check validity of the model. Build the model on development sample data and use the estimates obtained from this model to score the validation sample. If the response captured from the validation sample is nearly equal to the response captured from the development sample then we can say that the model is robust in predicting the responses for future dataset.

Logistic Regression Model Building:-Logistic regression technique is used to assess the impact of independent variables and probability of event of interest. The approach is explained in the following steps

- Create the dummies and slope dummies for the categorical independent data if desired. Because most of the statistical analysis tools like E-Guide, E-Miner will take string variables directly. So in such cases no need to create dummies and here one of the categories will be converted into base category for comparison.
- Data transformation for continues independent variables if necessary.
- Run logistic regression and check the goodness of fit of the model.

Model Fit Criteria:

1. Use the Deviance or Hosmer & Lemeshow test statistics to check the validity of the model. Higher the "P" value better is the model. Proceed to next steps only if we have higher value of P.

2. Test the null hypothesis for the independent variables, i.e. all $\beta = 0$. P value should be significant (i.e. $p < 0.05$) to reject the null hypothesis and prove that β values are not equal to 0.

3. Check the concordance and Tie. The rule of thumb test is (Concordance+ ½ Tie) should be greater than 60%.

4. Check the significance of the estimates of each of the variable. If any of the estimates are not significant, variable with highest P value will be dropped and steps i to vi are repeated with the new set of variables. This process will continue until all the variables in the model have significant estimates.

5. Frame the equation with the significant variables. Odds ratio and probability value for each of the profile is estimated.

6. Specificity and Sensitivity of the model is assessed and ROC (Receiving Operating Characteristic) graph is plotted. Area under the ROC is an indication of how well the classification of good in to good and bad to bad is decided by the identified model.

Model Fit Criteria:

7. Coefficient Stability: Coefficient stability is checked across development and validation sample. Once the model is performing satisfactorily on development sample, we use the same set of variables to model the validation sample. A robust model should perform equally well on validation sample too. Hence, the coefficients should be in a close range and should be of same sign.

8. Concordance: Consider a set of 100 individuals out of which 10 are the responders (denoted by 1) and 90 are non-responders (denoted by 0). Now we construct pairs for each responder with every non-responder. Hence, we get 900 such pairs (10*90 = 900). Using the model under development, we calculate the predicted response rate for each responder and non-responder in every pair. If responder's predicted probability is greater than non-responder's predicted probability, then the pair is concordant. If it is vice versa, then the pair is discordant and if both are equal, then the pair is tied. For a good model, the percent concordant pair lies above 65%.

9. Gini Coefficient: The Gini coefficient is one which is used to test the model accuracy. It is calculated by using following formula. For good model the Gini coefficient should be in the range of 40-60%.

Gini=2C-1   Where C= Area under the curve (ie Concordance+1/2 of Tie)

.

Model Fit Criteria:

10. Scoring: Satisfaction of the model comes when the model is doing well in terms of rank ordering, coefficient stability, Goodness of fit, Concordance and capturing both on development and validation samples

Now, take the coefficients of variables obtained from a model run on development sample and use it to predict response rate of validation sample. This method is known as scoring of the model. Scoring provides a good idea about how the model will perform when applied to another data set. Here, we are concerned about the capturing of the responders, say in first 40 % of the population.

The model is used to predict the response rate for a set of new data is taken from a different time frame to test the validity of the rules suggested by the model. The model will be applicable to the profiles similar to the once already present in the sample data used for model development. Model validation is performed by taking the optimum threshold level of probability.

Scorecard:

Once the model is finalized after checking all the model validation criteria, use the coefficient obtained from the model developed on development data to prepare the scoring code. The scoring code will be generated using different platform like teradata, SAS etc .This scoring code will be used for producing the score for future data sets of the same profiles and will facilitate in targeting the customer who have likely to respond for campaigns.

R code for Xsell model

```
###set working directory

setwd("C:/predictivemodelingfolder/xsellmodel")

### import required packages for modeling

library(car)

library(glmnet)

library(SamplingStrata)

library(sampling)

###Read the data

cust_data<-read.csv("cust_data.csv")
```

R code for Xsell model

```
### See the data summary (verify Data)

### Retrieve top 5 data

head(cust_data)

### Retrieve bottom 5 data

tail(cust_data)

### Review Summary of dataset

summary(cust_data)

### Retrieve frequency  of  character variable against the dependent variable

table(cust_data$Gender,cust_data$Responder)
```

R code for Xsell model

```
###Multicollinearity check

### Retain numeric variables and find correlation matrix  for them

dat1<- cust_data[,c(3,9,10,11,15,17)]

correlation <- cor(dat1,dat1)

### Calculate variance inflation factor to reduce varaibles based on multicollinearity

vif1 <- vif(lm(Responder ~ Age + WSI + IncomeGrp + No.of.prod1 + MSL_prod1 +
No.of.prod2+MSL_prod2 + num_of_cars + Family_doctor , data=cust_data))

### View the results of vif and ensure all the VIF values are lower than 2

### If VIF >2 drop the variable with high correlation and run vif again

read(vif1)
```

R code for Xsell model

```
###Missing value and capping treatment along with categorization

cust_data$GRP_age <- ifelse(
cust_data$Age=="",1,ifelse(cust_data$Age>50,3,ifelse(cust_data$Age>25,2,1)))

cust_data$GRP_channel<-
ifelse(cust_data$Channel=="Direct",1,ifelse(cust_data$Channel=="Broker",2,1))

cust_data$GRP_gender<-
ifelse(cust_data$Gender=="Male",1,ifelse(cust_data$Gender=="Female",2,1))

cust_data$GRP_marital_status<-
ifelse(cust_data$Marital_status=="Yes",1,ifelse(cust_data$Marital_status=="No",2,1))

cust_data$GRP_FS_code<-
ifelse(cust_data$FS_code=="A",1,ifelse(cust_data$FS_code=="E",3,2))

cust_data$GRP_Prosperity_Index<-
ifelse(cust_data$Prosperity_Index=="High",1,ifelse(cust_data$Prosperity_Index=="Medium",2,2))
```

R code for Xsell model (Contd…)

```
cust_data$GRP_No.of.prod1<-
ifelse(cust_data$No.of.prod1>5,1,ifelse(cust_data$No.of.prod1<=5,2,2))

cust_data$GRP_No.of.prod2<-ifelse(cust_data$No.of.prod2>5,1,ifelse(cust_data$No.of.prod2
>2,2,1))

cust_data$GRP_MSL_prod1<-ifelse(cust_data$MSL_prod1>24,1,ifelse(cust_data$MSL_prod1
>12 ,2,1))
```

R code for Xsell model

```
## sorting the data and generating the sample data

cust_data <- cust_data[order(cust_data$Responder,decreasing = TRUE),]

samp =strata(cust_data,c("Responder"),size=c(213,1922), method="srswor")

samp<-getdata(cust_data,samp)

### Logistic Equation; Use samp  data  for generating results

logistic1<- glm(Responder ~ GRP_age+ GRP_channel+ GRP_gender+ GRP_marital_status+
GRP_FS_code+ GRP_Prosperity_Index+ No.of.prod1+ No.of.prod2+ MSL_prod1 , data=samp)
```

EDU PRISTINE

Thank you!

Pristine

702, Raaj Chambers, Old Nagardas Road, Andheri (E), Mumbai-400 069. INDIA

www.edupristine.com

Ph. +91 22 3215 6191